

UNIVERSITÀ  
DEGLI STUDI  
DI PADOVA



DIPARTIMENTO  
DI INGEGNERIA  
DELL'INFORMAZIONE

**DIPARTIMENTO DI INGEGNERIA DELL'INFORMAZIONE**

**CORSO DI LAUREA IN INGEGNERIA INFORMATICA**

# **Sicurezza nei dispositivi IoT**

**Relatore:**

PROF. MAURO MIGLIARDI

**Laureando:**

Luca Sambin

**ANNO ACCADEMICO 2020 – 2021**

**Data di laurea**

22 Luglio 2021



*Alla mia famiglia*



# Sommario

IoT è un'abbreviazione del termine Internet of Things. Il termine descrive oggetti di uso quotidiano come le lampadine connesse a Internet. L'IoT è un campo che sta crescendo molto rapidamente con alcuni ricercatori e leader del settore che prevedono che ci saranno fino a 200 miliardi di dispositivi IoT connessi nel mondo entro il 2020. Molti dispositivi IoT sono sviluppati da aziende più piccole che cercano di capitalizzare un'esigenza specifica nel mercato. Per questo motivo, le aziende potrebbero favorire il lancio di un prodotto il più rapidamente possibile, il che potrebbe implicare che tali dispositivi potrebbero non essere stati adeguatamente testati.

Il mercato IoT e Smart Home sta attualmente vivendo una rapida crescita e tutti i segnali indicano che continuerà in futuro. Questa tesi si concentra sul test delle vulnerabilità nei dispositivi IoT che possono essere trovati in un ambiente domestico intelligente. Lo scopo di questa tesi è quello di creare maggiori conoscenze sulle vulnerabilità che si possono trovare nei dispositivi connessi a Internet che vengono utilizzati quotidianamente.

Questa tesi include esperimenti utilizzando OpenVAS, che è uno scanner di vulnerabilità sviluppato da Greenbone Security utilizzato per rilevare le vulnerabilità nei dispositivi IoT. I dispositivi testati sono Hassio (Raspberry Pi 3), Google Home Mini (prima generazione), Google Chromecast (prima generazione) e la telecamera IP Foscam FI9821W V2. Il firmware/software di tutti i dispositivi è aggiornato a maggio 2021.



# Indice

1	Introduzione.....	9
1.1	Storia ed espansione .....	11
1.2	Trend di mercato.....	13
1.3	IoT sfide per la sicurezza.....	16
1.4	Smart Home.....	18
1.4.1	Smart Home Scenario.....	19
1.5	Obiettivi di ricerca .....	20
1.5.1	Motivazione della ricerca .....	20
1.5.2	Obiettivo della ricerca .....	20
1.5.3	Domanda di ricerca.....	20
2	Smart Home.....	21
2.1	Proprietà della casa intelligente.....	21
2.2	Limiti delle risorse.....	22
2.3	Sicurezza e Test di vulnerabilità.....	23
2.4	Architetture e configurazioni.....	25
2.5	L'utente.....	29
3	Metodo.....	31
3.1	Descrizione dell'esperimento.....	32
3.2	Discussione sul metodo .....	32
3.3	Dispositivi utilizzati.....	34
4	Esperimento .....	39
4.1	Strumenti .....	41
4.2	Impostazioni .....	42
5	Risultati.....	43
5.1	Hassio .....	48
5.2	Google Home Mini (Prima generazione) .....	49
5.3	Google Chromecast (Prima generazione).....	50
5.4	Telecamera IP Foscam .....	52
5.6	Riepilogo dei risultati .....	57
6	Analisi e discussione .....	59
6.1	Analisi.....	59
6.2	Discussione.....	61
7	Conclusioni e lavoro futuro .....	65
	Bibliografia.....	67





# 1 Introduzione

Con il passare del tempo, un numero crescente di oggetti si connette a Internet, come gli spazzolini, le lampadine e i termostati [1]. Queste "cose" sono chiamate collettivamente dispositivi "IoT"; IoT è l'abbreviazione di Internet of Things, che descrive gli oggetti di uso quotidiano connessi a Internet [2].

Tecnicamente l'Internet of Things è un sistema di dispositivi informatici correlati, macchine meccaniche e digitali, oggetti, animali o persone a cui sono forniti identificatori univoci e la capacità di trasferire dati su una rete senza richiedere interazione con il computer [2].

L'Internet of Things è una delle più grandi rivoluzioni introdotte dalla rete globale negli ultimi tempi. Infatti, propone di fondere il mondo reale con quello virtuale creando un ambiente più intelligente. Un ambiente in grado di sentire, analizzare e adattarsi per rendere le nostre vite più semplici, sicure ed efficienti [3].



**Figura 1.1:** Alcuni esempi di applicazioni IoT in alcuni settori verticali [3].

Uno degli impieghi di tali dispositivi nella società potrebbe essere quello della produzione e distribuzione di energia rinnovabile, come visto nella figura precedente.

Infatti, i due acronimi presenti nella figura 1.1, si riferiscono nello specifico a:

- TSO è un acronimo di un operatore di trasmissione energetica (in inglese Transmission System Operator) che è un ente preposto alla trasmissione dell'energia sotto forma di gas naturale o di energia elettrica, usando opportune infrastrutture, a livello nazionale o regionale [97].
- DSO è un acronimo che indica i gestori del sistema di distribuzione (in inglese Distribution System Operator) che sono i soggetti responsabili della distribuzione e della gestione dell'energia dalle fonti di generazione ai consumatori finali [98].

Non si tratta di una nuova campagna pubblicitaria fatta per sponsorizzare una nuova tecnologia, ma di una realtà composta da un'infinità di piccole e grandi iniziative, che sfruttando le nuove tecnologie sono in grado di cambiare totalmente tutti gli aspetti della nostra vita.

Esistono innumerevoli esempi di applicazione, dalle smart home alla gestione di impianti di produzione industriale, passando dal monitoraggio e miglioramento di coltivazioni e allevamenti all'utilizzo in campo militare, risultando ben scalabili sia con i sistemi su scala globale e sia con i piccoli ambienti.

Le grandi aziende, leader nel settore informatico e in quello dei servizi, stanno sviluppando framework e soluzioni per l'implementazione e la messa in opera delle nuove possibilità offerte da questo nuovo mercato, investendo ingenti somme per non farsi sfuggire questa opportunità in piena fase di definizione e sviluppo.

## 1.1 Storia ed espansione

Nei primi anni 2000 nei laboratori "AutoID" del MIT venivano poste le basi per il concetto che sarebbe diventato la visione dell'internet delle cose; Kevin Ashton [4] in un articolo del RFID Journal scrisse: "Se avessimo computer che sapessero tutto quello che c'era da sapere sulle cose, utilizzando i dati che hanno raccolto senza alcun aiuto da parte nostra - saremmo in grado di tracciare e contare tutto e ridurre notevolmente sprechi, perdite e costi. Sapremmo quando le cose necessitano di essere sostituite o riparate e se fossero nuove o superate. Dobbiamo potenziare i computer con i propri mezzi per raccogliere informazioni, in modo che possano vedere, ascoltare e annusare il mondo da soli, in tutta la sua gloria casuale. La tecnologia RFID e dei sensori consente ai computer di osservare, identificare e comprendere il mondo, senza le limitazioni dei dati inseriti dall'uomo" [5][6].

Il concetto era semplice e potente, infatti se tutti gli oggetti della vita di ogni giorno fossero stati equipaggiati con identificatori e connettività wireless, questi oggetti avrebbero potuto comunicare tra di loro ed essere gestiti dai computer. A quel tempo non erano disponibili le tecnologie necessarie per realizzare questa visione e l'idea, per quanto allettante, rimase una visione futuristica. Successivamente, vi fu un periodo di calma dove gli interessati all'argomento rimasero "in pochi" più che altro come forma di ricerca [7].



**Figura 1.2:** Interesse nel tempo, analizzando l'interesse per nazione e argomenti correlati alle varie ricerche svolte su Google [8]

Negli ultimi anni, successivamente allo sviluppo di nuove tecnologie questa visione è diventata realizzabile con costi sostenibili e ragionevoli. Sono infatti nate diverse organizzazioni (AIOTI, IERC, IOT-WF, ...), finanziate dai vari paesi dell'unione europea e dalle società più rilevanti al mondo (Cisco, Ikea, Ibm, Qualcomm, ...), che assieme alle

vecchie (IEEE, ISO, ITU, ...) si propongono di promuovere, facilitare e guidare lo sviluppo di questo concetto garantendone, tra le altre cose, la sicurezza e la privacy.

Tali organizzazioni si pongono lo scopo di favorire ed accelerare l'adozione dell'IoT attraverso la ricerca e lo sviluppo in campi ben definiti, che vanno dalla politica a delle attività verticali e interdisciplinari focalizzate su argomenti chiave dell'IoT. Ad esempio nel 2018 l'organizzazione AIOTI, ha pubblicato raccomandazioni per le future priorità della ricerca nell'ambito dell'IoT dei programmi Horizon Europe e Digital Europe 2021-2027.

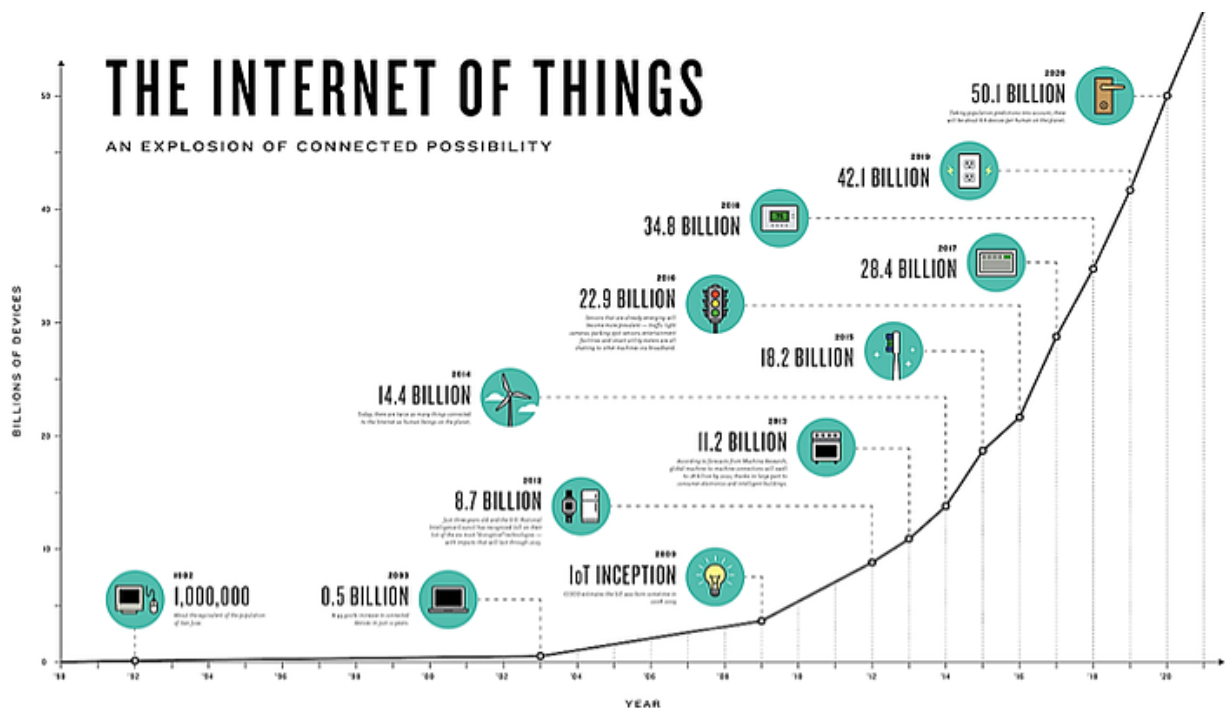
Proprio questo argomento è stato uno degli ultimi ad essere affrontato, forse perché va in conflitto con le strategie dei vari produttori, i quali puntano a raccogliere più dati possibili e nel modo più mirato possibile sui consumatori [9].

## 1.2 Trend di mercato

Secondo recenti previsioni, l'industria dell'IoT dovrebbe produrre 75,44 miliardi di dispositivi e generare 79,4 zettabytes di dati entro la fine del 2025 [10] [11].

Nel ventunesimo secolo, l'intelligenza artificiale e la tecnologia dei big data stanno guidando l'afflusso dei dati, e la tecnologia IoT ricoprirà un ruolo fondamentale in questi settori poiché li supporta a livello architetturale.

I dispositivi IoT stanno crescendo enormemente e continueranno a crescere nei prossimi anni grazie a nuovi sensori, che disporranno di una maggiore potenza di calcolo e di una connettività più affidabile, come mostrato nella figura sottostante.



**Figura 1.3:** Numero di dispositivi Internet of Things (IoT) attivi [12]

Tuttavia, l'utilizzo sempre più diffuso dei dispositivi IoT ha portato alla inevitabile condivisione di informazioni personali, portando così a una potenziale violazione della sicurezza, e di conseguenza ad una possibile compromissione della privacy degli utenti.

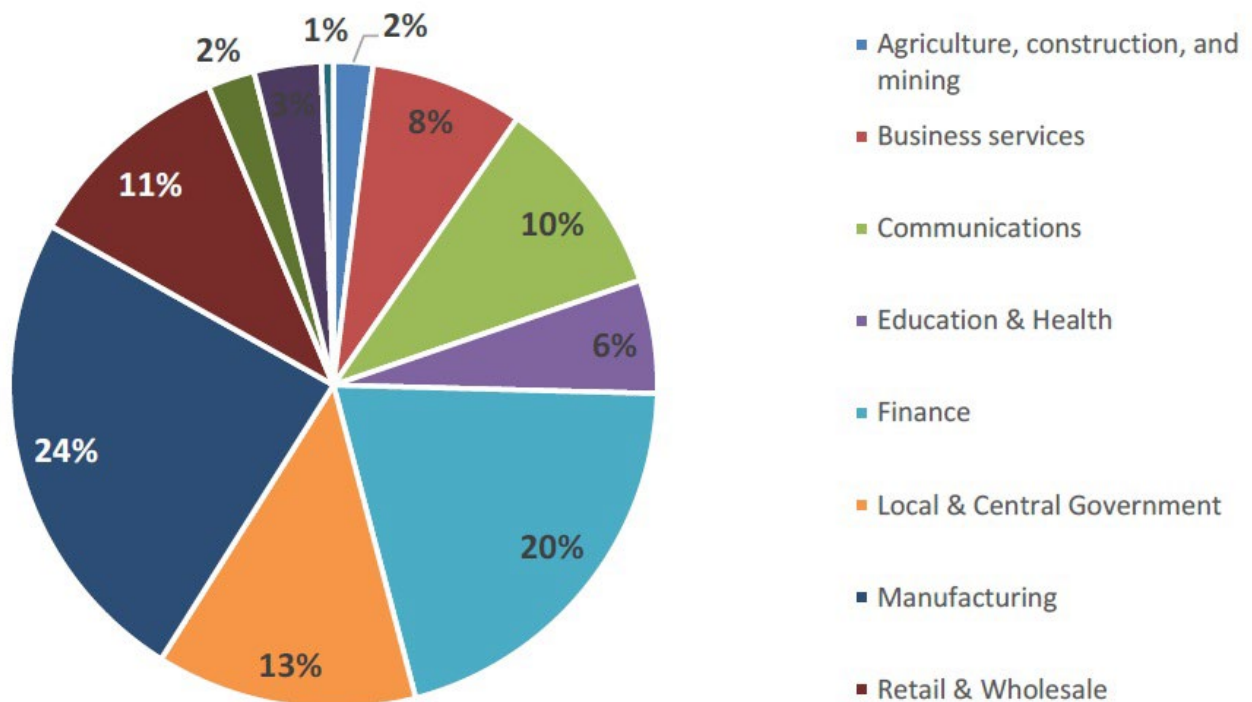
Infatti, nonostante la recente pandemia globale, che ha comportato un crollo della produzione e della realizzazione di dispositivi IoT, specialmente nel mercato asiatico, alcune delle previsioni fatte negli ultimi 5 anni risultano essere state rispettate a pieno, come quelle elencate in seguito.

Infatti già nel 2014, veniva previsto un impatto su tutti i settori produttivi, da parte dell'IoT.

In figura 1.3 e 1.4 due schemi con gli investimenti previsti durante quell'anno, divisi per settore produttivo.

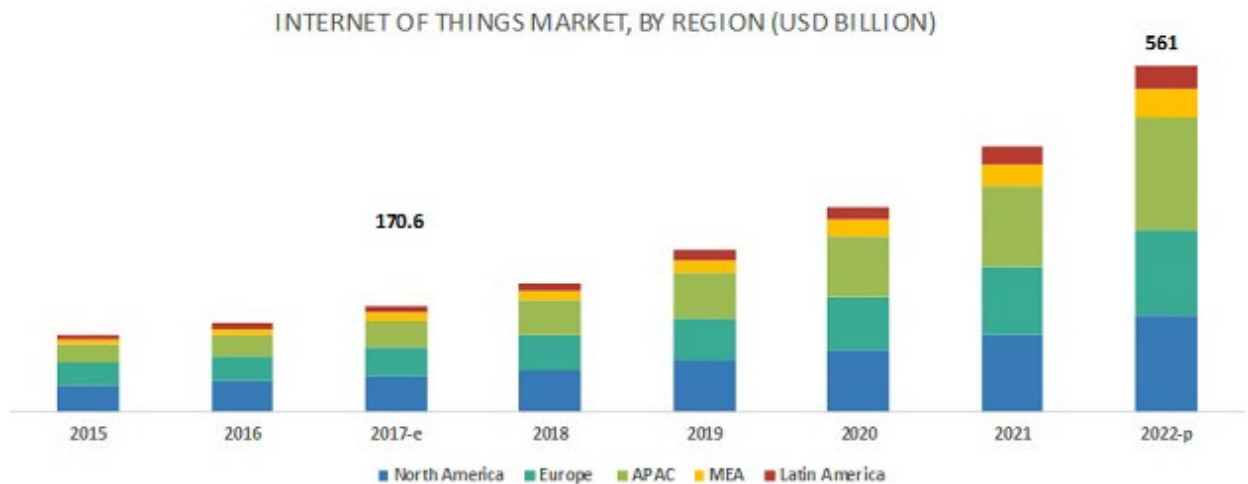
Country	2014	2020
Agriculture, construction, and mining	€ 7 311	€ 23 193
Business services	€ 28 334	€ 90 218
Communications	€ 37 388	€ 119 975
Education & Health	€ 22 060	€ 66 925
Finance	€ 73 709	€ 242 222
Local & Central Government	€ 49 742	€ 153 707
Manufacturing	€ 87 805	€ 286 539
Retail & Wholesale	€ 38 024	€ 124 412
Transport	€ 8 659	€ 27 728
Utilities	€ 10 630	€ 39 668
Others	€ 2 330	€ 7 017
<b>Total</b>	<b>€ 365 992</b>	<b>€ 1 181 603</b>

**Figura 1.4:** Dimensioni e previsioni del mercato IoT: scenario di base per mercato verticale (Milioni) [13]



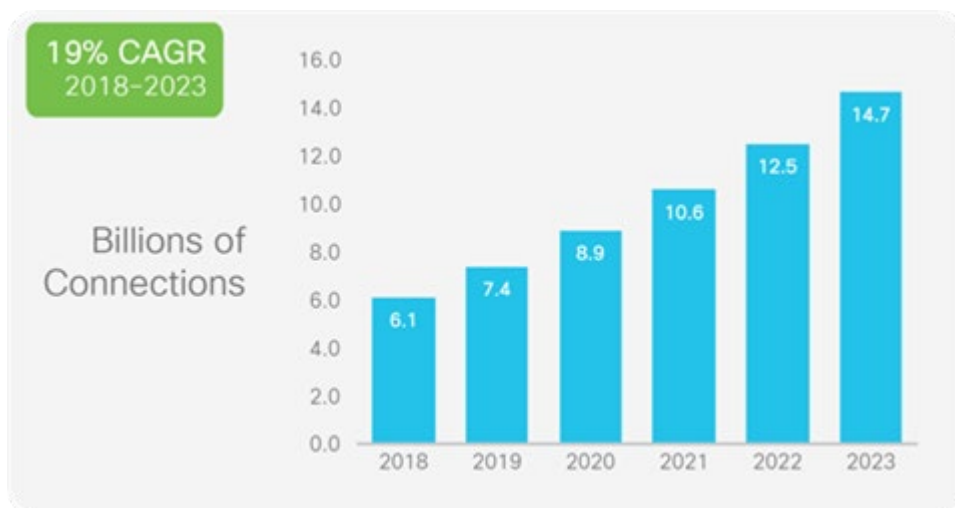
**Figura 1.5:** Dimensioni e previsioni del mercato IoT: scenario di base per mercato verticale (2020; %) [13]

Tornando ad oggi, Come si può vedere dalla figura seguente [14], si nota come sia ampia e in continua crescita, la quota di mercato per i vari continenti.



**Figura 1.6:** Analisi di MarketsandMarkets sui dispositivi IoT [14]

Inoltre, secondo Cisco [15], l'IoT sarà responsabile, a livello globale, dell'aumento delle connessioni M2M (Machine to Machine) che andranno a crescere di 2,4 volte, da 6,1 miliardi nel 2018 a 14,7 miliardi entro il 2023 (figura sottostante). Infatti ci saranno 1,8 connessioni M2M per ogni membro della popolazione mondiale entro il 2023.



**Figura 1.7:** Cisco, adozione globale di Internet e dispositivi connessi ad esso [15]

Oggigiorno, le maggiori attrattive di business in questo settore si possono raggruppare in 4 macro categorie: "Smart Manufacturing", "Smart Home", "Smart Health" e "Smart Customer

Experience". Parlando di "Smart Home" e "Smart Health" si nota subito quanto possa essere importante studiare ed approfondire le tematiche di sicurezza, delle tecnologie che andranno ad incidere su ambiti così cruciali della nostra vita.

## **1.3 IoT sfide per la sicurezza**

Diversi studiosi e professionisti del settore [16-24] affermano che la sicurezza è una delle grandi sfide nell'IoT. Il continuo aumentare dei dispositivi IoT richiede studi e test approfonditi per assicurarsi che tali dispositivi siano sicuri da usare per i consumatori. I dati che i dispositivi IoT archiviano e inviano possono essere personali e privati e quindi richiedono un'adeguata sicurezza.

Banafa [17] afferma che alcuni fornitori di IoT stanno rilasciando prodotti il più velocemente possibile con lo scopo di battere la concorrenza, vendendo le loro soluzioni innovative prima che chiunque altro possa farlo, il che può indicare che la sicurezza non sia sempre la priorità principale.

Perciò è fondamentale valutare la sicurezza di ogni dispositivo IoT, la quale sta diventando una delle più grandi sfide che esistono con tali dispositivi.

La ragione di ciò è il fatto che i dispositivi sono sempre connessi a Internet, il che implicata la possibile introduzione di rischi a cui sono affetti tutti i sistemi esposti a Internet, ai quali malintenzionati potrebbero accedervi (ad es. hacker).

Ci sono stati episodi in cui gli hacker hanno preso il controllo delle auto connesse a Internet e hanno ottenuto l'accesso a telecamere IP che gli hanno consentito di spiare gli utenti nelle proprie case. Molti dispositivi IoT soffrono di vulnerabilità di sicurezza, che potrebbero essere in parte attribuite alla corsa del produttore a fornire dispositivi innovativi senza dare la priorità a test di sicurezza e vulnerabilità adeguati [16] [17] [19].

Alcune delle sfide che esistono, attualmente, con i dispositivi IoT secondo accademici e professionisti del settore [18-24] sono:

- La mancanza di potenza di elaborazione e memoria disponibili nei dispositivi IoT. Gli approcci alla sicurezza che si basano su algoritmi di sicurezza sarebbero quindi vincolati e i dispositivi non sarebbero in grado di eseguire una crittografia complessa;
- L'applicazione di aggiornamenti al dispositivo, comprese le patch di



sicurezza, poiché non tutti i dispositivi supportano gli aggiornamenti over-the-air, richiedendo all'utente di aggiornare tali dispositivi manualmente.

Inoltre, i dispositivi meno recenti potrebbero non ricevere tali aggiornamenti, poiché non più supportati dal produttore.

- Garantire un'elevata disponibilità, poiché le persone si affidano maggiormente all'IoT nella loro vita quotidiana, gli sviluppatori devono assicurarsi che i dati nei dispositivi IoT siano sempre disponibili.
- Il rilevamento e la gestione delle vulnerabilità, attraverso l'utilizzo dei log per identificare se un sistema è stato compromesso e test di vulnerabilità per individuare i difetti nella sicurezza di un sistema.

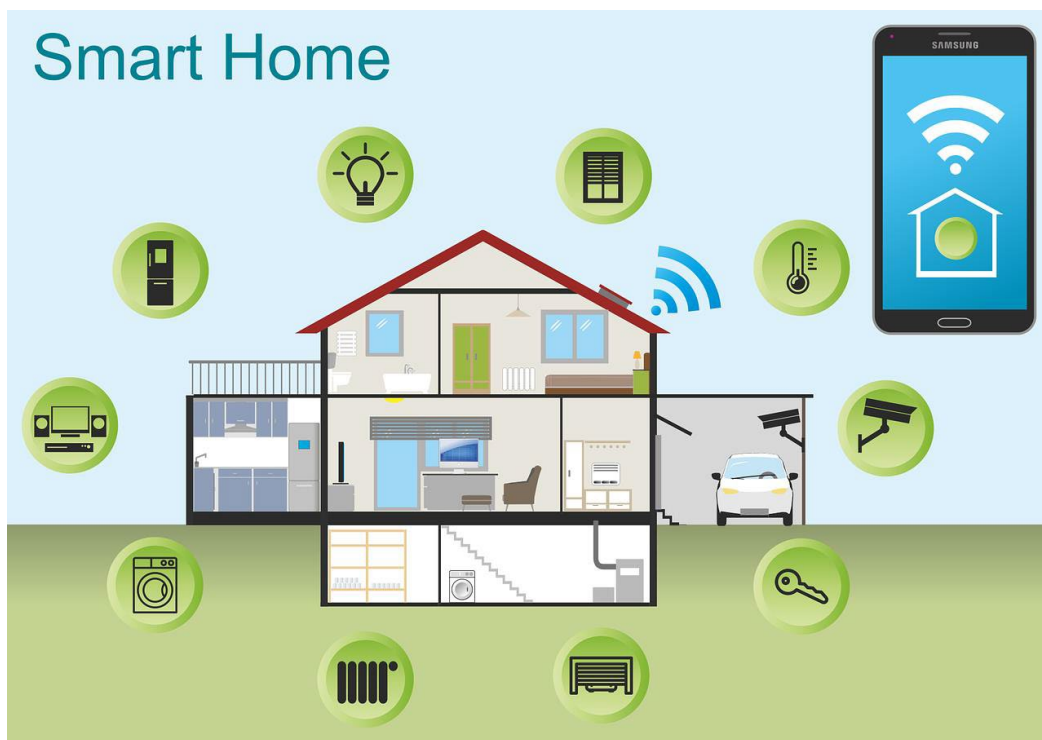
Il focus di questa tesi è sulle vulnerabilità nei dispositivi IoT comuni che si trovano in una smart home, poiché la maggiore dipendenza da tali dispositivi nella nostra quotidianità richiederà dati accessibili ogni volta che ne abbiamo bisogno. Se i dati non fossero disponibili quando una persona ne ha bisogno, le conseguenze potrebbero variare dal fastidio che la console (PS5, XBOX, ...) non funzioni, alla eventualità che il dispositivo dedicato alla gestione del diabete, connesso a Internet, smetta di funzionare [25], con ripercussioni catastrofiche. Queste tematiche saranno affrontate in modo più approfondito nei capitoli seguenti, specificatamente nei capitoli 2 e 3.

## 1.4 Smart Home

La casa intelligente è secondo Bugeja et al. [26] un ambiente in cui dispositivi ed apparecchi elettronici eterogenei sono collegati in rete per fornire servizi intelligenti in modo onnipresente agli individui. Jose et al. [31] spiegano che il concetto di una casa intelligente cambia nel tempo.

Con l'introduzione di Internet, le moderne smart home includono combinazioni di dispositivi IoT. Secondo Ramlee et al. [32] e Vacher et al. [33] anziani e disabili potrebbero anche beneficiare dell'assistenza che una smart home potrebbe fornirgli, controllando i dispositivi collegati ad essa tramite un computer, uno smartphone, un tablet o con la voce.

Rehman et al. [27] spiega che il vantaggio della smart home è il fatto che la casa dell'utente diventi più semplice da “utilizzare”. La semplificazione è data dalla possibilità che l'utente possa controllare le luci, le porte o le telecamere da remoto, ad esempio, al lavoro o durante le vacanze. La principale preoccupazione menzionata da Rehman et al. [27] e da Siboni et al. [28] è la sicurezza e la mancanza di standard di sicurezza. Zhang et al. [29] e Rehman et al. [27] spiegano che i dati inviati sulla rete domestica possono essere personali, privati e sensibili e che alcuni sistemi Smart Home e dispositivi IoT potrebbero non avere una sicurezza adeguata, poiché un hacker non ha strettamente bisogno di essere fisicamente all'interno della casa per controllare il sistema.



**Figura 1.8:** Smart Home con diversi dispositivi connessi, ad esempio lampadine intelligenti, tapparelle intelligenti, termostato intelligente e una telecamera IP [30]

### **1.4.1 Smart Home Scenario**

Lo scenario che viene analizzato in questa tesi è quello di una smart home e dei dispositivi che si possono trovare all'interno di essa.

La maggior parte delle case dispone di una TV, una smart TV o una TV collegata a un dispositivo intelligente che consente lo streaming da Internet, ad esempio un Chromecast o una Apple TV. Il seguente dispositivo, Google Chromecast, è stato selezionato per la sua notevole popolarità, in quanto risulta che ne siano stati venduti 55 milioni in tutto il mondo (notizia del 2017) [34].

Un altro dispositivo selezionato è stato lo smart speaker di Google, in dettaglio il Google Home Mini, vista la crescente popolarità e incredibile diffusione che stanno avendo tali tipi di dispositivi, i quali permettono una comunicazione più diretta e naturale con la smart home e i dispositivi al suo interno.

Inoltre è stata presa in considerazione una telecamera IP, in quanto è uno, fra i modi più semplici, per aumentare la sicurezza in una casa.

Infine è stato analizzato un dispositivo Home Assistant, che ha lo scopo di gestire e unificare l'accesso a tutti i vari dispositivi IoT presenti nella casa, che però non dispongono né di app dedicate né un'interfaccia utente accessibile via web o in un altro modo.

Queste tematiche saranno affrontate in modo più approfondito nei capitoli seguenti, specificatamente nei capitoli 2 e 3, descrivendo in dettaglio i vari dispositivi utilizzati in questa tesi.

## **1.5 Obiettivi di ricerca**

Questa sezione fornisce la motivazione alla base di questa tesi, insieme agli obiettivi e alla domanda di ricerca a cui questa tesi intende rispondere.

### **1.5.1 Motivazione della ricerca**

La motivazione alla base di questa tesi è quella di creare maggiore consapevolezza delle vulnerabilità nella sicurezza che esistono nei dispositivi IoT di tutti i giorni. A causa del grande aumento dei dispositivi IoT previsto dai professionisti del settore [35-37] nei prossimi anni, il problema può diventare sempre più grande con l'aumentare del numero di dispositivi, della loro diversità e dei loro fornitori. Oltre a questo, con più dispositivi automatizzati e connessi a Internet, che vanno dall'utilizzo ricreativo, come una console a dispositivi sanitari come inalatori o pompe automatiche per insulina, è necessario eseguire test di vulnerabilità poiché le conseguenze di un'indisponibilità in alcuni dispositivi potrebbero, secondo Sândescu et al. [39] e Econsultancy [25], in casi estremi portare persino a esiti devastanti, come la perdita di vite umane.

### **1.5.2 Obiettivo della ricerca**

L'obiettivo di questa tesi era mostrare la quantità di vulnerabilità che esistono nei dispositivi IoT di tutti i giorni e quanto siano gravi. L'esperimento si basa in parte su un esperimento condotto da Tekeoğlu et al. [38], dove OpenVAS è stato utilizzato per trovare vulnerabilità nelle telecamere IP.

### **1.5.3 Domanda di ricerca**

Infine, la domanda a cui questa tesi intende rispondere è la seguente:

In che misura i dispositivi IoT in un ambiente domestico sono vulnerabili?

La domanda è rilevante perché la quantità di dispositivi IoT sta aumentando secondo i professionisti del settore [35-37] e poiché la sicurezza non è sempre la massima priorità durante lo sviluppo di questi dispositivi [16] [17] [19].

## **2 Smart Home**

### **2.1 Proprietà della casa intelligente**

L'idea di smart home esiste da almeno 70 anni [42] ed è stata definita svariate volte da diversi autori da allora [43, 44, 45, 46].

Tuttavia, tre aspetti sono quasi sempre presenti tra le definizioni degli ultimi 20 anni.

In primo luogo, i dispositivi presenti nella casa devono essere collegati, principalmente tra loro, ma anche a Internet.

In secondo luogo, ci deve essere un modo “intelligente” per controllare il sistema, come un gateway centrale o un’app per smartphone.

Infine, deve esserci un qualche grado di automazione all'interno del sistema.

Ci sono principalmente tre diversi tipi di dispositivi presenti nelle case intelligenti: sensori, attuatori e dispositivi misti.

Esistono diversi tipi di sensori, ne sono un esempio i termometri, i sensori di luce o i pulsanti/interruttori, i quali forniscono informazioni sull'ambiente del mondo reale nella rete della smart home.

Alcuni esempi di attuatori sono le lampadine, le serrature intelligenti o le macchine da caffè, i quali agiscono in base alle informazioni raccolte dall'ambiente ed eseguono azioni secondo alcune automazioni preimpostate o istruzioni manuali.

Infine, i dispositivi misti sono i più potenti, infatti contengono sia sensori che attuatori, come sistemi di intrattenimento o sistemi di sorveglianza.

Oltre a questo, la maggior parte delle case intelligenti ha un gateway centrale che si collega alla casa e consente ai dispositivi di comunicare tra loro.

I computer e gli smartphone non sono generalmente considerati dispositivi domestici intelligenti, anche se possono interagire con altri dispositivi presenti nella stessa smart home.

## 2.2 Limiti delle risorse

I dispositivi IoT presenti nelle smart home hanno spesso limitazioni nelle loro risorse. Alcuni dispositivi potrebbero essere alimentati a batteria, limitando la loro capacità computazionale. Altri dispositivi hanno limitazioni nella potenza di calcolo (CPU), memoria o larghezza di banda. Queste restrizioni pongono nuove sfide, che non sono presenti per dispositivi con risorse meno limitate, come le workstation e i laptop.

Le principali sfide possono essere suddivise in due categorie: problemi di comunicazione e problemi di calcolo.

Una sfida ai problemi di comunicazione è data dai dispositivi che funzionano a batteria, in quanto necessitano di andare periodicamente offline con lo scopo di risparmiare energia. Di conseguenza, potrebbe esserci la necessità di combinare i dati ricevuti, poiché tali dati vengono trasmessi ad intervalli più ampi.

Questi cambiamenti influiscono sui prerequisiti per la comunicazione, creando anche un bisogno di protocolli di comunicazioni nuovi o aggiornati, in cui la trasmissione avvenga solo durante periodi specifici di tempo con pacchetti concatenati.

D'altro canto le sfide computazionali sono influenzate sia dalle limitazioni della CPU e sia dalla memoria. Le basse risorse della CPU possono rendere impossibili le soluzioni tradizionali, da attuare per il calcolo e la sicurezza [41]. Ad esempio, alcune funzioni crittografiche richiedono una gran quantità di risorse della CPU per essere abbastanza veloci da essere pratiche.

Alcune di queste funzioni non sono attualmente praticabili, per i dispositivi con limitazioni CPU così importanti. Infatti, i dispositivi IoT hanno vincoli come bassa potenza e minore velocità di calcolo e gli algoritmi di crittografia tradizionali come DES, 3DES e AES sono troppo complessi non sembrano utilizzabili per i dispositivi IoT [96].

Infine, come detto in precedenza, molti dei dispositivi IoT sono leggeri e a basso consumo energetico, il che significa che tali dispositivi devono dedicare la potenza di calcolo all'applicazione principale e i metodi di sicurezza tradizionali, che possono quindi richiedere un consumo eccessivo di energia da parte di tale dispositivo per essere eseguiti, siano almeno in parte sacrificati.

Tuttavia, esistono progetti per trovare algoritmi crittografici poco impegnativi per ambienti con risorse limitate, che permettano una comunicazione e una trasmissione dati sicura [47] [96].

## 2.3 Sicurezza e Test di vulnerabilità

La sicurezza IoT è un serio timore nell'era dei big data e dell'intelligenza artificiale.

Il numero crescente di dispositivi intelligenti pone una pressante necessità di comprendere l'IoT e le relative minacce alla sicurezza delle reti, con l'intento di sviluppare contromisure efficaci contro tali attacchi.

Khan et al. [59] affermano che i dispositivi IoT che non vengono aggiornati all'ultima versione del software, possono essere vulnerabili ai rischi per la sicurezza e la privacy.

Infatti, i dati inviati fra i dispositivi collegati alla stessa rete possono essere sensibili, privati o personali.

Rehman et al. [27], Zhang et al. [29] e Dorri et al. [60] spiegano che i dispositivi IoT e i sistemi domestici intelligenti potrebbero non avere una sicurezza adeguata, dando agli hacker la possibilità di prendere il controllo dei dispositivi collegati da remoto, con lo scopo di estrarne i dati o di attaccare tali dispositivi per causarne l'indisponibilità.

Il test di vulnerabilità è secondo Wang et al. [55] il processo di utilizzo di un computer o strumento per cercare punti deboli in un altro dispositivo, computer o rete.

A proposito di Vulnerability Scanner e Vulnerability Testing, Vernotte [61] afferma che, man mano che Internet aumenta di dimensione e diventa più complesso, risulta sempre più difficile proteggere tutte le transazioni che avvengono ogni millisecondo in tutto il mondo.

Doupé et al. [62] affermano che il più grande problema di sicurezza di Internet e del software sono le vulnerabilità delle applicazioni web, che possono essere applicate anche ai dispositivi IoT, i quali utilizzano anch'esse server web e applicazioni per comunicare.

Vernotte [61] afferma che gli scanner di vulnerabilità soffrono di un numero significativo di falsi positivi e falsi negativi e che sia necessario un approccio più strutturato ai test.

Avere un approccio strutturato combinato con un approccio agile ai test di vulnerabilità durante lo sviluppo del software è ripreso anche da Parizi et al. [56] e Sedaghat et al. [57], i quali affermano che gli sviluppatori non possono permettersi di credere che i requisiti di sicurezza iniziali siano perfetti o impenetrabili. Parizi et al. [56] affermano che man mano che il processo di sviluppo continua, il numero di componenti nel sistema generalmente aumenta, insieme alla quantità di possibili vulnerabilità che esistono in quei componenti.

Parizi et al. [56] spiegano come i vari test di vulnerabilità siano un argomento sottovalutato, in quanto integrando continui test di vulnerabilità nello sviluppo di applicazioni software, si

consentirebbe agli sviluppatori di apprendere dagli errori precedenti e impedire che gli stessi errori si propaghino e ripetano successivamente.

Sia Parizi et al. [56] e sia Sedaghat et al. [57] menzionano che la valutazione della sicurezza non può essere eseguita tutta in una volta, ma dovrebbe invece essere eseguita durante il processo di sviluppo. Parizi et al. [56] affermano che molti ingegneri del software non hanno la conoscenza adeguata delle vulnerabilità nella sicurezza, e queste mancanze unite al grande aumento dei vari dispositivi connessi a Internet, ad esempio i dispositivi IoT, potrebbero portare ad una quantità crescente di falle di sicurezza nei dispositivi di tutti i giorni, come Smart TV, telecamere IP o baby monitor.

Un articolo di Bugeja et al. [19] [58] racconta di alcuni hacker che hanno trovato 700 baby monitor collegati a Internet, i quali trasmettevano in streaming bambini addormentati nelle loro culle e 73.000 telecamere IP, che trasmettevano in streaming i filmati di sorveglianza su Internet. Sulla base di ciò, si potrebbe supporre che le telecamere IP siano vulnerabili a diversi tipi di attacchi.

Tekeoğlu et al. [38] menzionano uno strumento di scansione delle vulnerabilità chiamato OpenVAS, con il quale gli autori hanno condotto un esperimento per trovare le vulnerabilità nelle telecamere IP. I risultati dell'esperimento hanno mostrato che la telecamera in questione, fosse vulnerabile agli attacchi Denial of Service.

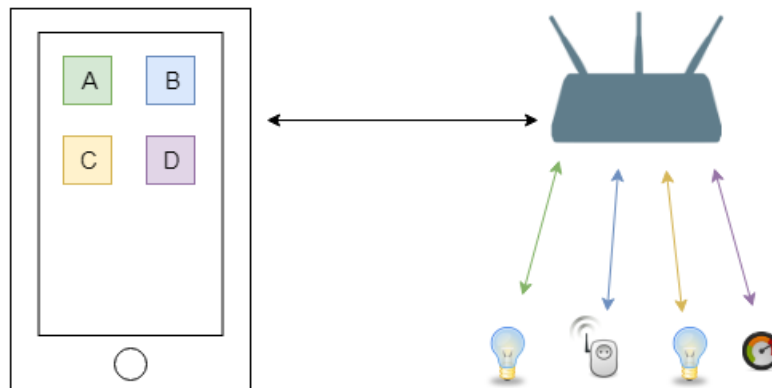
Dopo aver concluso lo sviluppo di un'applicazione, gli strumenti di scansione delle vulnerabilità possono, secondo Sedaghat et al. [57] essere utilizzati per presentare le vulnerabilità esistenti nell'applicazione agli esperti di sicurezza in modo organizzato e dettagliato.



## 2.4 Architetture e configurazioni

Esistono diversi modi per configurare case intelligenti al fine di fornire un'interfaccia per l'utente finale, che sia pratica e facilmente usufruibile.

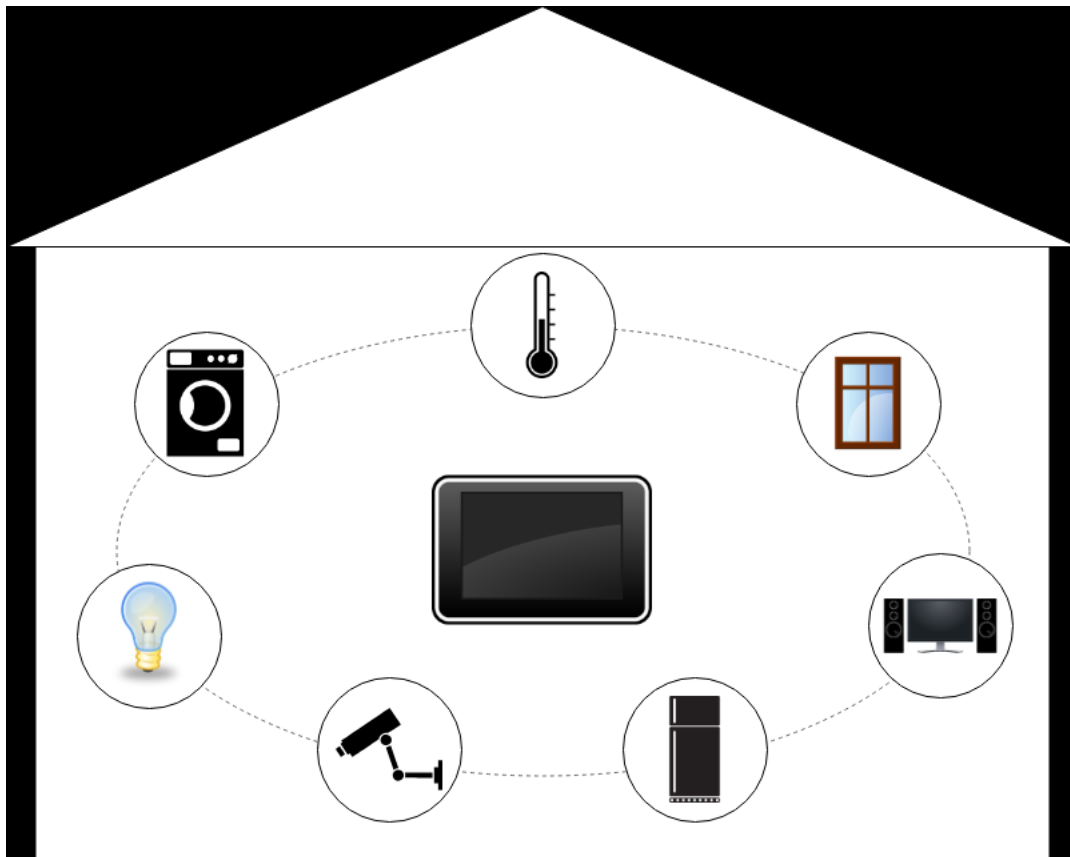
Questa sezione presenta quattro architetture comuni utilizzate per le case intelligenti e ne evidenzia i principali vantaggi e svantaggi.



**Figura 2.1:** Un'architettura basata su applicazioni singole. L'utente necessita di più app per gestire la smart home

I dispositivi consumer, che dispongono di connessione WiFi, sono spesso dotati di una app di accompagnamento in modo da risultare accessibili anche agli utenti non pratici.

I dispositivi intelligenti realizzati dallo stesso produttore sono spesso compatibili tra loro e possono essere controllati dalla stessa applicazione. Tuttavia, se gli utenti desiderano utilizzare dispositivi di produttori diversi, potrebbero finire con utilizzare molte app per controllare tutti i loro diversi dispositivi. Questo diventa facilmente opprimente quando il numero di dispositivi aumenta, come si può vedere nella figura 2.1.



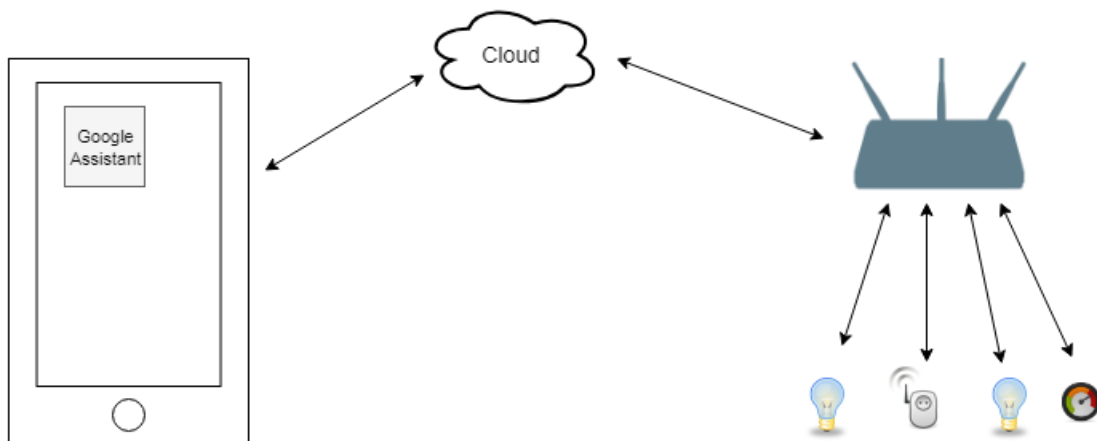
**Figura 2.2:** Un'architettura integrata come quella di Jacobsson et al. [40]

Nelle case moderne a volte ci sono sistemi smart integrati nell'edificio stesso.

Un tale sistema potrebbe, ad esempio, includere componenti per il riscaldamento, impianti di ventilazione e condizionamento. In questi casi l'interfaccia utente è spesso composta da dispositivi fisici situati nell'edificio, come ad esempio uno schermo touch screen o con l'utilizzo di pulsanti e sensori.

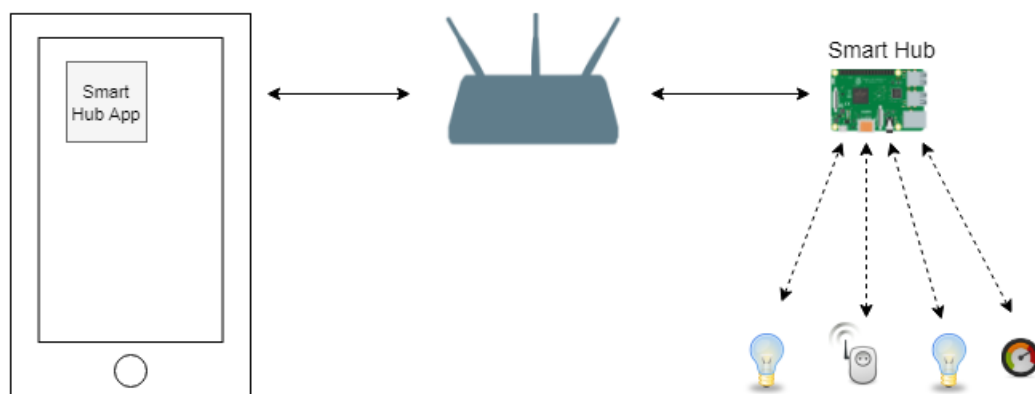
I pannelli di controllo comunicano con un soggetto centrale all'interno dell'edificio che esegue le modifiche richieste. Questi tipi di case o edifici intelligenti hanno una struttura efficiente e di solito sono facili da realizzare, tuttavia possono essere ampliati raramente.

La figura 2.2 illustra una casa intelligente con questa architettura.



**Figura 2.3:** Un'architettura basata su cloud come Google Assistant

Alcuni produttori sul mercato, come Amazon e Google, hanno un sistema di controllo basato sul cloud per le case intelligenti. In queste configurazioni forniscono un'interfaccia, come Amazon Alexa o Google Assistant, che consentono all'utente di connettere i dispositivi supportati all'interfaccia. Ciò consente di ampliare la compatibilità dei dispositivi presenti nella casa, unificando l'accesso a quest'ultimi, anche se prodotti da diversi fornitori, attraverso un'unica interfaccia. Nella maggior parte di questi casi l'utente finale può utilizzare un'app per smartphone per controllare tutti i dispositivi domestici intelligenti. Un esempio di un'architettura cloud domestica intelligente è illustrato nella Figura 2.3.



**Figura 2.4:** Un'architettura basata su hub, come Samsung SmartThings e HomeAssistant

L'architettura finale che viene presentata è un sistema basato su hub, come Samsung SmartThings Hub o Home Assistant. Un hub è un'unità centrale che, come l'architettura basata sul cloud, collega i dispositivi supportati ad un'unica interfaccia.

Un punto di forza rispetto al sistema basato su cloud è che i dispositivi non necessitano di comunicare via Internet. Tuttavia, poiché richiede un hub fisico da installare e configurare, questo è più costoso e potrebbe richiedere molto tempo, soprattutto nella fase iniziale di configurazione, per gli utenti finali.

## 2.5 L'utente

Ad eccezione della tecnologia utilizzata per la smart home per anziani o disabili, ci sono poche caratteristiche distintive degli utenti che le utilizzano, riscontrabili nelle varie ricerche [48].

Invece, ciò che può essere trovato sono caratteristiche dedotte o presunte dei potenziali utenti [48].

Per scoprire chi è l'utente della smart home, è invece necessario esaminare le statistiche e i sondaggi di marketing. Nel 2000, Pragnell et al. [49] hanno identificato le persone interessate alle tecnologie per la smart home, principalmente nei giovani sotto i 35 anni, che sono utilizzatori di dispositivi multimediali e internet nelle loro case, notando anche un livello relativamente alto di reddito. Ciò corrisponde alle statistiche attuali, che mostrano come l'utente più probabile sia un maschio, di età compresa tra i 25 ei 34 anni con un reddito elevato [50]. La diffusione delle tecnologie per le smart home è ancora bassa, con gli Stati Uniti in cima alla lista con un tasso di penetrazione del 32%; in tutto il mondo questo numero è solo del 7,5% [50].

Il gruppo di utenti più distintivo che fa uso di tale tecnologia, da quanto si evince sono gli anziani o disabili, secondo la ricerca condotta da C.Wilson et al. [48]. C'è molto entusiasmo nel capire come la tecnologia impiegata nelle smart home, possa aiutare le persone anziane e disabili a ottenere una vita più sicura, semplice e indipendente.

Ad oggi, la diffusione della tecnologia della casa intelligente tra le persone anziane o disabili è molto bassa. Sebbene molti utenti siano stati soddisfatti della tecnologia che hanno utilizzato, soprattutto nella comunicazione con i loro dottori, c'è molta paura e riluttanza verso le nuove tecnologie da parte di un ampio gruppo del pubblico di destinazione [51]. La maggior parte della paura deriva dalla sfiducia nei confronti della tecnologia, sia nella funzionalità, ma anche nelle intenzioni nascoste di aziende e istituzioni [51, 52].

Ricerche di mercato hanno scoperto che i servizi per la smart home devono avere un'elevata disponibilità ed avere una facilità di utilizzo per risultare attraenti per gli utenti [49, 46]. La ricerca ha presentato una forte connessione, tra gli utenti che non sono stati in grado di svolgere il compito che si erano originariamente prefissati e una fiducia in diminuzione per il sistema. L'interesse dell'utente per un sistema corrisponde direttamente alla fiducia degli utenti per suddetto sistema [46].

Un aspetto interessante nella ricerca riguardante gli utenti delle smart home è che esiste una disconnessione tra ciò che i ricercatori tecnici vedono come le esigenze degli utenti rispetto a

ciò che i ricercatori di scienze sociali vedono come bisogni degli utenti [48]. In effetti, questo divario è così grande che, mentre il reparto tecnico spinge la visione che la tecnologia delle smart home sia la prossima rivoluzione nell'elettrificazione, alcuni ricercatori di scienze sociali si chiedono quale tipo di esigenze tale tecnologia soddisfi effettivamente [48].

Nello sviluppo del prodotto spesso ci sono stati pochissimi contatti tra i potenziali utenti effettivi e gli sviluppatori [53], e lo stesso schema può essere osservato nella ricerca sulle smart home. Questo ha creato una ricerca che spesso non è basata sui veri bisogni degli utenti, ma piuttosto una spinta per la tecnologia sviluppata [54]. Wilson et al. [48] sostengono che la mancanza di attenzione su chi sia l'utente e cosa voglia è un fattore che contribuisce al relativo lento assorbimento della tecnologia per la casa intelligente, che è stato visto storicamente.

### 3 Metodo

Nel seguente capitolo verranno illustrate le varie metodologie utilizzate per raccogliere informazioni sulle vulnerabilità presenti nei dispositivi IoT utilizzati, leggendo articoli scientifici relativi all'argomento specifico. Il motivo alla base di ciò è raccogliere informazioni su quali tipi di studi sono stati effettuati su tale argomento e su come sono stati condotti.

Lo scopo di questo capitolo è spiegare i diversi metodi scientifici utilizzati in questa tesi e il ragionamento alla base del loro utilizzo, spiegando il metodo di revisione dei vari articoli consultati e successivamente analizzando il metodo sperimentale applicato.

Difatti, lo scopo di questa tesi è mostrare quante e quali vulnerabilità esistono nei vari dispositivi presi in considerazione, che possono essere trovati in una smart home, i quali verranno descritti nelle successive sezioni di questo capitolo. Per raggiungere questo obiettivo sono stati studiati diversi articoli per raccogliere informazioni su studi precedenti, sui risultati ottenuti e per capire quali strumenti siano stati utilizzati in questi articoli.

In aggiunta, saranno spiegati i motivi per cui è stata scelta tale piattaforma per effettuare le varie scansioni di vulnerabilità, a dispetto di altre, altrettanto conosciute e popolari.

Inoltre verranno illustrate le varie impostazioni e configurazioni utilizzate per condurre tale esperimento, molte delle quali si basano sugli articoli di ricerca consultati in precedenza.

Per trovare articoli di ricerca che contenessero informazioni pertinenti all'argomento trattato in questa tesi, sono stati consultati i database IEEE [65] e ACM [66]. Questo processo di ricerca e analisi è stato ripetuto per restringere il numero di articoli funzionali a tale argomento.

I risultati sono stati scelti in base alla rilevanza per l'argomento, con particolare attenzione alla data e alla quantità di citazioni dell'articolo o del paper.

### **3.1 Descrizione dell'esperimento**

Le informazioni recuperate dai metodi scientifici sono generalmente di due tipologie, qualitative o quantitative. I dati qualitativi sono, secondo Oates [64, pp. 266], immagini, parole, audio, trascrizioni delle interviste, osservazioni e note, i quali sono difficilmente misurabili. I dati quantitativi sono, sempre secondo Oates [64, p 245], dati basati su numeri generati da esperimenti che vengono poi analizzati utilizzando tabelle o grafici. Oates [64, pp. 127] spiega come un esperimento abbia una strategia che indaga le relazioni di causa ed effetto, mirando a dimostrare o confutare un legame tra un fattore e il risultato ottenuto. Un esperimento è quindi progettato per dimostrare o confutare un'ipotesi osservandone i risultati prodotti. Sulla base di questo, un esperimento è il metodo selezionato per redigere tale tesi, che genera dati quantitativi, poiché la domanda di ricerca richiede dati misurabili.

### **3.2 Discussione sul metodo**

Secondo Oates [64, pp. 72], la ragione alla base di una revisione della letteratura è assicurarsi che l'argomento trattato valga la pena, accertandosi che la ricerca non si limiti a ripetere il lavoro svolto da qualcun altro e che il ricercatore abbia creato nuove conoscenze.

Come accennato in precedenza, i dati raccolti dall'esperimento sono considerati quantitativi secondo Oates [64, pp. 245], che si adatta alla domanda di ricerca a cui questa tesi intende rispondere:

“In che misura i dispositivi IoT in un ambiente domestico sono vulnerabili?”

Per esaminare l'entità e la possibile gravità di una vulnerabilità i dati devono perciò essere misurabili.

Il motivo alla base di un esperimento con lo scanner di vulnerabilità OpenVAS, è in primo luogo perché è stato utilizzato in molti studi precedenti, ed alcuni esempi di questi sono Wang et al. [55], Gordin et al. [63] e Tekeoğlu et al. [38].

Infatti sempre secondo l'articolo di Wang et al. [55], i quali hanno confrontato diversi strumenti, in particolare OpenVAS, Nessus e NMAP con lo scopo di trovare uno scanner di vulnerabilità che potesse essere utilizzato per i corsi sulla sicurezza alla Columbia State University. La conclusione a cui Wang et al. [55] sono giunti è che OpenVAS rappresentasse l'opzione migliore delle tre, poiché il programma era ben progettato e gratuito.



Inoltre anche Gordin et al. [63] elencano tre strumenti di scansione delle vulnerabilità che secondo gli autori offrono i risultati migliori. I tre strumenti menzionati sono per l'appunto OpenVAS, Nessus e Metasploitable e la conclusione a cui sono giunti gli autori è che OpenVAS sia il più completo, infatti i risultati delle scansioni sono ben organizzati e dettagliati, con il notevole vantaggio di essere gratuito rispetto alle controparti.

In secondo luogo, poiché OpenVAS, come riportato precedentemente, mostra i risultati in una forma misurabile, basata su una scala di gravità da 0.0 a 10.0, consentendo così di presentare i risultati in modo chiaro e intuitivo. La gravità misura la criticità di una vulnerabilità, in base a diverse variabili, spiegate nei capitoli successivi.

Per limitare il numero di dispositivi da scansionare viene creato uno scenario, che limita tutti i possibili dispositivi IoT a concentrarsi solo su quelli che generalmente si possono trovare in una smart home.

Difatti, il motivo alla base di questo esperimento è aumentare la conoscenza delle vulnerabilità che potrebbero esistere nei dispositivi IoT presenti nella casa di qualcuno. Infatti, per questa tesi si è deciso di condurre un esperimento su diverse tipologie di dispositivi IoT che si possono trovare in una smart home. Tali dispositivi sono stati scelti poiché si possono trovare nelle maggior parte delle smart home, così da ottenere dei risultati che possano essere rilevanti per un maggior numero di utenti.

Una volta deciso quali dispositivi testare, ho cercato i vari strumenti disponibili su Internet, per condurre un buon esperimento. La maggior parte degli strumenti trovati erano utilizzabili attraverso un abbonamento o una quota d'iscrizione. Lo strumento selezionato, come spiegato in precedenza, è OpenVAS, un software di valutazione delle vulnerabilità, disponibile all'installazione sul sistema operativo Kali Linux. Sia Kali Linux che OpenVAS sono scaricabili gratuitamente, il che consente al lettore di riprodurre lo stesso esperimento condotto in questa tesi, senza dover pagare una quota o un abbonamento.

### 3.3 Dispositivi utilizzati

Di seguito sono riportati tutti i vari dispositivi analizzati in questa tesi, accompagnati da una breve descrizione delle caratteristiche che offrono.

#### **Hassio – Home Assistant**

Home Assistant (Hassio) è un software di automazione domestica gratuito e open source, progettato per essere il sistema di controllo centrale dei vari dispositivi smart domestici con particolare attenzione al controllo locale e alla privacy.

È possibile accedervi sia tramite un'interfaccia utente basata sul Web, sia attraverso l'applicazione ufficiale Home Assistant che è disponibile per Android e iOS, oppure utilizzando i comandi vocali tramite un assistente virtuale come Google Assistant o Amazon Alexa.

Le tecnologie IoT, i dispositivi, i servizi e il software sono supportati grazie alla presenza di componenti specifici, che rendono possibile l'integrazione con protocolli come Bluetooth, MQTT, Zigbee e Z-Wave. Le informazioni provenienti da tali entità possono essere utilizzate all'interno di script o con lo scopo di attivare automazioni create in precedenza, ad esempio per controllare l'illuminazione, il clima, i sistemi di intrattenimento e gli elettrodomestici.

Home Assistant è supportato e può essere installato su molte piattaforme [67]:

- ODROID, Raspberry Pi, Asus Tinkerboard, Intel NUC;
- Sistemi operativi come Windows, macOS, Linux;
- Macchine virtuali e sistemi NAS.

È possibile utilizzare Home Assistant come gateway o bridge per dispositivi che utilizzano diverse tecnologie IoT come Zigbee o Z-Wave, infatti l'hardware necessario può essere montato sui pin GPIO (General Purpose Input/Output) o utilizzando porte USB.

Inoltre, può connettersi direttamente o indirettamente a dispositivi IoT locali, hub, gateway, bridge di controllo o servizi cloud di molti fornitori diversi, inclusi altri ecosistemi domestici intelligenti aperti e chiusi.

Home Assistant ripone la sua attenzione al controllo locale dei dispositivi ai fini della privacy

e complice il suo stato di applicazione open source, è stato descritto come vantaggioso per la sicurezza della piattaforma, in particolare rispetto ai software di automazione closed-source basato su hardware proprietario e servizi cloud.

L'accesso remoto non è abilitato di default e i dati vengono memorizzati esclusivamente sul dispositivo stesso. Gli account utente possono essere protetti con l'autenticazione a due fattori, con lo scopo di scoraggiare e molte volte impedire l'accesso ai malintenzionati, anche se la password dell'utente dovesse essere nota agli aggressori. I componenti aggiuntivi ottengono una valutazione di sicurezza in base al loro accesso alle risorse di sistema.

Nel gennaio 2021, l'analista di sicurezza informatica Oriel Goel ha rilevato una vulnerabilità di sicurezza dell'attraversamento delle directory nelle integrazioni personalizzate di terze parti. Il problema è stato divulgato il 22 gennaio 2021 e finalmente corretto con la versione 2021.1.5 di Home Assistant. Non ci sono informazioni sull'utilizzo di tale vulnerabilità [68]. In questa tesi, Hassio è stato installato in un Raspberry Pi 3 Model B.

## **Google Chromecast (1° generazione)**

Chromecast è un dispositivo prodotto da Google che, una volta collegato al televisore, permette di visualizzare in streaming i contenuti prelevati dalla rete.

L'apparecchio è un potentissimo lettore multimediale, che adotta il sistema operativo Chrome OS e risulta essere grande come una comune chiavetta USB, la quale attraverso la rete Wi-Fi, invia i contenuti in streaming sullo schermo.

Vengono supportate diverse applicazioni: YouTube, Netflix, Google Play Musica, Google Play Film, Plex, Now, RaiPlay, Spotify, DAZN...

Per controllare Chromecast occorre utilizzare un dispositivo esterno come uno smartphone, utilizzando l'applicazione ufficiale Google Home, o un computer tramite l'estensione ufficiale Google Cast.

La compatibilità non è un problema per tale dispositivo, in quanto supporta numerosi sistemi operativi, tra cui Android, iOS, Chrome OS, Windows, Linux e macOS [69].

## **Google Home Mini (1° generazione)**

Il dispositivo Google Home Mini fa parte di una più grande famiglia di altoparlanti intelligenti, comunemente noti come smart speaker, progettati e distribuiti da Google.

Tale apparecchio, attraverso i suoi microfoni, consente agli utenti di pronunciare comandi vocali con l'intento d'interagire con i servizi tramite l'assistente virtuale integrato di Google chiamato Google Assistant.

La particolarità e la forza di tale dispositivo è il gran numero di servizi, sia interni a Google che di terze parti, che sono integrati, consentendo agli utenti di ascoltare musica, controllare la riproduzione di video o foto oppure ricevere aggiornamenti di notizie.

Inoltre, i dispositivi Google Home dispongono di un supporto integrato per l'automazione domestica, consentendo agli utenti di controllare elettrodomestici intelligenti con la propria voce.

Tutte le comunicazioni con i vari dispositivi domotici avvengono sfruttando la rete Wi-Fi ed in alcuni casi via Bluetooth [70].

## **Telecamera IP**

Una telecamera IP è un tipo di videocamera che genera un segnale video in forma digitalizzata e invia tale segnale tramite rete, senza necessità di conversione analogico-digitale. Inoltre è in grado di essere controllata direttamente tramite la rete stessa.

Il campo di utilizzo tipico delle telecamere IP sono i sistemi di videosorveglianza remota su larga scala con raccolta centralizzata delle immagini, di cui costituiscono i terminali [71].

Le telecamere IP si distinguono principalmente in due categorie:

- telecamere IP centralizzate, le quali forniscono esclusivamente il segnale video e non sono in grado di memorizzare localmente le immagini e i video acquisiti, perciò

necessitano di un dispositivo di memorizzazione esterno per la memorizzazione dei dati;

- telecamere IP decentralizzate: sono in grado di memorizzare localmente i video grazie a supporti interni come dischi rigidi o memorie rimovibili.

La telecamera IP presa in considerazione in questa tesi è Foscam FI9821W V2 fa parte dello scenario che viene introdotto.



## 4 Esperimento

Sulla base delle informazioni raccolte dalla revisione della letteratura, sono stati menzionati diversi strumenti per i test di vulnerabilità. Come descritto nel capitolo precedente, sia da Wang et al. [55] che da Gordin et al. [63], i quali hanno scritto articoli che confrontano diversi scanner di vulnerabilità, gli scanner introdotti e confrontati in questi articoli sono OpenVAS, Nessus, NMAP e Metasploitable.

Un altro esempio in cui viene utilizzato OpenVAS, è uno studio condotto da Tekeoğlu et al. [38] che ha utilizzato OpenVAS per scansionare una telecamera IP alla ricerca di vulnerabilità, simile a ciò che questa tesi mira a ottenere.

Sulla base delle informazioni raccolte da questi articoli, insieme al fatto che OpenVAS sia gratuito, disponibile per chiunque e adatto per l'esperimento previsto, si è deciso che questo fosse lo strumento che sarebbe stato utilizzato per condurre l'esperimento in questa tesi.

I dispositivi scansionati in questa tesi, come descritto in precedenza, sono:

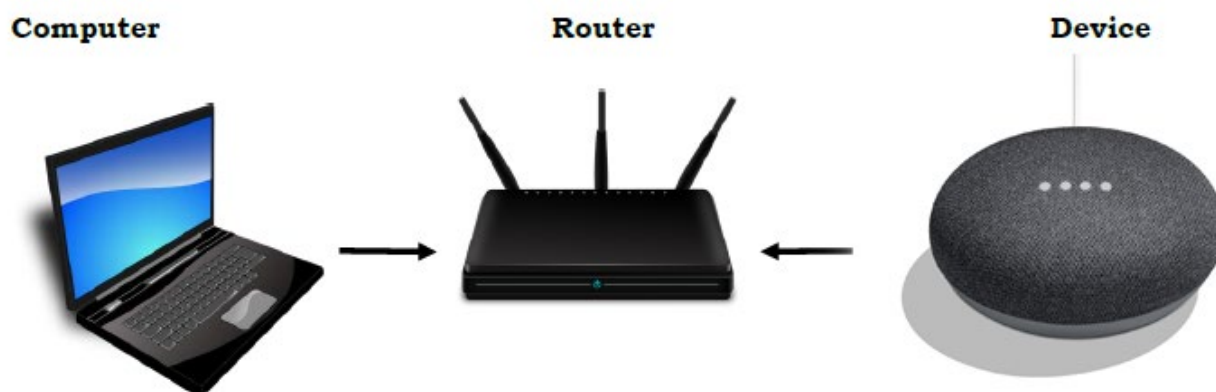
- Hassio (Raspberry Pi 3 Model B);
- Google Home Mini;
- Google Chromecast;
- Telecamera IP Foscam FI9821W V2.

Le versioni del software, l'IP locale e i diversi "ruoli" sono mostrati nella Tabella 4.1. Le scansioni effettuate, per questo esperimento, utilizzano una configurazione di installazione standard compilata da Offensive Security [72].

L'impostazione dell'esperimento si basa sullo studio condotto da Liang et al. [73] e consiste in un computer che funge da host per Kali Linux, in esecuzione su VirtualBox. OpenVAS, installato su Kali Linux che funge da "attaccante", il quale utilizza un router su una rete locale per simulare attacchi sui diversi dispositivi, elencati come vittime nella Tabella 4.1 e come mostrato nella Figura 4.1. Le scansioni sono state eseguite su una rete locale.

Dispositivo	Versione Software/Firmware	Indirizzo IP	Ruolo
PC	Windows 10	192.168.0.5	Server
Kali Linux (VB)	2021.1	192.168.0.5	Attaccante
Router Netgear	1.0.1.60	192.168.0.1	Local Network Gateway
Hassio	2021.5.4	192.168.0.60	Vittima
Chromecast	1.36.159268	192.168.0.31	Vittima
Home Mini	1.54.250118	192.168.0.18	Vittima
Foscam IP Camera	1.4.1.10 (patch 5)	192.168.0.8	Vittima

**Tabella 4.1:** Comprende l'elenco dei dispositivi, la relativa versione del software o del firmware, l'indirizzo IP e il ruolo nello scenario di tale dispositivo (aggressore o vittima)



**Figura 4.1:** Un computer con Kali Linux e OpenVAS installati, che cerca le vulnerabilità in un Google Home Mini su una rete locale

L'idea iniziale dell'esperimento è quella di scoprire le vulnerabilità nei dispositivi IoT, senza concentrarsi su un tipo specifico. Per trovare tali vulnerabilità, vengono completate cinque scansioni OpenVAS su ciascuno dei dispositivi introdotti nello scenario.

Come detto in precedenza, l'esperimento condotto in questa tesi può essere applicato alla maggior parte dei dispositivi IoT su una rete locale.



## 4.1 Strumenti

Gli strumenti software utilizzati per condurre l'esperimento sono i seguenti:

### **VirtualBox**

VirtualBox è un programma sviluppato da Oracle che consente all'utente di installare sistemi operativi aggiuntivi (come applicazioni) su un singolo computer. I sistemi operativi sono chiamati "Sistemi operativi guest (OS)" e consentono l'uso di sistemi operativi come Linux o Mac OS su un computer con Windows installato come sistema operativo principale o viceversa. Il programma è stato rilasciato nel gennaio 2007 e l'ultima versione, al momento della stesura di questa tesi, è la 6.1.22, rilasciata il 29 aprile 2021, che è per l'appunto la versione utilizzata in questo esperimento [74].

### **Kali Linux**

Kali Linux è un sistema operativo che consente agli utenti di condurre test di sicurezza e vulnerabilità. Kali viene fornito con centinaia di strumenti che si concentrano su Penetration Testing, Ricerca sulla sicurezza, Computer Forensics e Reverse Engineering. Il sistema operativo è stato rilasciato nel 2013 e l'ultima versione è la 2021.2, che è stata rilasciata il 1° giugno 2021. Tale versione è quella utilizzata in questo esperimento. Il progetto Kali Linux è stato creato e continua ad essere gestito da Offensive Security, che è il leader di settore nella formazione e certificazione online dei test di penetrazione per i professionisti della sicurezza delle informazioni. Kali Linux è gratuito e disponibile per chiunque [75-77].

### **OpenVAS**

OpenVAS è uno scanner di vulnerabilità completo e gratuito che consente all'utente di scansare le possibili vulnerabilità presenti nei vari dispositivi, specificandone i vari indirizzi IP e le relative porte. OpenVAS è sviluppato da Greenbone Networks; tale applicativo è stato rilasciato nel 2009 e l'ultima versione GVM-21.04, è stata rilasciata il 16 aprile 2021, ed è quella che viene utilizzata in questa tesi [78].

## 4.2 Impostazioni

Le impostazioni utilizzate in questo esperimento si basano sulla guida di Kali Linux. La guida mostra un processo passo passo su come impostare e utilizzare OpenVAS [79, 80].

### Configurazione di OpenVAS

L'elenco delle porte utilizzato nelle scansioni è *All TCP* e *Nmap 5.51 top 100 UDP* e cerca in tutte le 65535 porte TCP (Transmission Control Protocol), insieme alle 100 porte UDP (User Datagram Protocol) più utilizzate [81].

La configurazione della scansione utilizzata per ogni scansione è la modalità *Full e Fast* [82] che, in base all'elenco delle porte, ricerca attraverso ciascuna porta con diversi Network Vulnerability Tests (NVT). Ogni NVT appartiene a una “famiglia”, ad esempio “Attacchi di forza bruta”, “Buffer overflow”, “Denial of Service”, ...; In totale, la modalità *Full e Fast* contiene 62 famiglie con 49653 NVT combinati.

Lo scanner è accompagnato da un feed di test di vulnerabilità con una lunga cronologia e aggiornamenti quotidiani. Questo feed della community di Greenbone include più di 80.000 test di vulnerabilità.

## 5 Risultati

Nel seguente capitolo verranno spiegati i diversi termini seguiti dai risultati delle scansioni di ciascun dispositivo.

Le scansioni delle vulnerabilità verranno eseguite più volte per ogni dispositivo, questo per assicurarsi che i risultati siano coerenti e non solo un evento casuale.



Una volta completata la scansione, ne verranno elencate le vulnerabilità, se sono state rilevate, con informazioni di base su quale tipo di vulnerabilità, tipo di soluzione, gravità, qualità del rilevamento, host e in quale porta sono state trovate tali vulnerabilità.




### **Vulnerability:**

Visualizza il nome della vulnerabilità rilevata nell'indirizzo IP specifico (host). Se l'utente fa click sulla vulnerabilità, quest'ultima fornirà ulteriori informazioni su di essa e sulle soluzioni esistenti, se disponibili.

### **Solution Type:**

Questa colonna mostra un'icona che rappresenta una possibile soluzione, se disponibile; ci sono cinque diverse icone basate su altrettante soluzioni. Le icone daranno un suggerimento sui tipi di soluzione.

-  Workaround: sono disponibili informazioni su una configurazione o uno scenario di distribuzione specifico che può essere utilizzato per evitare l'esposizione alla vulnerabilità. Ci possono essere nessuna, una o più soluzioni alternative disponibili. Questa è di solito la “prima linea di difesa” contro una nuova vulnerabilità prima che venga emessa o addirittura scoperta una mitigation o un vendor fix del fornitore.
-  Mitigation: sono disponibili informazioni su una configurazione o uno scenario di distribuzione che consentono di ridurre il rischio della vulnerabilità, ma che non risolvono la vulnerabilità del prodotto interessato. Le mitigation possono includere l'utilizzo di dispositivi o controlli di accesso esterni al prodotto interessato. Inoltre, le mitigation possono essere emesse dall'autore originale di tale dispositivo e possono essere ufficialmente consigliate dal produttore di tale dispositivo.

-  Vendor fix: sono disponibili informazioni su una correzione ufficiale emessa dall'autore originale del prodotto interessato. Se non diversamente specificato, si presume che questa correzione risolva completamente la vulnerabilità.
-  No fix available: attualmente non è disponibile alcuna correzione. Le informazioni dovrebbero contenere dettagli sul motivo per cui non esiste una correzione.
-  Will not fix: non esiste una correzione per la vulnerabilità e non ce ne sarà mai una. Questo è spesso il caso in cui un prodotto non viene più mantenuto o comunque deprecato. Le informazioni dovrebbero contenere dettagli sul motivo per cui non verrà emessa alcuna correzione [83] [84].

### Gravità:

Mostra un valore su una scala da 0.0 a 10.0, in base alla criticità della vulnerabilità, calcolata dal calcolatore OpenVAS CVSS [85] [86].

Il calcolatore considera i fattori del vettore di accesso, che è classificato in tre diversi tipi, locale, adiacente e di rete, la complessità di accesso che assume valori decrescenti, da alto a basso. Autenticazione, che può essere multipla, singola o nessuna. La riservatezza, ovvero l'accesso alle informazioni, varia da nessuna a completa. L'integrità è l'accesso alla modifica che può essere nessuno, parziale o completo. La disponibilità è l'accessibilità delle risorse può essere nessuna, parziale o completa. Tali fattori sono descritti nella tabella 5.1.

Fattori	Valori		
Access Vector (Vettore di accesso)	Local, l'attaccante deve avere un accesso fisico al sistema	Adjacent network, l'attaccante deve avere accesso al dominio di collisione o alla trasmissione	Network, l'attaccante può sfruttare la vulnerabilità da remoto
Access Complexity (Complessità di accesso)	Alta, condizioni di accesso specializzate. Ad esempio, l'attaccante deve avere privilegi elevati	Media, esistono condizioni di accesso; l'attaccante è limitato a un certo livello di autorizzazione	Bassa, le condizioni di accesso non esistono

Authentication (Autenticazione)	Multipla, l'attaccante deve autenticarsi due o più volte	Singola, richiede all'aggressore di accedere al sistema	Nessuna, autenticazione non richiesta
Confidentiality (Riservatezza)	Nessuna, la vulnerabilità sfruttata non influirà sulla riservatezza	Bassa, accesso per leggere alcuni file, ma non è possibile selezionare quali	Alta, l'attaccante ha pieno accesso in lettura a tutti i file di sistema
Integrity (Integrità)	Nessuna, la vulnerabilità sfruttata non influirà sull'integrità	Bassa, l'attaccante può modificare i file di sistema, ma non può selezionare quali	Alta, l'attaccante può modificare tutti i file di sistema
Availability (Disponibilità)	Nessuna, la vulnerabilità sfruttata non influirà sulla disponibilità	Bassa, l'attaccante può ridurre le prestazioni per la disponibilità	Alta, l'attaccante può negare l'accesso a tutte le risorse

**Tabella 5.1:** Una breve spiegazione di ciascun fattore e del valore corrispondente, che impattano sulla gravità della vulnerabilità

La gravità è il punteggio di base calcolato e assegnato alla vulnerabilità, che è calcolato dalla sfruttabilità (*exploitability*) di tale vulnerabilità e dal suo impatto.

Per calcolare *l'exploitability*, i valori di Access Vector, Access Complexity e Authentication vengono moltiplicati per 20:

$$Exploitability = 20 * Access\ Vector * Access\ Complexity * Authentication.$$

Il valore *Impact* è calcolato dai valori di Riservatezza, Integrità e Disponibilità moltiplicati per 10,41:

$$Impact = 10,41 * (1 - (1 - Confidentiality) * (1 - Integrity) * (1 - Availability)).$$

Il punteggio di base viene quindi calcolato nel modo seguente:

$$Base\ Score = ((0,6 * impact) + (0,4 * Exploitability) - 1,5) * f(impact).$$

La funzione  $f(impact)$  è 0 se Confidentiality, Integrity e Availability sono impostate sul valore "Nessuna". In caso contrario, se una delle voci Confidentiality, Integrity o Availability è "Bassa" o "Alta", il valore assegnato è un valore costante di 1,176 [86].

Ad esempio, se Confidentiality, Integrity e Availability sono tutte impostate sul valore "Nessuna", il punteggio di gravità sarà 0,0, anche se i valori di Vettore di accesso, Complessità di accesso e Autenticazione mostrano che il dispositivo è vulnerabile agli

attacchi. Tuttavia, se Confidentiality, Integrity e Availability sono impostate su Bassa o Alta, il punteggio di gravità aumenterà.

La tabella 5.2 mostra tre diversi esempi di vulnerabilità con diversi livelli di gravità in base ai calcoli sopra menzionati.

Calcolatore CVSS	Esempio 1	Esempio 2	Esempio 3
Access Vector	Local (0.395)	Adjacent network (0.646)	Network (1.0)
Access Complexity	Bassa (0.71)	Alta (0.35)	Media (0.61)
Authentication	Multipla (0.45)	Nessuna (0.704)	Singola (0.56)
Confidentiality	Nessuna (0.0)	Bassa (0.275)	Alta (0.66)
Integrity	Bassa (0.275)	Alta (0.66)	Alta (0.66)
Availability	Alta (0.66)	Alta (0.66)	Alta (0.66)
Risultato	5.0 (Media)	6.5 (Media)	8.5 (Alta)

**Tabella 5.2:** Vengono rappresentati tre esempi con valori diversi di ciascuna categoria utilizzando il calcolatore CVSS. Ogni esempio mostra anche i valori costanti, tra parentesi, del valore selezionato in ciascuna categoria, che sono stati utilizzati per calcolare il punteggio di base

Il livello di gravità si basa sul Common Vulnerability Scoring System (CVSS) [87]. In OpenVAS le valutazioni sono impostate su Basso (0,0 – 3,9), Medio (4,0 – 6,9) e Alto (7,0 – 10,0) [88].

#### **Quality of Detection (QoD):**

La qualità del rilevamento (QoD) è un valore compreso tra 0 % e 100 %, che descrive l'affidabilità del rilevamento della vulnerabilità.

In OpenVAS, esiste un valore minimo predefinito impostato al 70%, utilizzato per filtrare i risultati visualizzati nei report.

**Host:**

Un host è un singolo sistema connesso a una rete e che può essere scansionato. Uno o più host costituiscono la base di destinazione di una scansione.

Gli host nelle destinazioni di una scansione e nei rapporti di scansione sono identificati dal loro indirizzo di rete, un indirizzo IP o un nome host [89].

**Location:**

Visualizza quale porta e se la vulnerabilità è stata rilevata su una porta TCP o UDP [90].

**Summary:**

Informazioni sulla vulnerabilità secondo OpenVAS.

**Solution:**

La soluzione che OpenVAS suggerisce per tale vulnerabilità.

Le pagine seguenti includono i risultati delle scansioni di vulnerabilità condotte sui dispositivi, nel seguente ordine:

1. Hassio
2. Google Home Mini (Prima generazione)
3. Google Chromecast (Prima generazione)
4. Telecamera IP Foscam FI9821W V2

## **5.1 Hassio**

### **Risultati Hassio**

I risultati hanno mostrato che il dispositivo Home Assistant non aveva alcuna vulnerabilità secondo tale esperimento.

### **Osservazioni su Hassio**

Sulla base delle informazioni raccolte, i risultati hanno mostrato che Hassio era al sicuro dai vari test delle vulnerabilità effettuati. Ciò potrebbe indicare che il dispositivo è sicuro o che gli strumenti utilizzati per condurre questo esperimento non sono stati in grado di rilevare le vulnerabilità presenti nel dispositivo.



## 5.2 Google Home Mini (Prima generazione)

### Risultati Google Home Mini

Vulnerability	Severity	QoD	Host IP	Name	Location	Created
SSL/TLS: Deprecated TLSv1.0 and TLSv1.1 Protocol Detection	4.3 (Medium)	98 %	192.168.0.18		9000/tcp	Mon, May 3, 2021 10:34 AM UTC
SSL/TLS: Deprecated TLSv1.0 and TLSv1.1 Protocol Detection	4.3 (Medium)	98 %	192.168.0.18		8443/tcp	Mon, May 3, 2021 10:34 AM UTC
SSL/TLS: Deprecated TLSv1.0 and TLSv1.1 Protocol Detection	4.3 (Medium)	98 %	192.168.0.18		8009/tcp	Mon, May 3, 2021 10:34 AM UTC
SSL/TLS: Deprecated TLSv1.0 and TLSv1.1 Protocol Detection	4.3 (Medium)	98 %	192.168.0.18		10101/tcp	Mon, May 3, 2021 10:34 AM UTC
SSL/TLS: Deprecated TLSv1.0 and TLSv1.1 Protocol Detection	4.3 (Medium)	98 %	192.168.0.18		10001/tcp	Mon, May 3, 2021 10:34 AM UTC

**Figura 5.1:** I risultati della ricerca All TCP top 100 UDP su Google Home Mini dopo che cinque test sono stati completati

Vulnerabilità	Severity	Summary	Solution
SSL/TLS: Deprecated TLSv1.0 and TLSv1.1 Protocol Detection	4.3	È stato rilevato l'utilizzo del protocollo TLSv1.0 e/o TLSv1.1, entrambi deprecati su questo sistema	Si consiglia di disabilitare i protocolli TLSv1.0 e/o TLSv1.1 deprecati a favore dei protocolli TLSv1.2+ (Mitigation)

**Tabella 5.3:** Visualizzazione delle vulnerabilità e della gravità delle vulnerabilità rilevate nel dispositivo Google Home Mini con un riepilogo e una soluzione suggerita

### Osservazioni su Home Mini

Le scansioni dello smart speaker Home Mini hanno dato un solo risultato in totale, su cinque porte differenti, con un valore di gravità pari a 4.3, come mostrato nella Figura 5.1 e nella Tabella 5.3.

La vulnerabilità ha rivelato che lo smart speaker utilizza i protocolli TLSv1.0 e TLSv1.1, i quali sono stati deprecati da Google a marzo 2020.

La soluzione menzionata in OpenVAS è quella di disabilitare tali protocolli, adottando quelli più recenti, a partire dalla versione 1.2.

## 5.3 Google Chromecast (Prima generazione)

### Risultati Google Chromecast

Vulnerability		Severity ▼	QoD	Host		Location	Created
				IP	Name		
SSL/TLS: Deprecated TLSv1.0 and TLSv1.1 Protocol Detection	↩	4.3 (Medium)	98 %	192.168.0.31		9000/tcp	Mon, May 3, 2021 10:23 AM UTC
SSL/TLS: Deprecated TLSv1.0 and TLSv1.1 Protocol Detection	↩	4.3 (Medium)	98 %	192.168.0.31		8443/tcp	Mon, May 3, 2021 10:23 AM UTC
SSL/TLS: Deprecated TLSv1.0 and TLSv1.1 Protocol Detection	↩	4.3 (Medium)	98 %	192.168.0.31		8009/tcp	Mon, May 3, 2021 10:23 AM UTC
SSL/TLS: Certificate Signed Using A Weak Signature Algorithm	↩	4.0 (Medium)	80 %	192.168.0.31		9000/tcp	Mon, May 3, 2021 10:23 AM UTC

**Figura 5.2:** I risultati della ricerca All TCP top 100 UDP su Google Chromecast dopo che cinque test sono stati completati

Vulnerabilità	Severity	Summary	Solution
SSL/TLS: Deprecated TLSv1.0 and TLSv1.1 Protocol Detection	4.3	È stato rilevato l'utilizzo del protocollo TLSv1.0 e/o TLSv1.1, entrambi deprecati su questo sistema	Si consiglia di disabilitare i protocolli TLSv1.0 e/o TLSv1.1 deprecati a favore dei protocolli TLSv1.2+ (Mitigation)
SSL/TLS: Certificate Signed Using A Weak Signature Algorithm	4.0	Il servizio remoto utilizza un certificato SSL/TLS nella catena di certificati che è stato firmato utilizzando un algoritmo di hashing crittograficamente debole	I server che utilizzano certificati SSL/TLS firmati con un debole algoritmo di hashing SHA-1, MD5, MD4 o MD2 dovranno ottenere nuovi certificati SSL/TLS firmati SHA-2 per evitare gli avvisi di certificato SSL/TLS dei vari browser web (Mitigation)

**Tabella 5.4:** Visualizzazione delle vulnerabilità e della gravità delle vulnerabilità rilevate nel dispositivo Google Chromecast con un riepilogo e una soluzione suggerita

## **Osservazioni su Chromecast**
















Le scansioni del dispositivo Chromecast hanno dato due risultati in totale, su tre porte differenti, con un valore di gravità, rispettivamente pari a 4.3 e 4.0, come mostrato nella Figura 5.2 e nella Tabella 5.4.

La prima vulnerabilità ha rivelato che tale dispositivo utilizza gli stessi protocolli visti in precedenza nel Google Home Mini, TLSv1.0 e TLSv1.1, i quali sono stati deprecati da Google a marzo 2020. Come in precedenza, la soluzione menzionata in OpenVAS è quella di disabilitare tali protocolli, adottando quelli più recenti, a partire dalla versione 1.2.

La seconda vulnerabilità ha rivelato che Chromecast utilizza l'algoritmo di hashing obsoleto SHA-1 rilasciato nel 1995 ed è vulnerabile agli attacchi dal 2005. La soluzione menzionata per tale vulnerabilità in OpenVAS consiste nell'ottenere nuovi certificati SSL/TLS firmati con l'algoritmo di hashing SHA-2. Inoltre, sia Microsoft e Google hanno avvertito gli utenti sin dal 2017, che i siti Web che utilizzano SHA-1 non sono più sicuri, secondo quanto riportato da PCWorld [91][92].

## 5.4 Telecamera IP Foscam

### Risultati Telecamera IP

Vulnerability		Severity ▼	QoD	Host IP	Name	Location	Created
<a href="#">Lighttpd Multiple vulnerabilities</a>		7.5 (High)	99 %	<a href="#">192.168.0.8</a>		443/tcp	Sat, May 8, 2021 8:14 AM UTC
<a href="#">Lighttpd Multiple vulnerabilities</a>		7.5 (High)	99 %	<a href="#">192.168.0.8</a>		88/tcp	Sat, May 8, 2021 8:14 AM UTC
<a href="#">SSL/TLS: OpenSSL CCS Man in the Middle Security Bypass Vulnerability</a>		5.8 (Medium)	70 %	<a href="#">192.168.0.8</a>		443/tcp	Sat, May 8, 2021 7:56 AM UTC
<a href="#">SSL/TLS: Certificate Expired</a>		5.0 (Medium)	99 %	<a href="#">192.168.0.8</a>		443/tcp	Sat, May 8, 2021 7:39 AM UTC
<a href="#">SSL/TLS: Report Weak Cipher Suites</a>		5.0 (Medium)	98 %	<a href="#">192.168.0.8</a>		443/tcp	Sat, May 8, 2021 7:39 AM UTC
<a href="#">SSL/TLS: Report Vulnerable Cipher Suites for HTTPS</a>		5.0 (Medium)	98 %	<a href="#">192.168.0.8</a>		443/tcp	Sat, May 8, 2021 7:39 AM UTC
<a href="#">SSL/TLS: SSLv3 Protocol CBC Cipher Suites Information Disclosure Vulnerability (POODLE)</a>		4.3 (Medium)	80 %	<a href="#">192.168.0.8</a>		443/tcp	Sat, May 8, 2021 7:39 AM UTC
<a href="#">SSL/TLS: Deprecated TLSv1.0 and TLSv1.1 Protocol Detection</a>		4.3 (Medium)	98 %	<a href="#">192.168.0.8</a>		443/tcp	Sat, May 8, 2021 7:39 AM UTC
<a href="#">jQuery &lt; 1.9.0 XSS Vulnerability</a>		4.3 (Medium)	80 %	<a href="#">192.168.0.8</a>		443/tcp	Sat, May 8, 2021 7:48 AM UTC
<a href="#">jQuery &lt; 1.6.3 XSS Vulnerability</a>		4.3 (Medium)	80 %	<a href="#">192.168.0.8</a>		88/tcp	Sat, May 8, 2021 7:48 AM UTC
<a href="#">jQuery &lt; 1.6.3 XSS Vulnerability</a>		4.3 (Medium)	80 %	<a href="#">192.168.0.8</a>		443/tcp	Sat, May 8, 2021 7:48 AM UTC
<a href="#">jQuery &lt; 1.9.0 XSS Vulnerability</a>		4.3 (Medium)	80 %	<a href="#">192.168.0.8</a>		88/tcp	Sat, May 8, 2021 7:48 AM UTC
<a href="#">SSL/TLS: Deprecated SSLv2 and SSLv3 Protocol Detection</a>		4.3 (Medium)	98 %	<a href="#">192.168.0.8</a>		443/tcp	Sat, May 8, 2021 7:39 AM UTC
<a href="#">SSL/TLS: Diffie-Hellman Key Exchange Insufficient DH Group Strength Vulnerability</a>		4.0 (Medium)	80 %	<a href="#">192.168.0.8</a>		443/tcp	Sat, May 8, 2021 7:39 AM UTC

**Figura 5.3:** I risultati della ricerca All TCP top 100 UDP sulla Telecamera IP dopo che cinque test sono stati completati

I risultati raccolti dalla scansione delle vulnerabilità della Telecamera IP, hanno mostrato che c'erano diverse vulnerabilità, come mostrato nella Figura 5.4 e nella Tabella 5.5.

Vulnerabilità	Severity	Summary	Solution
Lighttpd Multiple vulnerabilities	7.5	Questo host esegue Lighttpd ed è soggetto a molteplici vulnerabilità	Aggiornare alla versione 1.4.35 o successiva (Vendor Fix)
SSL/TLS: OpenSSL CCS Man in the Middle Security Bypass Vulnerability	5.8	OpenSSL è soggetto a vulnerabilità di bypass della sicurezza	Aggiornare all'ultima versione (Vendor Fix)
SSL/TLS: Certificate Expired	5.0	Il certificato SSL/TLS del server remoto è scaduto	Sostituire il certificato SSL/TLS con uno nuovo (Mitigation)

SSL/TLS: Report Weak Cipher Suites	5.0	Questa routine riporta tutte le suite di crittografia SSL/TLS deboli accettate dal servizio.  NOTA: non viene segnalata alcuna gravità per i servizi SMTP con "TLS opportunistico" e suite di crittografia deboli sulla porta 25/tcp. Se vengono configurate suite di crittografia troppo forti per questo servizio, l'alternativa sarebbe quella di ricorrere a una comunicazione in chiaro ancora più insicura	La configurazione di questo servizio dovrebbe essere cambiata in modo che non accetti le suite di cifratura più deboli (Mitigation)
SSL/TLS: Report Vulnerable Cipher Suites for HTTPS	5.0	Questa routine segnala tutte le suite di crittografia SSL/TLS accettate da un servizio in cui i vettori di attacco esistono solo sui servizi HTTPS	La configurazione di questi servizi dovrebbe essere modificata in modo che non accetti più le suite di crittografia più deboli (Mitigation)
SSL/TLS: SSLv3 Protocol CBC Cipher Suites Information Disclosure Vulnerability (POODLE)	4.3	Questo host è soggetto a una vulnerabilità legata alla divulgazione di informazioni	Possibili Mitigation sono: - Disabilitare SSLv3 - Disabilitare le suite di crittografia che supportano le modalità di crittografia CBC - Abilitare TLS_FALLBACK_SCSV se il servizio fornisce TLSv1.0+
SSL/TLS: Deprecated TLSv1.0 and TLSv1.1 Protocol Detection	4.3	È stato rilevato l'utilizzo del protocollo TLSv1.0 e/o TLSv1.1, entrambi deprecati su questo sistema	Si consiglia di disabilitare i protocolli TLSv1.0 e/o TLSv1.1 deprecati a favore dei protocolli TLSv1.2+ (Mitigation)

jQuery < 1.9.0 XSS Vulnerability	4.3	jQuery prima della versione 1.9.0 è vulnerabile agli attacchi Cross-site Scripting (XSS). La funzione jQuery(strInput) non differenzia i selettori dall'HTML in modo affidabile. Nelle versioni vulnerabili, jQuery ha determinato se l'input era HTML cercando il carattere '<' in qualsiasi punto della stringa, offrendo agli aggressori maggiore flessibilità quando tentano di costruire un payload dannoso. Nelle versioni fisse, jQuery considera l'input HTML solo se inizia esplicitamente con il carattere '<', limitando l'exploit solo agli aggressori che possono controllare l'inizio di una stringa, che è molto meno comune	Aggiornare alla versione 1.9.0 o successiva (Vendor Fix)
jQuery < 1.6.3 XSS Vulnerability	4.3	Vulnerabilità di scripting cross-site (XSS) in jQuery prima della versione 1.6.3, quando si utilizza location.hash per selezionare elementi, consente agli aggressori remoti di inserire script Web o HTML arbitrari tramite un tag predisposto	Aggiornare alla versione 1.6.3 o successiva o applica la patch (Vendor Fix)
SSL/TLS: Deprecated SSLv2 and SSLv3 Protocol Detection	4.3	È stato possibile rilevare l'utilizzo del protocollo SSLv2 e/o SSLv3 deprecato su questo sistema	Si consiglia di disabilitare i protocolli SSLv2 e/o SSLv3 deprecati a favore dei protocolli TLSv1+ (Mitigation)
SSL/TLS: Diffie-Hellman Key	4.0	Il servizio SSL/TLS utilizza gruppi Diffie-Hellman con forza	Distribuire (Ephemeral) Elliptic-Curve Diffie-

Exchange Insu-cient DH Group Strength Vulnerability		insufficiente (dimensione della chiave < 2048)	Hellman (ECDHE) o utilizzare un gruppo Diffie-Hellman a 2048 bit o più potente. Per i server Web Apache, a partire dalla versione 2.4.7, mod_ssl utilizzerà parametri DH che includono numeri primi con lunghezze superiori a 1024 bit (Workaround)
--	--	---	---

**Tabella 5.5:** Visualizzazione delle vulnerabilità e della gravità delle vulnerabilità rilevate nel dispositivo Telecamera IP con un riepilogo e una soluzione suggerita

### Osservazioni sulla Telecamera IP

Rispetto agli altri dispositivi testati, i risultati ottenuti dalla Telecamera IP sono stati quelli che si sono distinti di più. Infatti, tali risultati hanno mostrato che la quantità di vulnerabilità (11) è la più alta di ogni dispositivo testato. Inoltre, la vulnerabilità peggiore (7.5) è stata quella più grave rilevata in ogni dispositivo testato.

La vulnerabilità con gravità 7.5, Lighttpd Multiple vulnerabilities, è nota principalmente poiché consentirà agli aggressori remoti di eseguire comandi SQL arbitrari e leggere file arbitrari tramite hostname. L'unica soluzione consiste nell'aggiornare il web server Lighttpd alla versione 1.4.35 o successive.

La vulnerabilità con gravità 5.8, SSL/TLS: OpenSSL CCS Man in the Middle Security Bypass Vulnerability, può consentire agli aggressori di ottenere informazioni sensibili conducendo un attacco man-in-the-middle. Inoltre, tale attacco può portare ad altri attacchi. L'unica possibile soluzione è controllare la presenza di aggiornamenti che correggano tale vulnerabilità.

La vulnerabilità con gravità 5.0, SSL/TLS: Certificate Expired, invece può essere risolta rimpiazzando il certificato SSL/TLS con uno nuovo.

Le vulnerabilità con gravità 5.0, SSL/TLS: Report Weak Cipher Suites e SSL/TLS: Report Vulnerable Cipher Suites for HTTPS, possono essere aggirate cambiando la configurazione di questi servizi, in modo che non accettino più le suite di crittografia deboli elencate nel report (RC4, 1024 bit RSA, ...).

La vulnerabilità con gravità 4.3, SSL/TLS: SSLv3 Protocol CBC Cipher Suites Information Disclosure Vulnerability (POODLE), consentirà a un aggressore (man-in-the-middle) di ottenere l'accesso al flusso di dati in testo normale. Le possibili soluzioni a tale vulnerabilità, possono essere viste nella tabella precedente.

La vulnerabilità con gravità 4.3, SSL/TLS: Deprecated TLSv1.0 and TLSv1.1 Protocol Detection, ha rivelato che tale dispositivo utilizza gli stessi protocolli visti in precedenza nel Google Home Mini e nel Google Chromecast, TLSv1.0 e TLSv1.1, i quali sono stati deprecati da Google a marzo 2020. Come in precedenza, la soluzione menzionata in OpenVAS è quella di disabilitare tali protocolli, adottando quelli più recenti, a partire dalla versione 1.2.

Le vulnerabilità con gravità 4.3, jQuery < 1.9.0 XSS Vulnerability e jQuery < 1.6.3 XSS Vulnerability, vengono ben descritte nella tabella precedente e come unica soluzione proposta c'è quella di aggiornare a una versione più recente.

La vulnerabilità con gravità 4.3, SSL/TLS: Deprecated SSLv2 and SSLv3 Protocol Detection, potrebbe consentire ad un utente malintenzionato di utilizzare i difetti crittografici noti, per intercettare la connessione tra i client e il servizio per ottenere l'accesso ai dati sensibili trasferiti all'interno della connessione protetta. La soluzione proposta è quella di disabilita SSLv2 e SSLv3 in favore dei protocolli TLSv1+.

La vulnerabilità con gravità 4.0, SSL/TLS: Diffie-Hellman Key Exchange Insufficient DH Group Strength Vulnerability, consentirebbe ad un utente malintenzionato di decrittografare la comunicazione SSL/TLS offline. La soluzione proposta è descritta in dettaglio nella tabella precedente.

Molte delle soluzioni di vulnerabilità suggeriscono un aggiornamento del firmware, ma tutti i test sono stati effettuati sulla versione firmware 1.4.1.10 (patch 5) della Telecamera IP (l'ultima versione disponibile al momento della scansione).



## 5.6 Riepilogo dei risultati

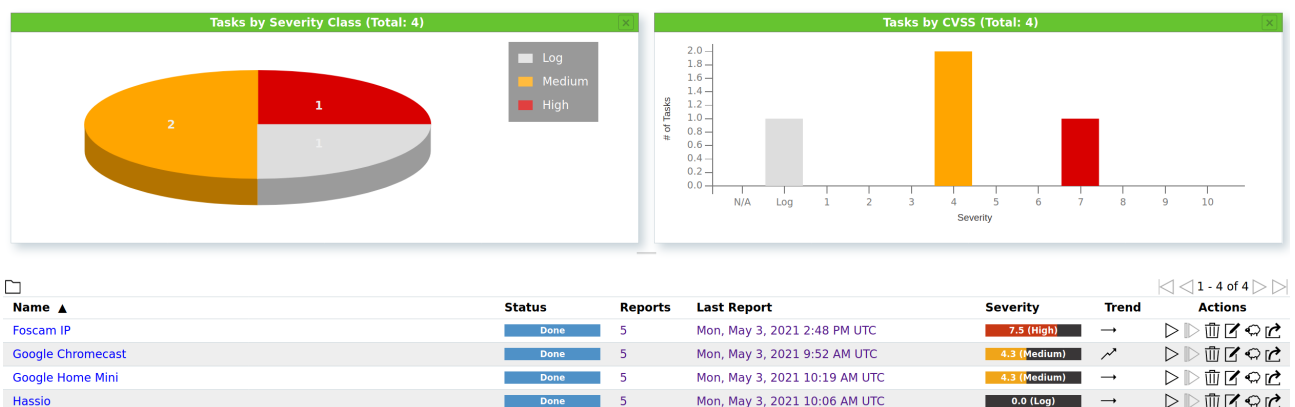
Tutti i dispositivi testati, escluso Home Assistant, hanno dimostrato di essere suscettibili a diversi tipi di vulnerabilità, come mostrato nelle Figure 5.4.

I risultati delle 20 scansioni effettuate hanno mostrato che la telecamera IP Foscam, possedeva il maggior numero di vulnerabilità e la vulnerabilità con la gravità più alta (7.5).

Lo smart speaker Google Home Mini e Google Chromecast presentano la seconda vulnerabilità più grande (4.3).

Entrambi i dispositivi hanno mostrato di avere lo stesso tipo di vulnerabilità, ossia la deprecazione dei protocolli TLSv1.0 e/o TLSv1.1. Inoltre, i risultati del dispositivo Google Chromecast hanno mostrato che in aggiunta alla vulnerabilità precedente, ne esistesse un'altra causata dall'utilizzo da parte del dispositivo di un algoritmo di hashing obsoleto.

Le scansioni su Hassio non hanno rilevato alcuna vulnerabilità, il che potrebbe essere dovuto alla mancanza di vulnerabilità o al fatto che OpenVAS non è riuscito a trovarle.



**Figure 5.4:** OpenVAS. I grafici riportati in alto, mostrano il numero di vulnerabilità scoperte, ordinate in categorie; log (le vulnerabilità trovate non erano rilevanti; ad esempio, test per rilevare quale sistema operativo è in esecuzione in un dispositivo), medio (tra 4.0 e 6.9) e alto (tra 7.0 e 10.0). L'elenco in basso mostra i diversi dispositivi che sono stati sottoposti alla scansione delle vulnerabilità, il loro stato, la data della scansione e la vulnerabilità più grave rilevata



## 6 Analisi e discussione

Lo scopo di questo capitolo è analizzare e discutere i risultati raccolti dall'esperimento e, in alcuni casi, trovare soluzioni ad alcune delle vulnerabilità scoperte.

### 6.1 Analisi

L'analisi sarà strutturata nel modo seguente, in primo luogo verranno esaminate e valutate le vulnerabilità più gravi dei singoli dispositivi. Successivamente, verrà fatto un riepilogo dei risultati ottenuti da tutti i dispositivi e delle conseguenze che possono avere tali vulnerabilità.

#### **Hassio**

Non sono state rilevate vulnerabilità durante l'utilizzo di OpenVAS, ma ciò non significa necessariamente che il dispositivo sia sicuro. Alcuni degli altri strumenti menzionati in questa tesi, ad esempio Nessus o NMAP, possono essere utilizzati per scansionare il dispositivo invece di utilizzare solo OpenVAS. Utilizzando altri strumenti, potrebbero essere individuate delle vulnerabilità.

#### **Google Home Mini**

La vulnerabilità rilevata nello smart speaker consiste nell'utilizzo dei protocolli TLSv1.0 e TLSv1.1, i quali sono stati deprecati da Google a marzo 2020.

Come riportato nel capitolo precedente, la soluzione menzionata in OpenVAS è quella di disabilitare tali protocolli, adottando quelli più recenti, a partire dalla versione 1.2.

La vulnerabilità si trovava sulle porte TCP 9000, 8443, 8009, 10101, 10001, ma non ci sono articoli con ulteriori informazioni sulle varie porte, né è stata trovata una soluzione per chiudere tale porta per consentire ulteriori test.

#### **Google Chromecast**

Come appena descritto nel dispositivo Home Mini, Google Chromecast presenta lo stesso tipo di vulnerabilità e la stessa possibile soluzione. L'unica differenza sta nelle porte in cui è stata trovata e cioè sulle porte TCP 9000, 8443 e 8009.

L'altra vulnerabilità rilevata nel Chromecast è un algoritmo di firma obsoleto (SHA-1), che è stato vulnerabile all'hacking dal 2005. La soluzione suggerita a questa vulnerabilità è l'aggiornamento a un algoritmo di sicurezza più recente, ad esempio SHA-2.

La vulnerabilità è stata trovata sulla porta TCP 9000, ma non ci sono articoli con ulteriori informazioni sulla porta specifica, né è stata trovata una soluzione per chiudere tale porta per consentire ulteriori test.

### **Telecamera IP Foscam**

Come analizzato ampiamente nel capitolo precedente, le vulnerabilità riscontrate richiedono di aggiornare uno o più componenti (web server, certificati, protocolli, ...) di tale dispositivo. La particolarità di tali suggerimenti, necessari per risolvere tali vulnerabilità, sta nel fatto che tale dispositivo è già stato aggiornato alla versione più recente. Si nota però che tale dispositivo è stato rilasciato nel mercato nell'anno 2013 ed ha ricevuto aggiornamenti software fino a luglio del 2019. Questo denota l'importanza di avere un supporto duraturo, garantito dagli aggiornamenti della casa produttrice, al momento di scegliere un dispositivo oppure un altro. Infatti molte delle vulnerabilità sarebbero "facilmente" risolvibili aggiornando i vari componenti, in quanto molti dei componenti hardware supporterebbero tali versioni aggiornate.

## 6.2 Discussione

Questa tesi si concentra sulle varie vulnerabilità presenti nei vari dispositivi IoT che si possono trovare in una smart home. La rilevanza di tale tesi è data dall'adozione di tali dispositivi IoT, i quali stanno diventando sempre più comuni e sempre più persone si affidano a loro ogni giorno. A causa di questa dipendenza, la stabilità e l'integrità dei dispositivi diventerà sempre più importante, poiché le conseguenze di un'interruzione temporanea potranno portare a rischi per la sicurezza e, in casi estremi, rischi per la salute [17][25]. Sulla base di studi precedenti condotti da Wang et al. [55], Gordin et al. [63], Tekeolu et al. [38] e utilizzando le impostazioni impiegate da Liang et al. [73], in combinazione con le informazioni raccolte dalla revisione della letteratura, le quali indicavano che OpenVAS fosse uno strumento adatto per esaminare le vulnerabilità anche se ci fossero altre opzioni, come Nessus, Nmap o Metasploitable [55][63], le scansioni sono state ripetute più volte, per assicurarsi che i risultati potessero essere riprodotti ed è diventato chiaro che le varie scansioni, a volte, potessero generare risultati diversi. Per questo motivo vengono visualizzati tutti i risultati ottenuti dalle cinque scansioni per ogni dispositivo.

OpenVAS, che è stato utilizzato per eseguire la scansione delle vulnerabilità, è un software gratuito disponibile per il download da parte di chiunque, il che potrebbe indicare che le scansioni effettuate non siano altrettanto valide rispetto alle alternative a pagamento, come Nessus. Ma anche con questo software gratuito, i risultati hanno mostrato chiaramente che 3 dispositivi su 4 presentavano varie vulnerabilità, soprattutto certificati e protocolli deprecati od obsoleti.

Sembra esserci una preoccupazione generale da parte di professionisti e studiosi del settore, i quali indicano che la sicurezza nei dispositivi IoT non è così alta come dovrebbe essere, come riportato dai professionisti del settore [17] e studiosi come Rauscher et al. [94]. Ma, come è evidente dal grande aumento della quantità di dispositivi IoT, si può sostenere che i consumatori non sembrano pensare che la mancanza di sicurezza sia un problema, almeno nell'immediato, così grande come fanno notare gli esperti, e questo lo si evince dall'aumento annuale dei dispositivi connessi a Internet [35-37].

Con la sicurezza limitata che alcuni dispositivi IoT hanno dimostrato di avere, combinata con il grande aumento di quest'ultimi, potrebbe portare a una quantità crescente di problemi. Se il problema diventasse abbastanza grande e le persone iniziassero a rendersi conto, che alcuni dei loro dispositivi potrebbero non essere sicuri come pensano, il mercato IoT e Smart Home

potrebbe subire un duro colpo, poiché le persone diventerebbero più caute nell'acquisto di dispositivi che potrebbero compromettere la loro privacy e sicurezza.

L'esperimento condotto in questa tesi ha testato quattro dispositivi che possono essere trovati realisticamente nelle case di tutto il mondo, scansionati utilizzando un software gratuito disponibile per chiunque. In tre dispositivi su quattro sono state rilevate vulnerabilità che possono consentire a un utente malintenzionato di compromettere tali dispositivi. Ad esempio, le vulnerabilità riscontrate nella telecamera IP Foscam, consentirebbero ad un hacker di disabilitare temporaneamente il dispositivo. Se la telecamera venisse utilizzata per scopi di sicurezza, i rischi e le conseguenze potrebbero essere gravi, se un malintenzionato fosse in grado di disabilitare tale telecamera [38] [19].

Il fatto che queste vulnerabilità esistano nei dispositivi prodotti e rilasciati da alcune delle più grandi aziende del mondo dovrebbe far riflettere. Infatti, tali risultati fanno sorgere molte preoccupazioni, ma che dire dei dispositivi creati da aziende più piccole con meno capitali e risorse? È più probabile che i dispositivi rilasciati da aziende più piccole siano più o meno sicuri?

L'aumento dei dispositivi, in corso da diversi anni può essere in parte attribuito alle aziende più grandi che rilasciano più prodotti, ma allo stesso modo un numero enorme di questi dispositivi viene rilasciato da aziende più piccole, le quali cercano di trovare il loro posto nel mercato IoT e delle smart home.

Le vulnerabilità che sono state scoperte da OpenVAS in questa tesi avrebbero potuto essere facilmente trovate dai produttori durante il processo di sviluppo, ma sembra che non sia così, in quanto non sono state corrette e alcune non verranno affatto corrette secondo i fornitori, come mostrato nella Tabella 5.5. È importante rimarcare che tali vulnerabilità non dovrebbero esistere in un prodotto rilasciato al pubblico, soprattutto da grandi aziende.

Le scansioni di vulnerabilità condotte in questa tesi potrebbero essere facilmente eseguite da chiunque voglia testare i propri dispositivi su una rete locale, e future aggiunte a tale campo di studio potrebbero essere effettuate utilizzando altri dispositivi o altri strumenti, per vedere se i risultati ottenuti siano simili a quelli presentati in questa tesi.

Una delle soluzioni che riduce significativamente, la quantità di vulnerabilità è mantenere sempre aggiornati i dispositivi IoT all'ultima versione del firmware e del software, che è ripresa da Khan et al. [59]. Questo fatto è stato evidente nel dispositivo Home Assistant, in quanto fosse affetto da una vulnerabilità (directory traversal) nelle integrazioni di terze parti, scoperta dall'analista di sicurezza informatica Oriel Goel. Tale vulnerabilità consiste nello sfruttare un'insufficiente validazione di sicurezza o una scarsa sanificazione dell'input di nomi di file forniti dall'utente, come i caratteri speciali utilizzati per "raggiungere la root

directory” che vengono passati alle API dei file a livello web server. L’obiettivo dell’attacco è utilizzare un’applicazione specifica per poter guadagnare, in modo non autorizzato, i privilegi di accesso al file system. Con tale accesso è possibile raggiungere dei dati al di fuori della root directory, come il file shadow delle password o altri file contenenti dati importanti e sensibili [68, 93]. Con il rilascio della versione 2021.1.5, disponibile dal 23 gennaio 2021, tale vulnerabilità è stata risolta.

Infine, in linea generale, si nota che i dispositivi open source che riscuotono un gran successo fra gli utenti, sono quelli che presentano meno vulnerabilità, in quanto anche persone esterne a tali progetti possono contribuire alla scoperta di vulnerabilità e alla loro risoluzione (open contribution), basti vedere il caso di Home Assistant appena citato.





## 7 Conclusioni e lavoro futuro

Lo scopo di questa tesi era quello di presentare le vulnerabilità che si possono riscontrare nei dispositivi IoT presenti nelle smart home.

Dopo che l'esperimento è stato condotto e i risultati sono stati esaminati, era chiaro che ci fossero delle vulnerabilità legate all'utilizzo di protocolli e certificati oramai deprecati o scaduti in tre dei quattro dispositivi testati. I dispositivi che sono stati scansionati utilizzavano la versione più recente del software o del firmware al momento in cui è stata condotta la scansione iniziale (maggio 2021), il che indica che le vulnerabilità esistono attualmente nei dispositivi che potrebbero essere trovati nelle case (intelligenti) di molte persone. Con il grande aumento dei dispositivi IoT [35-37], questo potrebbe diventare un problema sempre più grande, se la sicurezza continuerà ad essere un elemento secondario nello sviluppo dei dispositivi.

Tutti i dispositivi che sono stati testati in questa tesi non sono la versione hardware più recente, che potrebbe essere considerata una situazione realistica, in base a quali dispositivi sono effettivamente presenti nelle case delle persone, poiché non tutti acquisteranno la versione più recente di Chromecast o Home Mini rilasciate a cadenza annuale.

Per ricavare più informazioni dai risultati ottenuti in questa tesi, si potrebbero fare le seguenti aggiunte:

- Testare la versione più recente dei dispositivi, ad esempio Google Nest Mini o il più recente Chromecast (Google Chromecast con Google TV);
- Testare altri dispositivi utilizzando metodi simili;
- Testare dispositivi simili utilizzando strumenti diversi, come Nessus, Nmap o Metasploitable, che potrebbero presentare risultati diversi;
- Ogni dispositivo può anche essere testato eseguendo attacchi veri e propri (Denial of Service, ...) per confermare che le vulnerabilità esistano nella pratica;

In questa tesi è stato utilizzato il feed della comunità Greenbone, che è la versione gratuita di OpenVAS. OpenVAS offre anche una versione a pagamento chiamata Greenbone Security Feed. La differenza tra la versione a pagamento e quella non a pagamento è che lo scanner della versione a pagamento, include più test di vulnerabilità della rete che sono specificatamente destinati agli ambienti aziendali [95]. Utilizzando la versione a pagamento, è

possibile trovare più vulnerabilità che potrebbero essere un qualcosa da considerare in un lavoro futuro.

# Bibliografia

- [1] Beebom, Rajesh Mishra, “15 Examples of Internet of Things Technology in Use Today”, Gennaio 2020. Online: <https://beebom.com/examples-of-internet-of-things-technology/>
- [2] IoT for all, Calum McClelland, “What is IoT? - A Simple Explanation of the Internet of Things”, Febbraio 2021. Online: <https://www.iotforall.com/what-is-internet-of-things>
- [3] O. Vermesan and P. Friess, “Building the Hyperconnected Society”, 2015. Online: [http://www.internet-of-things-research.eu/pdf/Building\\_the\\_Hyperconnected\\_Society\\_IERC\\_2015\\_Cluster\\_eBook\\_978-87-93237-98-8\\_P\\_Web.pdf](http://www.internet-of-things-research.eu/pdf/Building_the_Hyperconnected_Society_IERC_2015_Cluster_eBook_978-87-93237-98-8_P_Web.pdf)
- [4] Wikipedia, Kevin Ashton, 2021. Online: [https://en.wikipedia.org/wiki/Kevin\\_Ashton](https://en.wikipedia.org/wiki/Kevin_Ashton)
- [5] L. R. LLC, “An Introduction to the Internet of Things (IoT)”, Novembre 2013. Online: [https://www.cisco.com/c/dam/en\\_us/solutions/trends/iot/introduction\\_to\\_IoT\\_november.pdf](https://www.cisco.com/c/dam/en_us/solutions/trends/iot/introduction_to_IoT_november.pdf)
- [6] K. Ashton, “That ‘Internet of Things’ Thing”, Giugno 2009. Online: <https://www.rfidjournal.com/that-internet-of-things-thing>
- [7] Postscapes, “Internet of Things (IoT) History” Dicembre 2019. Online: <https://www.postscapes.com/iot-history/>
- [8] Google, “Google trends, IoT”, Giugno 2021. Online: <https://trends.google.com/trends/explore?date=all&geo=IT&q=internet%20of%20things&hl=en-US>
- [9] O. Vermesan, P. Friess, and et al., “Internet of Things - IoT Governance, Privacy and Security Issues”, Gennaio 2015. Online: [http://www.internet-of-things-research.eu/pdf/IERC\\_Position\\_Paper\\_IoT\\_Governance\\_Privacy\\_Security\\_Final.pdf](http://www.internet-of-things-research.eu/pdf/IERC_Position_Paper_IoT_Governance_Privacy_Security_Final.pdf)

[10] Forbes, L. Columbus, “Roundup Of Internet Of Things Forecasts And Market Estimates, 2016”. Online: <https://www.forbes.com/sites/louiscolombus/2016/11/27/roundup-of-internet-of-things-forecasts-and-market-estimates-2016/##2b7dc31292d5>

[11] ZDNet, L. Dignan, “IoT devices to generate 79.4ZB of data in 2025, says IDC”, 2019. Online: <https://www.zdnet.com/article/iot-devices-to-generate-79-4zb-of-data-in-2025-says-idc/>

[12] IBM, J. R. Anna Gerber, “Connecting all the things in the Internet of Things”, Gennaio 2020. Online: <https://developer.ibm.com/technologies/iot/articles/iot-lp101-connectivity-network-protocols/>

[13] S. Aguzzi, D. Bradshaw, M. Canning, M. Cansfield, P. Carter, G. Cattaneo. Sergio Gusmeroli, G. Micheletti, D. Rotondi e R. Stevens, “Definition of a Research and Innovation Policy Leveraging Cloud Computing and IoT Combination”, 2013. Online: [2013ec.europa.eu/newsroom/dae/document.cfm?doc\\_id=9472](http://2013ec.europa.eu/newsroom/dae/document.cfm?doc_id=9472)

[14] MarketsandMarkets, “Internet of Things (IoT) Market by Software Solution (Real-Time Streaming Analytics, Security Solution, Data Management, Remote Monitoring, and Network Bandwidth Management), Service, Platform, Application Area, and Region - Global Forecast to 2022”. Online: <https://www.marketsandmarkets.com/Market-Reports/internet-of-things-market-573.html>

[15] Cisco, “Cisco Annual Internet Report (2018–2023) White Paper”, 2020. Online: <https://www.cisco.com/c/en/us/solutions/collateral/executive-perspectives/annual-internet-report/white-paper-c11-741490.html>

[16] Bugeja Joseph, Vogel Bahtijar, Jacobsson Andreas, and Varshney Rimpu. “IoTSM: An End-to-end Security Model for IoT Ecosystems”, 2019 IEEE International Conference on Pervasive Computing and Communications Workshops (PerCom Workshops). IEEE, 2019, pp. 267-272.

[17] A. Banafa, “Three Major Challenges Facing IoT”, IEEE, March 2017. Online: <https://iot.ieee.org/newsletter/march-2017/three-major-challenges-facing-iot.html>

- [18] S. Singh and N. Singh, "Internet of Things (IoT): Security challenges, business opportunities & reference architecture for E-commerce", 2015 International Conference on Green Computing and Internet of Things (ICGCIoT), 2015, pp. 1577-1581.
- [19] J. Bugeja, D. Jönsson and A. Jacobsson, "An Investigation of Vulnerabilities in Smart Connected Cameras", 2018 IEEE International Conference on Pervasive Computing and Communications Workshops (PerCom Workshops), 2018, pp. 537-542.
- [20] A. Gerber, "Top 10 IoT security challenges", IBM, Marzo 2020. Online: <https://developer.ibm.com/articles/iot-top-10-iot-security-challenges/>
- [21] CMSWiRE, D. Roe, "7 Big Problems with the Internet of Things", Aprile 2021. Online: <https://www.cmswire.com/cms/internet-of-things/7-big-problems-with-the-internet-of-things-024571.php>
- [22] C. Chen, Z. Zhang, S. Lee and S. Shieh, "Penetration Testing in the IoT Age" in Computer, vol. 51, no. 4, 2018, pp. 82-85.
- [23] G. Baldini, A. Skarmeta, E. Fournieret, R. Neisse, B. Legeard and F. Le Gall, "Security certification and labelling in Internet of Things", 2016 IEEE 3rd World Forum on Internet of Things (WF-IoT), 2016, pp. 627-632.
- [24] A. Boudguiga et al., "Towards Better Availability and Accountability for IoT Updates by Means of a Blockchain," 2017 IEEE European Symposium on Security and Privacy Workshops (EuroS&PW), 2017, pp. 50-58.
- [25] Econsultancy, "10 examples of the Internet of Things in healthcare", Econsultancy, Gennaio 2021. Online: <https://econsultancy.com/internet-of-things-healthcare/>
- [26] Bugeja, Joseph, Andreas Jacobsson, and Paul Davidsson. "Smart Connected Homes" Internet of Things A to Z: Technologies and Applications. Wiley, 2018, pp. 359-384.
- [27] S. ur Rehman and V. Gruhn, "An approach to secure smart homes in cyber-physical systems/Internet-of-Things", 2018 Fifth International Conference on Software Defined Systems (SDS), 2018, pp. 126-129.

- [28] S. Siboni et al., “Security Testbed for Internet-of-Things Devices”, in IEEE Transactions on Reliability, vol. 68, no. 1, March 2019. pp. 23-44.
- [29] Z. Zhang, M. C. Y. Cho and S. Shieh, “Emerging Security Threats and Countermeasures in IoT”, 2015, In Proceedings of the 10th ACM Symposium on Information, Computer and Communications Security (ASIA CCS '15). 2015, pp.1-6.
- [30] Pixabay, Pixaline, “Smart Home”. Online: <https://pixabay.com/illustrations/smart-home-house-technology-2005993/>
- [31] A. C. Jose and R. Malekian, “Improving Smart Home Security: Integrating Logical Sensing Into Smart Home”, in IEEE Sensors Journal, vol. 17, no. 13, pp. 4269-4286, 2017.
- [32] R. A. Ramlee, M. A. Othman, M. H. Leong, M. M. Ismail and S. S. S. Ranjit, “Smart home system using android application”, 2013 International Conference of Information and Communication Technology (ICoICT), 2013, pp. 277-280.
- [33] M. Vacher et al., “The sweet-home project: Audio technology in smart homes to improve well-being and reliance”, 2011 Annual International Conference of the IEEE Engineering in Medicine and Biology Society, 2011, pp. 5291-5294.
- [34] CNET, T. Collins, “Google has sold 55 million Chromecast devices”, 2017. Online: <https://www.cnet.com/news/google-has-sold-55-million-chromecast-and-chromecast-built-in-devices/>
- [35] Gartner Inc., “Gartner Says 8.4 Billion Connected "Things" Will Be in Use in 2017, Up 31 Percent From 2016”, 2017. Online: <https://www.gartner.com/en/newsroom/press-releases/2017-02-07-gartner-says-8-billion-connected-things-will-be-in-use-in-2017-up-31-percent-from-2016>
- [36] Ericsson, ” IoT connections outlook”. Online: <https://www.ericsson.com/en/mobility-report/dataforecasts/iot-connections-outlook>

- [37] Forbes, L. Columbus, “2017 Roundup Of Internet Of Things Forecasts”, Dicembre 2017. Online: <https://www.forbes.com/sites/louiscolumbus/2017/12/10/2017-roundup-of-internet-of-things-forecasts/?sh=258a67251480>
- [38] A. Tekeoğlu and A. S. Tosun, “Investigating Security and Privacy of a Cloud-Based Wireless IP Camera: NetCam”, 2015 24th International Conference on Computer Communication and Networks (ICCCN), 2015, pp. 1-6.
- [39] C. Săndescu, O. Grigorescu, R. Rughiniş, R. Deaconescu and M. Calin, “Why IoT security is failing. The Need of a Test-Driven Security Approach”, 2018 17th RoEduNet Conference: Networking in Education and Research (RoEduNet), 2018, pp. 1-6.
- [40] A. Jacobsson, M. Boldt, and B. Carlsson, “A risk analysis of a smart home automation system,” Future Generation Computer Systems, vol. 56, no. Supplement C, pp. 719 – 733, 2016.
- [41] M. M. Hossain, M. Fotouhi, and R. Hasan, “Towards an analysis of security issues, challenges, and open problems in the internet of things,” in 2015 IEEE World Congress on Services, Giugno 2015, pp. 21–28.
- [42] D. Sung, “Smart home visions through the ages: The history of home automation”, Marzo 2021. Online: <https://www.the-ambient.com/features/visions-through-the-ages-history-of-home-automation-178>
- [43] N. King, “Smart home - a definition,” Settembre 2003. Online: [https://www.housinglin.org.uk/\\_assets/Resources/Housing/Housing\\_advice/Smart\\_Home\\_-\\_A\\_definition\\_September\\_2003.pdf](https://www.housinglin.org.uk/_assets/Resources/Housing/Housing_advice/Smart_Home_-_A_definition_September_2003.pdf)
- [44] SmartHomes, “What is a smart home”. Online: <https://smartofficesandsmarthomes.com/smarthomes/>
- [45] R. J. Robles and T.-h. Kim, “Applications, systems and methods in smart home technology: A review,” Int. Journal of Advanced Science And Technology, vol. 15, 2010.

- [46] G. Sandström, Smart Homes and User Values Long-term evaluation of ITservices in Residential and Single Family Dwellings. Stockholm: KTH, 2009.
- [47] K. A. McKay, L. Bassham, M. S. Turan, and N. Mouha, Report on lightweight cryptography. US Department of Commerce, National Institute of Standards and Technology, 2017.
- [48] C. Wilson, T. Hargreaves, and R. Hauxwell-Baldwin, “Smart homes and their users: a systematic analysis and key challenges”, *Personal and Ubiquitous Computing*, vol. 19, no. 2, pp. 463–476, 2015.
- [49] M. Pragnell, L. Spence, and R. Moore, The market potential for Smart Homes. YPS for the Joseph Rowntree Foundation York, Novembre 2000.
- [50] STATISTA, “Smart home”. Online: <https://www.statista.com/outlook/dmo/smart-home/worldwide>
- [51] J. F. Coughlin, L. A. D’Ambrosio, B. Reimer, and M. R. Pratt, “Older adult perceptions of smart home technologies: implications for research, policy & market innovations in healthcare”, in *Engineering in Medicine and Biology Society, 2007. EMBS 2007. 29th Annual International Conference of the IEEE. IEEE, 2007*, pp. 1810–1815.
- [52] A. McLean, “Ethical frontiers of ICT and older users: cultural, pragmatic and ethical issues”, *Ethics and information technology*, vol. 13, no. 4, pp. 313–326, 2011.
- [53] H. Rohrer, “The role of users in the social shaping of environmental technologies”, *Innovation: the european journal of social science research*, vol. 16, no. 2, pp. 177–192, 2003.
- [54] C. Tweed and G. Quigley, “The design and technological feasibility of home systems for the elderly”, The Queens University, Belfast, 2000.
- [55] Y. Wang and J. Yang, “Ethical Hacking and Network Defense: Choose Your Best Network Vulnerability Scanning Tool”, 2017 31st International Conference on Advanced Information Networking and Applications Workshops (WAINA), pp. 110-113.



- [56] R. M. Parizi, K. Qian, H. Shahriar, F. Wu and L. Tao, "Benchmark Requirements for Assessing Software Security Vulnerability Testing Tools", 2018 IEEE 42nd Annual Computer Software and Applications Conference (COMPSAC), pp. 825-826.
- [57] S. Sedaghat, F. Adibniya and M. Sarram, "The investigation of vulnerability test in application software" 2009 International Conference on the Current Trends in Information Technology (CTIT), pp. 1-5.
- [58] J. Bugeja, A. Jacobsson and P. Davidsson, "On Privacy and Security Challenges in Smart Connected Homes", 2016 European Intelligence and Security Informatics Conference (EISIC), pp. 172-175.
- [59] W. Z. Khan, M. Y. Aalsalem and M. K. Khan, "Five acts of consumer behavior: A potential security and privacy threat to Internet of Things", 2018 IEEE International Conference on Consumer Electronics (ICCE), pp. 1-3.
- [60] A. Dorri, S. S. Kanhere, R. Jurdak and P. Gauravaram, "Blockchain for IoT security and privacy: The case study of a smart home", 2017 IEEE International Conference on Pervasive Computing and Communications Workshops (PerCom Workshops), pp. 618-623.
- [61] A. Vernotte, "Research Questions for Model-Based Vulnerability Testing of Web Applications", 2013 IEEE Sixth International Conference on Software Testing, Verification and Validation, pp. 505-506.
- [62] Doupé, Adam & Cova, Marco & Vigna, Giovanni. 2010. Why Johnny Can't Pentest: An Analysis of Black-Box Web Vulnerability Scanners. Proc. DIMVA 2010. 6201. pp. 111-131.
- [63] I. Gordin, A. Graur, A. Potorac and D. Balan, "Security assessment of OpenStack cloud using outside and inside software tools", 2018 International Conference on Development and Application Systems (DAS), pp. 170-174.
- [64] B.J. Oates. "Researching Information Systems and Computing". SAGE Publications Ltd., 2006.

- [65] “IEEE Xplore Digital Library”, Ieeexplore.ieee.org, 2019. Online: <https://ieeexplore.ieee.org/Xplore/home.jsp>
- [66] “ACM Digital Library”, Dl.acm.org, 2019. Online: <https://dl.acm.org/>
- [67] Home Assistant. Online: <https://www.home-assistant.io/>
- [68] Home Assistant, Wikipedia. Online: [https://en.wikipedia.org/wiki/Home\\_Assistant](https://en.wikipedia.org/wiki/Home_Assistant)
- [69] Chromecast, Wikipedia -> <https://it.wikipedia.org/wiki/Chromecast>
- [70] Home Mini, Wikipedia. Online: [https://it.wikipedia.org/wiki/Google\\_Home](https://it.wikipedia.org/wiki/Google_Home)
- [71] Ip Camera, Wikipedia. Online: [https://it.wikipedia.org/wiki/Telecamera\\_IP](https://it.wikipedia.org/wiki/Telecamera_IP), [https://en.wikipedia.org/wiki/IP\\_camera](https://en.wikipedia.org/wiki/IP_camera)
- [72] Kali.org, “Configuring and Tuning OpenVAS in Kali Linux”, Kali. Online: <https://www.kali.org/blog/configuring-and-tuning-openvas-in-kali-linux/>
- [73] L. Liang, K. Zheng, Q. Sheng and X. Huang, “A Denial of Service Attack Method for an IoT System”, 2016 8th International Conference on Information Technology in Medicine and Education (ITME), pp. 360-364.
- [74] VirtualBox, “Welcome to VirtualBox.org”, VirtualBox. Online: <https://www.virtualbox.org/>
- [75] Kali.org, “What is Kali Linux?”, Kali. Online: <https://www.kali.org/docs/introduction/what-is-kali-linux/>
- [76] Dookie, “Kali Linux Blog”, Kali, Giugno 2021. Online: <https://www.kali.org/blog/>
- [77] Offensive Security. Online: <https://www.offensive-security.com/>
- [78] OpenVAS, “OpenVAS - Open Vulnerability Assessment Scanner”. Online: <https://openvas.org/>

[79] Linuxhint, John Otieno, “How to Install and Configure OpenVAS on Kali Linux”. Online: <https://linuxhint.com/install-openvas-kali-linux/>

[80] Youtube, “How to install and setup OpenVAS Vulnerability Scanner in Kali Linux 2020.1”. Online: <https://www.youtube.com/watch?v=oFeFufGQUVY>

[81] Tech Target, Linda Rosencrance, George Lawton, Chuck Moozakis, “UDP (User Datagram Protocol)”. Online: <https://searchnetworking.techtarget.com/definition/UDP-User-Datagram-Protocol>

[82] Greenbone, “Configuring and Managing Scan Configurations”. Online: <https://docs.greenbone.net/GSM-Manual/gos-21.04/en/scanning.html#configuring-and-managing-scan-configurations>

[83] K. Charles, “OpenVAS Terms to Know”, Security ORB, Giugno 2018. Online: <https://www.securityorb.com/general-security/openvas-term-to-know/>

[84] Greenbone Security Manager, Glossary, Solution Type. Online: <https://docs.greenbone.net/GSM-Manual/gos-21.04/en/glossary.html#solution-type>

[85] Greenbone, “VT Development”, Greenbone, Settembre 2018. Online: <https://community.greenbone.net/t/vt-development/226>

[86] Greenbone Security Manager, Managing SecInfo. Online: <https://docs.greenbone.net/GSM-Manual/gos-21.04/en/managing-secinfo.html>

[87] NVD, “Vulnerability Metrics”, NIST. Online: <https://nvd.nist.gov/vuln-metrics/cvss>

[88] Greenbone Security Manager, Glossary, CVSS. Online: <https://docs.greenbone.net/GSM-Manual/gos-21.04/en/glossary.html#cvss>

[89] Greenbone Security Manager, Glossary, Host. Online: <https://docs.greenbone.net/GSM-Manual/gos-21.04/en/glossary.html#host>

- [90] Greenbone Security Manager, Glossary, Port List. Online:  
<https://docs.greenbone.net/GSM-Manual/gos-21.04/en/glossary.html#port-list>
- [91] L. Constantin, “Stop using SHA1 encryption: It’s now completely unsafe, Google proves”, PCWorld. Online: <https://www.pcworld.com/article/3173791/stop-using-sha1-it-s-now-completely-unsafe.html>
- [92] L. Constantin, “Microsoft finally bans SHA-1 certificates in Internet Explorer and Edge”, PCWorld, Maggio 2017. Online: <https://www.pcworld.com/article/3195921/microsoft-finally-bans-sha-1-certificates-in-internet-explorer-and-edge.html>
- [93] Wikipedia, Directory Traversal attack. Online:  
[https://it.wikipedia.org/wiki/Directory\\_traversal\\_attack](https://it.wikipedia.org/wiki/Directory_traversal_attack)
- [94] J. Rauscher and B. Bauer, "Safety and Security Architecture Analyses Framework for the Internet of Things of Medical Devices," 2018 IEEE 20th International Conference on e-Health Networking, Applications and Services (Healthcom), 2018, pp. 1-3.
- [95] H. Poston “A brief Introduction to the OpenVAS Vulnerability Scanner”, Infosec, Ottobre 2018. Online: <https://resources.infosecinstitute.com/a-brief-introduction-to-the-openvas-vulnerability-scanner/#gref>
- [96] Mustafa, Ghulam & Ashraf, Rehan & Mirza, Muhammad & Jamil, Abid & Muhammad, “A review of data security and cryptographic techniques in IoT based devices”, 2018. Online: [https://www.researchgate.net/publication/327325243\\_A\\_review\\_of\\_data\\_security\\_and\\_cryptographic\\_techniques\\_in\\_IoT\\_based\\_devices](https://www.researchgate.net/publication/327325243_A_review_of_data_security_and_cryptographic_techniques_in_IoT_based_devices)
- [97] Wikipedia, Operatore di trasmissione energetica. Online:  
[https://it.wikipedia.org/wiki/Operatore\\_di\\_trasmissione\\_energetica](https://it.wikipedia.org/wiki/Operatore_di_trasmissione_energetica)
- [98] Iberdrola, “DSO - how to convert grid management towards a smarter system?”. Online: <https://www.iberdrola.com/innovation/distribution-system-operation>