



UNIVERSITÀ DEGLI STUDI DI MILANO

FACOLTÀ DI SCIENZE E TECNOLOGIE

Corso di Laurea in Sicurezza dei Sistemi e delle Reti Informatiche

STUDIO ED IMPLEMENTAZIONE DI UN SISTEMA PER L'ANALISI ED IL BLOCCO DI TRAFFICO CIFRATO MALEVOLO

Relatore:
Marco ANISETTI
Correlatore:
Nicola BENA

Tesi di Laurea di:
Michele MASTROBERTI
Matricola: 975314

Anno Accademico 2022/2023

Ringraziamenti

Desidero esprimere la mia sincera gratitudine a tutte le persone che hanno reso possibile la realizzazione di questa tesi. Ringrazio il mio Relatore, il Prof. Marco Anisetti ed il mio correlatore, il Dott. Nicola Bena, per la loro guida e il loro sostegno durante tutto il percorso di ricerca.

Un ringraziamento speciale va ai miei genitori e alla mia famiglia per il loro amore, il loro incoraggiamento costante e il loro supporto. Senza di loro, questo traguardo non sarebbe stato raggiunto.

Desidero anche ringraziare i miei colleghi e amici per le interessanti discussioni, le collaborazioni stimolanti e il sostegno morale nei momenti di difficoltà. Ogni conversazione e ogni confronto hanno arricchito il mio percorso di studio.

Infine, voglio ringraziare l'intera comunità accademica e tutte le persone che, direttamente o indirettamente, hanno contribuito a questo lavoro. Siete stati fonte di ispirazione e di motivazione.

Grazie a tutti coloro che hanno reso possibile questo viaggio accademico. Sono grato per tutte le esperienze e le conoscenze acquisite e guardo al futuro con entusiasmo e gratitudine.

Prefazione

Nel contesto in continua evoluzione dell'informatica e della sicurezza informatica la protezione dei sistemi e delle reti da minacce esterne assume un ruolo di primaria importanza. La presente tesi si inserisce in questo contesto, proponendo un approccio innovativo alla sicurezza delle reti, basato sull'analisi delle impronte digitali crittografiche per la rilevazione di connessioni malevole.

Le ragioni che hanno guidato la realizzazione di questo progetto sono di natura strategica e cruciale. Con l'incremento esponenziale del traffico di rete e la diffusione delle comunicazioni crittografate, le tradizionali metodologie di rilevamento delle minacce si sono spesso dimostrate insufficienti. Questa tesi si propone di affrontare questa sfida, proponendo un'implementazione basata sull'analisi delle impronte digitali crittografiche, allo scopo di identificare e mitigare le minacce informatiche in modo più efficace.

Organizzazione della tesi

La tesi è organizzata come segue:

- nel Capitolo 1 viene presentata un'introduzione generale alla tesi, delineando il contesto e le motivazioni alla base dell'importanza della sicurezza informatica.
- Nel Capitolo 2 viene presentato il contesto in cui si colloca la ricerca, sottolineando l'importanza della sicurezza delle reti e l'evoluzione delle minacce informatiche. Vengono esaminati i limiti delle attuali soluzioni di sicurezza informatica e si delinea la necessità di un nuovo approccio.
- Nel Capitolo 3 viene presentato il tool di sicurezza basato sull'analisi delle impronte digitali crittografiche JA3. Si descrive dettagliatamente il funzionamento del tool e la sua architettura, evidenziando le caratteristiche distintive che lo differenziano dalle soluzioni esistenti.

- Nel Capitolo 4 viene affrontata la fase sperimentale del progetto. Vengono presentati i risultati dei test condotti per valutare la capacità di carico sostenibile del tool e viene analizzata l'efficacia del tool nel rilevare e bloccare connessioni indesiderate.
- Nel Capitolo 5 vengono riassunti i risultati principali emersi dalla ricerca condotta, enfatizzando i successi e fornendo diverse aree di sviluppo futuro.

Indice

	ii
Ringraziamenti	iii
Prefazione	iv
1 Introduzione	1
1.1 Contesto	1
1.2 Motivazioni	2
2 Stato dell'arte	3
2.1 Introduzione	3
2.1.1 Analisi del traffico	4
2.1.2 Analisi delle firme	5
2.1.3 Analisi comportamentale	6
2.1.4 Analisi dei metadati	8
2.2 Rilevazione e blocco del traffico cifrato	10
2.2.1 JA3: Fingerprinting SSL/TLS	10
2.2.2 Approcci basati sul Machine Learning	11
2.2.3 Application Behavior Analysis	16
2.3 IDS e IPS	17
2.3.1 Introduzione agli IDS e IPS	18
2.3.2 Funzionamento degli IDS e IPS	18
2.3.3 Utilizzo nel Rilevamento delle Intrusioni	19
2.3.4 Vantaggi e Limitazioni	21
2.4 Firewall avanzati e tecnologie di supporto	23
2.5 eBPF	23
2.5.1 Introduzione a eBPF	23

2.5.2	Applicazione di eBPF	24
2.5.3	Vantaggi e Limitazioni	28
2.6	Utilizzo di Librerie Custom	29
2.6.1	Vantaggi e Limitazioni	31
3	Progettazione e Implementazione	33
3.1	Definizione dei Requisiti	33
3.1.1	Contesto e Motivazione	33
3.1.2	Obiettivi Specifici	34
3.1.3	Vincoli e Limitazioni	34
3.2	Architettura	36
3.2.1	Struttura Funzionale	36
3.2.2	Organizzazione Modulare	37
3.3	Implementazione	39
3.3.1	Inizializzazione e Configurazione	39
3.3.2	Funzioni di Analisi e Rilevamento	40
3.3.3	Avvio e Statistiche	43
3.3.4	Conclusioni	45
4	Fase Sperimentale	46
4.1	Introduzione	46
4.2	Valutazione del Carico Lavorativo	47
4.2.1	Configurazione Sperimentale	47
4.2.2	Risultati dei Test	48
4.2.3	Conclusioni	55
4.3	Analisi dell'Efficacia del Tool	56
4.3.1	Metodologia	56
4.3.2	Risultati Conseguiti	57
4.4	Conclusioni	64
5	Conclusioni	66
5.1	Riassunto delle Conclusioni	66
5.2	Lavori Futuri	67

Capitolo 1

Introduzione

In un mondo in costante evoluzione, caratterizzato dall'ubiquità delle reti e dalla crescente complessità delle minacce informatiche, la sicurezza delle informazioni è diventata un tema di fondamentale importanza. La protezione dei sistemi e delle reti da attacchi informatici è diventata una priorità per individui, aziende e organizzazioni di ogni dimensione. Questo capitolo introduttivo offre una visione più dettagliata del contesto e delle motivazioni alla base di questa tesi, delineando l'importanza della sicurezza informatica e presentando il contributo di questo lavoro.

1.1 Contesto

L'avvento delle reti di computer e delle tecnologie di comunicazione ha trasformato radicalmente la nostra società e la nostra economia. Oggi, l'accesso a Internet è diventato onnipresente, e una vasta gamma di servizi e applicazioni viene erogata attraverso reti globali. Questo progresso tecnologico ha portato con sé una serie di sfide, tra cui la crescente complessità delle minacce informatiche.

Le minacce informatiche sono diventate sempre più sofisticate e pericolose. Gli attacchi informatici possono prendere molte forme, tra cui malware, ransomware, attacchi DDoS (Distributed Denial of Service), phishing e molto altro. Le conseguenze di tali attacchi possono essere devastanti, con perdite finanziarie, violazioni della privacy e danni alla reputazione aziendale.

L'evoluzione delle minacce informatiche è stata alimentata dalla crescente digitalizzazione dei processi aziendali, dalla diffusione delle tecnologie cloud e dalla proliferazione dei dispositivi connessi, noti come Internet of Things (IoT). Questi sviluppi

hanno creato nuovi vettori di attacco e hanno reso più complessa la protezione delle infrastrutture IT.

Per affrontare queste sfide, le organizzazioni di tutto il mondo investono ingenti risorse nella sicurezza delle informazioni. Soluzioni tradizionali come firewall, antivirus e sistemi di rilevamento delle intrusioni (IDS) sono spesso integrate in complesse architetture di sicurezza. Tuttavia, l'evoluzione delle minacce richiede approcci innovativi e flessibili alla sicurezza.

1.2 Motivazioni

Le sfide poste dalla crescente complessità delle minacce informatiche richiedono un approccio innovativo alla sicurezza delle reti e dei sistemi. In questo contesto, questa tesi si propone di esplorare un nuovo paradigma basato sull'analisi delle impronte digitali crittografiche (JA3) per la rilevazione di connessioni malevole.

Le motivazioni alla base di questo progetto sono molteplici. In primo luogo, l'analisi delle impronte digitali crittografiche offre un modo innovativo di rilevare le minacce informatiche, concentrandosi sul comportamento delle connessioni di rete anziché sulla firma di malware specifici. Questo approccio consente di identificare attività sospette anche in assenza di segni evidenti di un attacco.

In secondo luogo, la crittografia è diventata ampiamente utilizzata nelle comunicazioni su Internet, rendendo il traffico di rete sempre più opaco per le soluzioni tradizionali di sicurezza. L'analisi delle impronte digitali crittografiche si basa sulla caratterizzazione unica di ciascuna implementazione di protocollo crittografico, consentendo di identificare connessioni malevole anche quando il traffico è crittografato.

Infine, questa tesi contribuirà alla comprensione e allo sviluppo di un tool di sicurezza basato su JA3, fornendo una base solida per ulteriori ricerche e implementazioni in questo campo.

Capitolo 2

Stato dell'arte

2.1 Introduzione

Nell'ambito della sicurezza informatica, il proliferare di minacce criptate negli ultimi anni costituisce una preoccupazione sempre più rilevante. Questo fenomeno è alimentato dall'avanzamento delle tecnologie di crittografia e dall'ampia adozione delle comunicazioni cifrate, che offrono opportunità sempre più frequenti ai criminali informatici per mascherare le loro attività malevole.

Le minacce criptate si rivelano particolarmente insidiose, poiché consentono di trasmettere e archiviare dati senza rivelarne il contenuto, sfuggendo così alle tradizionali tecniche di rilevamento e prevenzione delle minacce. Questa capacità di occultare il traffico malevolo rappresenta una sfida significativa per la sicurezza informatica, poiché può consentire l'esecuzione di attacchi dannosi senza essere tempestivamente individuati e contrastati.

L'utilizzo delle minacce criptate è associato a una vasta gamma di potenziali danni e pericoli per gli utenti e le organizzazioni. Tra i rischi più comuni vi sono il furto di dati sensibili, la diffusione di malware, il danneggiamento delle infrastrutture di rete e l'attuazione di attacchi mirati. Queste minacce possono colpire sia singoli utenti che aziende, con conseguenze che spaziano dalla perdita di informazioni riservate alla compromissione dell'integrità dei sistemi e alla violazione della privacy.

Affrontare con successo le minacce criptate richiede un'approfondita comprensione delle loro modalità operative e delle tecniche di crittografia utilizzate dai criminali informatici. Solo attraverso l'adozione di approcci innovativi e avanzati di rilevamento e blocco del traffico cifrato sarà possibile mitigare l'impatto di queste minacce e preservare l'integrità e la sicurezza delle reti e dei dati.

Nelle prossime sezioni, saranno esaminati diversi approcci e contromisure per gestire le minacce criptate e proteggere le reti e i sistemi informatici. Tra le strategie analizzate troviamo: Analisi del traffico (2.1.1), Analisi delle firme (2.1.2), Analisi comportamentale (2.1.3) e Analisi dei metadati (2.1.4). Questi approcci combinano diverse metodologie e tecniche per rilevare e mitigare le minacce criptate, fornendo una base solida per lo sviluppo di sistemi di rilevamento delle intrusioni e firewall innovativi e resilienti.

2.1.1 Analisi del traffico

L'analisi del traffico è una contromisura fondamentale per identificare potenziali minacce criptate. Essa si basa sull'osservazione del flusso di dati e delle caratteristiche del traffico, come indirizzi IP sospetti, porte di rete insolite e modelli di comunicazione anomali.

Vantaggi

- **Rilevamento di minacce criptate con canali di comunicazione nascosti:** L'analisi del traffico consente di individuare minacce che utilizzano canali di comunicazione nascosti o non standard per trasmettere dati crittografati. Attraverso l'osservazione dei pattern di traffico, è possibile identificare attività sospette e intraprendere azioni correttive.
- **Identificazione di indirizzi IP sospetti:** L'analisi del traffico può rilevare indirizzi IP sospetti o associati a attività malevole. Ciò consente di bloccare o monitorare attentamente le comunicazioni provenienti da tali indirizzi, riducendo così il rischio di intrusioni o attacchi.
- **Rilevamento di porte di rete insolite:** Le porte di rete insolite possono indicare attività malevole o tentativi di bypassare le misure di sicurezza. L'analisi del traffico consente di individuare l'uso di porte non standard e di intraprendere azioni preventive per proteggere il sistema.

Svantaggi

- **Possibilità di occultamento del traffico:** Gli attaccanti possono utilizzare tecniche di occultamento per mascherare il loro traffico e renderlo meno rilevabile all'analisi del traffico. Ad esempio, possono criptare i dati, utilizzare canali di

comunicazione alternativi o modificare il flusso del traffico per evitare la rilevazione. Questa sfida richiede l'adozione di metodi più sofisticati per identificare e contrastare tali minacce.

- **Complessità dell'analisi dei pattern di traffico:** L'analisi del traffico richiede un'analisi approfondita dei pattern e delle caratteristiche del traffico per identificare potenziali minacce. Questa analisi può richiedere tempo e risorse significative, soprattutto in ambienti di rete complessi con un alto volume di traffico.

Un esempio può essere trovato in [1]. L'autore fornisce un'approfondita analisi dell'utilizzo delle tecniche di analisi del traffico per il rilevamento delle intrusioni. L'articolo dimostra l'efficacia dell'analisi del traffico nel rilevare anomalie di rete e intrusioni, fornendo una base solida per lo sviluppo di sistemi di rilevamento delle intrusioni basati su questa tecnica.

2.1.2 Analisi delle firme

L'analisi delle firme è una potente contromisura utilizzata per identificare minacce criptate sfruttando firme digitali e database di minacce conosciute. Questa tecnica si basa sulla rilevazione di specifici modelli di dati crittografati che corrispondono alle firme digitali di minacce precedentemente identificate e catalogate.

Per capire meglio il funzionamento dell'analisi delle firme, è utile immaginare un vasto database contenente firme digitali di malware, virus e altri tipi di minacce informatiche noti. Queste firme digitali rappresentano specifiche sequenze di dati crittografati che caratterizzano un particolare tipo di minaccia. Ad esempio, un virus specifico potrebbe avere una firma digitale unica che identifica i suoi comportamenti tipici quando cifrato.

Quando il traffico di rete viene analizzato, l'analisi delle firme confronta i dati crittografati presenti con le firme digitali nel database. Se viene trovata una corrispondenza, significa che il traffico contiene una minaccia conosciuta. Di conseguenza, il sistema può attivare le contromisure appropriate per contrastare l'attacco.

Vantaggi

- **Rilevamento di minacce conosciute:** L'analisi delle firme consente di rilevare minacce criptate già note e catalogate in un database di firme digitali.

Quando un file crittografato corrisponde a una firma presente nel database, viene identificato come minaccia nota, consentendo di prendere tempestivamente le misure appropriate per contrastarla.

- **Efficienza nella rilevazione:** Poiché l'analisi delle firme si basa su pattern di dati crittografati precedentemente identificati, può rilevare minacce in modo rapido ed efficiente. Questo metodo è particolarmente utile per individuare varianti di minacce conosciute che utilizzano lo stesso pattern di crittografia.
- **Aggiornamento dei database di firme:** I database di firme vengono costantemente aggiornati con nuove firme di minacce criptate. Ciò consente di mantenere il sistema di rilevamento all'avanguardia, garantendo la protezione contro le minacce più recenti.

Svantaggi

- **Inefficace contro minacce sconosciute:** L'analisi delle firme è limitata nell'individuare minacce criptate che non corrispondono a firme digitali presenti nel database. Queste minacce sconosciute possono sfuggire alla rilevazione e rappresentare un rischio per il sistema.
- **Necessità di aggiornamenti frequenti:** Poiché nuove minacce criptate vengono costantemente sviluppate, i database di firme digitali devono essere regolarmente aggiornati per garantire un'adeguata protezione. Ciò richiede un impegno costante per mantenere i sistemi di rilevamento aggiornati.

Un esempio di firma digitale utilizzata nell'analisi delle firme è il *CryptoWall*, un noto ransomware. La sua firma digitale può essere rappresentata come:

SHA256: E8DEA9D52E32B9F37C8217A9C1FE0C50AC9EF1CE2F9E9914C76F692EF0B8E0D1

L'analisi delle firme consente di rilevare la presenza del ransomware CryptoWall confrontando la firma digitale del file crittografato con quella presente nel database di firme. [2]

2.1.3 Analisi comportamentale

L'analisi comportamentale è una contromisura che si basa sull'osservazione del comportamento degli utenti e dei sistemi per individuare attività sospette e anomale.

Questo approccio si concentra sull'identificazione di modelli di comportamento che possono indicare la presenza di minacce criptate.

Vantaggi

- **Rilevamento di minacce sconosciute:** L'analisi comportamentale è in grado di individuare minacce criptate che non corrispondono a pattern o firme digitali conosciute. Questo metodo si basa sull'osservazione dei comportamenti anomali che possono indicare la presenza di una minaccia, consentendo di reagire tempestivamente.
- **Adattabilità:** L'analisi comportamentale è in grado di adattarsi alle nuove minacce criptate che evolvono nel tempo. Poiché si concentra sui comportamenti anziché sulle firme specifiche, può rilevare varianti di minacce conosciute e nuovi schemi di attacco.
- **Riduzione dei falsi positivi:** Questo approccio riduce il rischio di falsi positivi, poiché si basa su evidenze comportamentali anziché su criteri rigidi di corrispondenza di pattern o firme.

Svantaggi

- **Complessità dell'analisi:** L'analisi comportamentale richiede l'utilizzo di algoritmi avanzati per l'identificazione dei modelli di comportamento anomali. Ciò richiede competenze specializzate e l'impiego di risorse computazionali significative.
- **Possibili falsi negativi:** L'analisi comportamentale potrebbe non rilevare alcune minacce criptate se il loro comportamento non è sufficientemente anomalo da scatenare un'allerta. Le minacce che si mimetizzano bene nel normale comportamento del sistema o degli utenti potrebbero non essere identificate.
- **Necessità di aggiornamenti costanti:** Poiché il comportamento delle minacce criptate può evolversi nel tempo, è necessario mantenere aggiornati gli algoritmi di analisi comportamentale per garantire una rilevazione efficace.

Nell'ambito dell'analisi comportamentale delle minacce interne, l'approccio proposto dal framework BAIT ha individuato un comportamento sospetto all'interno dell'organizzazione. Attraverso l'uso di algoritmi di machine learning e l'analisi di

tracce comportamentali di soggetti reali, il sistema ha rilevato un insieme di attività anomale, segnalando un tentativo di esfiltrazione di dati sensibili da parte di un utente. Grazie alla combinazione di tecniche di apprendimento supervisionato e di bootstrapping, l'analisi ha fornito un elevato livello di precisione e richiamo nel rilevamento di questa minaccia interna. (Esempio adattato dall'articolo [3])

2.1.4 Analisi dei metadati

L'analisi dei metadati è una contromisura che si basa sull'ispezione dei metadati associati ai dati per rivelare modelli sospetti o inconsueti, come comunicazioni anomale o attività inusuali. I metadati possono fornire informazioni preziose sul contesto e sulle caratteristiche delle comunicazioni crittografate, consentendo di identificare comportamenti fuori dal comune.

Vantaggi

- **Individuazione di pattern anomali:** L'analisi dei metadati può rivelare pattern di comunicazioni sospette o inconsuete, come un elevato numero di connessioni a un determinato server o l'uso di protocolli non comuni. Questa informazione può aiutare a individuare potenziali minacce criptate.
- **Rilevamento di comportamenti fuori dal comune:** L'analisi dei metadati può identificare comportamenti anomali, come un utente che accede a risorse di rete non pertinenti al suo ruolo o che trasferisce una grande quantità di dati in tempi insolitamente brevi. Questi comportamenti possono indicare attività sospette.
- **Integrazione con altre fonti di informazioni:** L'analisi dei metadati può essere integrata con altre fonti di informazioni, come i log di sistema o i dati di monitoraggio delle reti. Questa integrazione consente di ottenere una visione più completa delle attività e di rilevare correlazioni tra i diversi eventi.

Svantaggi

- **Privacy e protezione dei dati:** L'analisi dei metadati può implicare la raccolta e l'elaborazione di informazioni sensibili, sollevando preoccupazioni sulla privacy e sulla protezione dei dati personali. È importante adottare adeguate misure di sicurezza e rispettare le normative sulla privacy durante l'implementazione di questa contromisura.

- **Complessità dell'analisi:** L'analisi dei metadati richiede l'utilizzo di algoritmi complessi per identificare pattern e comportamenti anomali. Ciò richiede competenze specializzate e l'impiego di risorse computazionali significative.
- **Possibili falsi positivi:** L'analisi dei metadati potrebbe generare falsi positivi, identificando comportamenti come sospetti quando in realtà sono legittimi. Questo può causare problemi di notifica e richiedere ulteriori verifiche per distinguere tra attività legittime e minacce criptate.

Un esempio di analisi dei metadati è l'individuazione di un utente che, in un breve lasso di tempo, stabilisce connessioni con un alto numero di indirizzi IP sconosciuti e non comuni. Questo può suggerire un tentativo di evasione delle misure di sicurezza o un'attività fraudolenta. L'analisi dei metadati può aiutare a identificare tali modelli di comunicazione sospetti.

(Esempio adattato dall'articolo [4])

In Tabella 1 sono riassunte le contromisure discusse, insieme ai loro vantaggi e svantaggi principali.

Tabella 1: Riassunto delle contromisure

Contromisura	Vantaggi	Svantaggi
Analisi del traffico	Individua attività sospette basate sulle caratteristiche del traffico	Potenziati falsi positivi o negativi
Analisi delle firme	Rileva minacce conosciute e attacchi noti	Può essere inefficace contro nuove minacce
Analisi comportamentale	Identifica comportamenti anomali e attività sospette	Richiede una definizione accurata dei comportamenti normali
Analisi dei metadati	Rileva modelli di comunicazione inconsueti	Possibili questioni sulla privacy e sulle normative

2.2 Rilevazione e blocco del traffico cifrato

Per affrontare il problema del rilevamento e del blocco del traffico cifrato in modo efficace, è necessario considerare approcci innovativi che superino le tradizionali limitazioni a livello 3/4 dello stack ISO/OSI. In questa sezione, verranno esaminati alcuni approcci rilevanti che mirano a risolvere questa sfida.

2.2.1 JA3: Fingerprinting SSL/TLS

JA3 è un metodo di fingerprinting basato sulle caratteristiche del *Three Way Handshake* SSL/TLS (Secure Sockets Layer/Transport Layer Security). Durante la negoziazione di una connessione SSL/TLS, i client e i server comunicano attraverso una serie di messaggi di handshake, che includono informazioni come le versioni del protocollo, le suite di cifratura supportate e altre opzioni di configurazione.

Il processo di fingerprinting JA3 consiste nel calcolare un hash univoco basato sui valori dei campi crittografici presenti nel handshake SSL/TLS. In pratica, vengono presi in considerazione diversi campi, come la versione del protocollo, la lista delle suite di cifratura, i metodi di compressione supportati e le estensioni del client. Tutte

queste informazioni vengono combinate per creare un'unica impronta digitale (JA3 hash) che rappresenta il comportamento del client durante la negoziazione SSL/TLS.

L'idea alla base di JA3 è che diverse implementazioni di software e librerie SSL/TLS generano diverse combinazioni di campi nel handshake, creando impronte digitali uniche per ciascun client. Pertanto, un client utilizzante una specifica versione di software o configurazione di TLS avrà una JA3 hash specifica associata alla sua negoziazione SSL/TLS.

Una volta ottenuta l'impronta digitale del client, questa viene confrontata con un database di impronte digitali noti, che contiene impronte associate a clienti noti e benigni. Se la JA3 hash corrisponde a un client noto, la connessione può essere considerata affidabile e consentita. D'altra parte, se la JA3 hash non corrisponde a nessuna impronta digitale nota o corrisponde ad una impronta associata a clienti sospetti o malevoli, la connessione può essere bloccata o monitorata più attentamente.

Inoltre, JA3 può essere esteso per includere anche le impronte digitali dei server SSL/TLS. In questo caso, durante la negoziazione, il server invierà la sua propria impronta digitale, chiamata JA3S hash. Anche in questo caso, il JA3S hash del server può essere confrontato con un database di impronte digitali noti per identificare server benigni o sospetti.

L'utilizzo di JA3 per il fingerprinting SSL/TLS è stato dimostrato efficace nel rilevare il traffico cifrato malevolo e anomalo. La tecnica di fingerprinting è particolarmente utile per identificare comunicazioni malevoli all'interno di reti aziendali, poiché le impronte digitali generiche o insolite possono suggerire attività sospette.

Un aspetto vantaggioso di JA3 è la sua natura leggera, in quanto richiede solo l'analisi dei campi di handshake, senza la necessità di decifrare il traffico cifrato. Questo lo rende un'opzione interessante per il rilevamento del traffico cifrato malevolo senza sacrificare le prestazioni di rete.

2.2.2 Approcci basati sul Machine Learning

L'uso di algoritmi di Machine Learning ha dimostrato di essere un approccio promettente per il rilevamento e la classificazione del traffico cifrato malevolo. Questi algoritmi possono analizzare grandi quantità di dati di traffico in modo efficiente e, con un adeguato addestramento, possono identificare modelli e comportamenti associati a minacce o attività sospette. Di seguito, verranno esposte alcune delle tecniche di Machine Learning comunemente utilizzate nel contesto del rilevamento del traffico cifrato malevolo.

Support Vector Machine (SVM)

Support Vector Machine (SVM) è un algoritmo di apprendimento supervisionato, ampiamente utilizzato nel contesto del rilevamento del traffico cifrato malevolo. La sua versatilità e capacità di gestire dati complessi lo rendono una scelta popolare per affrontare questa sfida.

Nel rilevamento del traffico cifrato malevolo, SVM può essere addestrato su un dataset contenente esempi di traffico normale e malevolo. Questi esempi sono rappresentati da un insieme di feature estratte dai pacchetti crittografati, che possono includere informazioni come protocolli di rete, dimensioni dei pacchetti, tempi di latenza, e altro. L'obiettivo dell'addestramento SVM è definire un iperpiano che possa separare efficacemente i dati di traffico in due classi: benigno e malevolo.

Una volta addestrato, il modello SVM può essere utilizzato per classificare il traffico cifrato in tempo reale. Quando nuovi dati di traffico vengono presentati al modello, questo determina a quale classe appartengono in base alle caratteristiche estratte. Ad esempio, il modello SVM può identificare il traffico associato a un attacco informatico o a una minaccia di sicurezza, consentendo ai sistemi di prendere misure adeguate.

Un vantaggio chiave di SVM è la sua capacità di gestire dati ad alta dimensionalità e di affrontare distribuzioni non lineari. Questo significa che può riconoscere pattern complessi e sottostanti nei dati di traffico cifrato, anche quando le relazioni tra le feature non sono semplicemente lineari. Inoltre, SVM offre opzioni per gestire il bilanciamento tra le classi, il che è essenziale nel rilevamento delle minacce, in cui i dati malevoli possono essere significativamente meno frequenti dei dati benigni.

In conclusione, SVM è uno strumento potente nel rilevamento del traffico cifrato malevolo, in grado di gestire dati complessi e affrontare sfide legate alla classificazione. La sua flessibilità e capacità di separare efficacemente le classi lo rendono un alleato prezioso nella difesa delle reti e nella sicurezza informatica.

Regressione Lineare

La regressione lineare è un metodo di apprendimento automatico utilizzato anche nel contesto del rilevamento del traffico cifrato malevolo. A differenza degli algoritmi di classificazione come il Support Vector Machine (SVM) e il Random Forest che mirano a classificare i dati in categorie discrete, la regressione lineare è un algoritmo di apprendimento supervisionato che si concentra sulla previsione di valori continui.

Nel contesto del rilevamento del traffico cifrato malevolo, la regressione lineare può essere utilizzata per stimare parametri o valori legati ai dati di traffico. Ad esempio,

è possibile utilizzare la regressione lineare per:

- Predire la quantità di traffico cifrato in un determinato periodo di tempo.
- Stimare la velocità di crescita del traffico cifrato nel tempo.
- Prevedere il consumo di risorse di rete in base al traffico cifrato.

Per utilizzare la regressione lineare, è necessario avere un dataset contenente dati di traffico cifrato e le relative variabili dipendenti o target da predire. Le feature estratte dai pacchetti crittografati possono essere utilizzate come variabili indipendenti per l'analisi.

Una volta addestrato il modello di regressione lineare, è possibile utilizzarlo per fare previsioni su nuovi dati di traffico cifrato. Ad esempio, se si desidera stimare la crescita futura del traffico cifrato, il modello può essere utilizzato per generare una linea di tendenza basata sui dati storici.

Tuttavia, è importante notare che la regressione lineare assume una relazione lineare tra le variabili indipendenti e dipendenti, il che potrebbe non essere sempre il caso nei dati di traffico cifrato. Pertanto, è fondamentale eseguire un'analisi dei dati e valutare se la regressione lineare è il modello più appropriato per il problema specifico.

In conclusione, la regressione lineare è un'altra tecnica di apprendimento automatico che può essere utilizzata nel rilevamento del traffico cifrato malevolo per stimare valori continui o parametri legati ai dati di traffico.

Random Forest

Random Forest è un potente algoritmo di apprendimento supervisionato ampiamente utilizzato nel contesto del rilevamento del traffico cifrato malevolo. Si basa su un principio di aggregazione di decisioni, che lo rende particolarmente efficace nella gestione di dati complessi e rumorosi.

Nel rilevamento del traffico cifrato malevolo, Random Forest si distingue per la sua abilità nel gestire dati eterogenei e con un alto grado di variabilità. Questo algoritmo opera creando un insieme di alberi decisionali, noti come "forest." Ogni albero è addestrato su un sottoinsieme casuale dei dati di addestramento, e la loro combinazione viene utilizzata per ottenere una classificazione finale.

La forza di Random Forest risiede nella sua capacità di gestire il rumore nei dati. Nei contesti di sicurezza informatica, i dati di traffico cifrato possono essere

soggetti a fluttuazioni e interferenze, ma Random Forest è in grado di mitigare questi effetti indesiderati. Inoltre, questo algoritmo è altamente resistente all'overfitting, il che significa che è meno incline a creare modelli che si adattano troppo ai dati di addestramento e quindi perdono la capacità di generalizzazione.

Un aspetto notevole di Random Forest è la sua abilità nel riconoscere pattern complessi associati al traffico cifrato malevolo. Poiché gli attacchi informatici possono essere altamente sofisticati e sfruttare comportamenti sottili, Random Forest è in grado di identificarli efficacemente, anche quando le relazioni tra le feature non sono lineari o ovvie.

Inoltre, Random Forest offre la possibilità di valutare l'importanza delle feature utilizzate per la classificazione. Questo è fondamentale nel contesto del rilevamento delle minacce, poiché consente di identificare quali caratteristiche del traffico cifrato contribuiscono maggiormente alla distinzione tra traffico benigno e malevolo.

In conclusione, Random Forest è un algoritmo versatile e robusto nel rilevamento del traffico cifrato malevolo. La sua capacità di gestire dati complessi e rumorosi, insieme alla sua abilità nel riconoscere pattern sofisticati, lo rende uno strumento prezioso nella difesa delle reti e nella sicurezza informatica.

Deep Learning

Il Deep Learning è una potente branca dell'apprendimento automatico che ha dimostrato di essere altamente efficace nel contesto del rilevamento del traffico cifrato malevolo. Questo approccio si basa sull'uso di reti neurali artificiali profonde, che sono in grado di apprendere automaticamente le caratteristiche complesse e i pattern associati alle minacce nel traffico cifrato.

Nel rilevamento del traffico cifrato malevolo, le reti neurali profonde possono essere addestrate su grandi dataset contenenti sessioni di traffico cifrato. Questi dataset includono sia esempi di traffico benigno che di traffico malevolo, consentendo alle reti neurali di imparare le differenze cruciali tra i due.

Una delle caratteristiche distintive delle reti neurali profonde è la loro capacità di catturare relazioni complesse e non lineari tra le feature del traffico cifrato. Questo è particolarmente importante, poiché gli attacchi informatici possono essere altamente sofisticati e possono sfuggire a modelli più semplici.

Una volta addestrati, i modelli di Deep Learning possono essere utilizzati per la classificazione in tempo reale del traffico cifrato. Questo significa che possono identificare istantaneamente se una sessione di traffico è benigna o sospetta, consentendo una risposta rapida alle minacce.

Tuttavia, è importante notare che l'addestramento di reti neurali profonde richiede una grande quantità di dati e risorse computazionali. Inoltre, la configurazione dei parametri delle reti neurali può essere complessa e richiedere un'ottimizzazione attenta.

In conclusione, il Deep Learning rappresenta un approccio avanzato e potente nel rilevamento del traffico cifrato malevolo. Le reti neurali profonde sono in grado di catturare pattern complessi e sono particolarmente adatte per affrontare minacce informatiche sofisticate. Tuttavia, è importante considerare le sfide legate alla raccolta di dati e alla configurazione delle reti neurali per garantire risultati ottimali.

Considerazioni

Nel contesto dell'applicazione di tecniche di Machine Learning per il rilevamento del traffico cifrato malevolo, un importante contributo è stato fornito dall'articolo "Machine Learning for Encrypted Malware Traffic Classification" di Blake Anderson e David McGrew [5]. Questo studio affronta in modo approfondito le sfide legate alla classificazione del traffico cifrato malevolo, offrendo un'analisi dettagliata e soluzioni pratiche.

L'articolo identifica due sfide principali che influenzano l'efficacia dell'apprendimento automatico in questo contesto: l'accuratezza del ground truth e la non-stazionarietà nella distribuzione dei dati di rete cifrati. Gli autori conducono una serie di esperimenti per valutare le prestazioni di sei comuni algoritmi di Machine Learning quando sono confrontati con dati di traffico di rete reali.

Uno degli aspetti rilevanti emersi dalla ricerca è l'importanza dell'ingegneria delle feature. Gli autori hanno scoperto che l'iterazione sul set di feature iniziale e l'inclusione di feature suggerite dagli esperti del dominio hanno avuto un impatto significativo sulle prestazioni del sistema di classificazione. Ad esempio, il confronto tra una regressione lineare che utilizza un set di feature più espressivo e il metodo Random Forest con una rappresentazione standard del traffico di rete ha dimostrato che la regressione lineare ha superato il Random Forest su tutti i criteri considerati.

Inoltre, gli autori discutono come l'aggiunta di feature aggiuntive possa mitigare le sfide legate all'etichettatura accurata del ground truth. Ad esempio, considerando il fatto che il malware spesso effettua controlli di connettività visitando siti web standard, è difficile distinguerli da client benigni che visitano gli stessi siti utilizzando rappresentazioni di feature standard. Tuttavia, l'aggiunta di feature relative ai metadati della handshake TLS ha reso possibile questa distinzione.

In conclusione, l'articolo di Anderson e McGrew offre un'analisi dettagliata delle sfide e delle soluzioni nel contesto della classificazione del traffico cifrato malevolo utilizzando Machine Learning. Le loro scoperte forniscono importanti linee guida per i professionisti della sicurezza di rete e dimostrano che l'ingegneria delle feature gioca un ruolo cruciale nel migliorare le prestazioni dei modelli di classificazione.

2.2.3 Application Behavior Analysis

L'analisi del comportamento delle applicazioni, nota anche come *Application Behavior Analysis*, è una tecnica innovativa per il rilevamento e la mitigazione del traffico cifrato malevolo. Questo approccio si concentra sull'identificazione delle caratteristiche uniche del comportamento delle diverse applicazioni che generano il traffico cifrato. Ogni applicazione ha modelli di traffico specifici, e l'analisi di tali modelli può rivelare anomalie indicative di attività malevole.

Funzionamento delle Applicazioni

L'analisi del comportamento delle applicazioni si basa sulla raccolta e l'elaborazione dei dati di traffico generati dalle applicazioni in esame. Questi dati includono informazioni sulle comunicazioni di rete, la frequenza delle connessioni, i protocolli utilizzati, le dimensioni dei pacchetti e altri metadati associati alle transazioni.

Un sistema di analisi del comportamento delle applicazioni può utilizzare tecniche di machine learning e algoritmi avanzati per estrarre i modelli comportamentali di ciascuna applicazione. Durante la fase di addestramento, il sistema viene esposto a un campione rappresentativo di traffico delle applicazioni legittime e apprende i loro modelli di comportamento normali.

Identificazione delle Anomalie

Una volta addestrato, il sistema è in grado di rilevare comportamenti anomali nel traffico delle applicazioni. Quando una nuova attività di rete è in corso, il sistema confronta i modelli del traffico attuale con quelli appresi durante la fase di addestramento. Qualsiasi deviazione significativa dai modelli comportamentali noti viene considerata un'anomalia.

Le anomalie possono essere indicative di traffico cifrato malevolo. Ad esempio, un'applicazione legittima potrebbe utilizzare una sequenza di porte di comunicazione

ben definita, mentre un traffico malevolo potrebbe tentare di eludere questa caratteristica tipica utilizzando porte casuali o inusuali. L'analisi del comportamento delle applicazioni può rilevare queste anomalie e avviare azioni di mitigazione per proteggere la rete da attività malevole.

Vantaggi dell'Analisi del Comportamento delle Applicazioni

L'utilizzo dell'analisi del comportamento delle applicazioni offre diversi vantaggi nel contesto del rilevamento del traffico cifrato malevolo. Innanzitutto, questa tecnica si basa sui comportamenti distintivi delle applicazioni e non sul contenuto dei dati cifrati, rispettando quindi la privacy degli utenti e delle comunicazioni. Inoltre, poiché si concentra su modelli comportamentali, può essere efficace anche nel rilevamento di nuove minacce che utilizzano tecniche di evasione avanzate.

Un altro vantaggio dell'analisi del comportamento delle applicazioni è la sua capacità di rilevare attività malevole in tempo reale. Il sistema può identificare rapidamente anomalie nel traffico e intraprendere azioni correttive senza interrompere le attività legittime degli utenti.

Svantaggi dell'Analisi del Comportamento delle Applicazioni

Tuttavia, l'analisi del comportamento delle applicazioni può essere soggetta a falsi positivi, ovvero può identificare erroneamente attività legittime come anomalie. Questo può portare a interruzioni non necessarie dei servizi e causare frustrazioni agli utenti.

Inoltre, l'efficacia dell'analisi del comportamento delle applicazioni dipende dalla disponibilità di dati di traffico rappresentativi per l'addestramento del sistema. Se i dati di addestramento sono limitati o non rappresentativi, il sistema potrebbe non essere in grado di rilevare correttamente nuove minacce o comportamenti anomali.

2.3 IDS e IPS

Nel contesto della sicurezza informatica, gli *IDS* (Intrusion Detection Systems) e gli *IPS* (Intrusion Prevention Systems) rivestono un ruolo di estrema importanza nella preservazione dell'integrità, della confidenzialità e della disponibilità delle reti e dei sistemi informativi. Questi sistemi costituiscono un fondamento cruciale per la sicurezza cibernetica moderna, consentendo alle organizzazioni di identificare, monitorare e mitigare le minacce digitali in tempo reale.

2.3.1 Introduzione agli IDS e IPS

Gli *IDS* e gli *IPS* sono concepiti per operare in maniera proattiva e reattiva nel panorama della sicurezza informatica. Il loro obiettivo primario è quello di difendere contro un'ampia gamma di minacce, che includono attacchi informatici, malware, tentativi di intrusione, violazioni della politica di sicurezza e altro ancora.

Gli *IDS* si concentrano principalmente sulla rilevazione delle attività malevole, esaminando il traffico di rete, i registri di sistema e altri dati pertinenti per individuare eventuali indicatori di compromissione. D'altra parte, gli *IPS* non si limitano alla rilevazione, ma possono anche intraprendere azioni preventive, come il blocco delle connessioni sospette o l'applicazione di politiche di sicurezza predefinite per mitigare attacchi noti.

2.3.2 Funzionamento degli IDS e IPS

Gli *IDS* e gli *IPS* operano attraverso diverse modalità di rilevazione e prevenzione delle intrusioni, ognuna delle quali contribuisce alla loro efficacia nella difesa delle reti e dei sistemi informativi.

Rilevazione delle Firme

La rilevazione basata su firme è una delle modalità più tradizionali di funzionamento degli *IDS* e degli *IPS*. Questo metodo implica la comparazione del traffico di rete o dei dati di sistema con un vasto database di firme di attacchi noti. Ogni firma rappresenta un modello di comportamento associato a un tipo specifico di attacco informatico. Quando il traffico o le attività in corso corrispondono in modo significativo a una di queste firme, il sistema segnala la possibile presenza di una minaccia.

Ad esempio, se un *IDS* rileva una sequenza di byte in una connessione di rete che corrisponde esattamente a una firma di un noto worm, emetterà un avviso o prenderà azioni di rilevamento, come l'isolamento della connessione o la registrazione degli eventi.

Rilevazione Basata su Anomalie

La rilevazione basata su anomalie è una modalità più avanzata e sofisticata utilizzata dagli *IDS* e gli *IPS*. Questo approccio si basa sull'idea che il comportamento normale di una rete o di un sistema sia prevedibile e costante nel tempo. Gli *IDS* e gli *IPS*

creano modelli del comportamento normale attraverso l'analisi storica dei dati di traffico e delle attività di sistema.

Durante l'operazione di rilevazione basata su anomalie, il sistema monitora costantemente il traffico e le attività e confronta i dati in tempo reale con il modello previsto di comportamento. Quando vengono rilevate deviazioni significative da questo modello, il sistema segnala un'anomalia. Questo tipo di rilevazione è particolarmente utile per individuare attività sospette che potrebbero non corrispondere a firme di attacchi noti ma che comunque rappresentano una minaccia.

Ad esempio, se un utente che normalmente accede a una rete aziendale solo durante le ore lavorative inizia a effettuare accessi notturni non autorizzati, il sistema basato su anomalie potrebbe segnalare questa deviazione come un comportamento sospetto.

Rilevazione Euristica

La rilevazione euristica è un'ulteriore modalità di funzionamento degli *IDS* e degli *IPS*. Questa modalità si basa su regole euristiche predefinite che rappresentano modelli di comportamento potenzialmente dannosi. Queste regole sono progettate per identificare comportamenti che potrebbero indicare una minaccia, anche se non corrispondono esattamente a firme di attacchi noti o deviazioni significative dai modelli di comportamento previsti.

Le regole euristiche possono essere progettate per rilevare comportamenti come il tentativo di accesso ripetuto con password errate, la condivisione di file sensibili all'esterno della rete aziendale o l'esecuzione di comandi sospetti su un sistema. Quando una di queste regole viene attivata, il sistema può segnalare l'evento come possibile minaccia.

In sintesi, gli *IDS* e gli *IPS* sfruttano queste modalità di rilevazione in combinazione o separatamente per identificare e, in caso di *IPS*, anche prevenire le intrusioni e le attività malevole nelle reti e nei sistemi informativi.

2.3.3 Utilizzo nel Rilevamento delle Intrusioni

Gli *IDS* e gli *IPS* possono essere implementati in diverse posizioni all'interno di un'infrastruttura di rete, inclusi dispositivi di rete, server, sistemi endpoint e gateway. La loro distribuzione strategica consente di rilevare e prevenire intrusioni in diverse fasi del percorso del traffico di rete, fornendo così una difesa in profondità contro le minacce.

Questi sistemi vengono utilizzati sia per proteggere reti interne che per difendere contro minacce provenienti dall'esterno. L'implementazione di *IDS* e *IPS* contribuisce a garantire che le organizzazioni possano identificare e rispondere prontamente a potenziali attacchi informatici, mantenendo la sicurezza delle risorse digitali.

In un contesto più ampio, l'articolo "Overview of Intrusion Detection and Intrusion Prevention"[6] fornisce una visione approfondita di come gli *IDS* e gli *IPS* possano essere utilizzati per rilevare e prevenire intrusioni. L'articolo presenta una distinzione chiara tra *IDS* e *IPS*, in cui un *IDS* si concentra sulla rilevazione delle intrusioni, mentre un *IPS* cerca di reagire in tempo reale per prevenire la manipolazione indesiderata dei sistemi informatici.

Uno degli aspetti fondamentali evidenziati dall'articolo è la differenza tra il rilevamento basato su firme (misuse detection) e il rilevamento basato su anomalie (anomaly detection). Mentre il rilevamento basato su firme implica la definizione di modelli di comportamento indesiderato o non autorizzato (firme), il rilevamento basato su anomalie si concentra sul riconoscimento di comportamenti anomali rispetto al comportamento atteso. Quest'ultimo approccio offre la possibilità di individuare nuovi tipi di attacchi, ma può generare più falsi allarmi.

L'articolo sottolinea anche l'importanza del tempestivo aggiornamento delle regole e delle firme utilizzate dagli *IDS* e dagli *IPS* per garantire l'efficacia della rilevazione e la riduzione dei falsi allarmi. Questo aspetto è fondamentale per il corretto funzionamento di tali sistemi.

Nel contesto degli *IDS* e degli *IPS*, le organizzazioni devono valutare attentamente le proprie esigenze e risorse. Ad esempio, mentre le organizzazioni militari o quelle che gestiscono dati altamente sensibili possono preferire l'uso di *IPS* per reagire in tempo reale e impedire l'accesso non autorizzato, altre organizzazioni, come biblioteche pubbliche o siti web di informazioni sportive, potrebbero optare per una soluzione di *IDS* per la semplice rilevazione e segnalazione delle intrusioni.

In sintesi, l'articolo citato offre una panoramica dettagliata di come gli *IDS* e gli *IPS* possano essere impiegati per rilevare e prevenire intrusioni e fornisce importanti considerazioni sulle differenze tra queste due tipologie di sistemi. La sua analisi è stata fondamentale per comprendere il contesto in cui gli *IDS* e gli *IPS* sono stati utilizzati in diverse organizzazioni, fornendo una base solida per valutare l'efficacia di tali sistemi nel contesto specifico di questa ricerca.

2.3.4 Vantaggi e Limitazioni

Gli *IDS* e gli *IPS* sono strumenti fondamentali nel panorama della sicurezza cibernetica, ma è importante considerare sia i loro vantaggi che le limitazioni associate al loro utilizzo.

Vantaggi

Tra i principali vantaggi di questi sistemi, vi sono:

- **Rilevazione Tempestiva delle Minacce:** Gli *IDS* sono in grado di individuare potenziali attacchi e intrusioni in tempo reale o quasi reale. Questo significa che possono rilevare attività sospette mentre si verificano, consentendo alle organizzazioni di reagire prontamente per mitigare i danni.
- **Prevenzione Attiva:** Gli *IPS* vanno oltre la semplice rilevazione e possono intraprendere azioni attive per impedire un'attività malevola. Ciò significa che possono bloccare in modo proattivo le minacce, impedendo loro di avere successo.
- **Riduzione dei Rischi:** L'utilizzo di *IDS* e *IPS* aiuta a ridurre i rischi legati a violazioni di sicurezza e intrusioni. Rilevando e prevenendo le minacce, le organizzazioni possono proteggere le proprie risorse digitali e dati sensibili.
- **Allarme Anticipato:** La capacità di generare avvisi tempestivi in caso di comportamenti sospetti consente agli amministratori di rete di essere informati prontamente su potenziali minacce. Questo aiuta a avviare indagini e risposte rapide.
- **Conformità Normativa:** Gli *IDS* e gli *IPS* sono spesso necessari per conformarsi a regolamenti e standard di sicurezza. La loro implementazione può aiutare le organizzazioni a soddisfare requisiti normativi.

Limitazioni

Tuttavia, è essenziale considerare anche le limitazioni associate agli *IDS* e agli *IPS*:

- **Falsi Positivi:** Una delle sfide più comuni è il rischio di falsi positivi. Questi si verificano quando il sistema identifica erroneamente attività normale come minacciosa. I falsi positivi possono causare allarmi inutili e richiedere risorse per la loro verifica.

- **Complessità delle Regole:** La gestione delle regole di rilevamento può diventare complessa, specialmente in ambienti tortuosi con molteplici dispositivi e flussi di dati. Mantenere le regole aggiornate e pertinenti è una sfida costante.
- **Impatto sulla Rete:** L'analisi dettagliata del traffico da parte degli IDS e la prevenzione attiva da parte degli IPS possono generare un sovraccarico della rete, specialmente in situazioni di alto traffico. Questo può rallentare le operazioni normali.
- **Costi Operativi:** L'implementazione e la gestione di IDS e IPS richiedono risorse umane e finanziarie. Gli investimenti in hardware, software e personale specializzato possono essere significativi.
- **Necessità di Aggiornamenti:** Per rimanere efficaci, IDS e IPS richiedono aggiornamenti costanti delle firme e delle regole per adattarsi alle nuove minacce. La mancata manutenzione può ridurre l'efficacia nel tempo.

Nonostante queste limitazioni, gli IDS e gli IPS rimangono una componente critica della strategia di sicurezza cibernetica di molte organizzazioni. Il bilancio tra vantaggi e limitazioni dipende dalle esigenze specifiche dell'organizzazione, dalla sua infrastruttura e dalla natura delle minacce affrontate. Una corretta progettazione e gestione di questi sistemi può contribuire in modo significativo a rafforzare la sicurezza informatica.

Conclusioni

Nel corso della ricerca sono state esplorate diverse tecniche per il rilevamento e il blocco del traffico cifrato malevolo. Si è cercato di sfruttare le potenzialità di ciascuna di queste tecniche per raggiungere l'obiettivo di creare un firewall intelligente e innovativo.

Dopo un'attenta valutazione di queste tecniche, si è giunti alla conclusione che l'applicazione e l'utilizzo appropriato del ja3 rendono l'implementazione del firewall più corretta e semplice. La sua capacità di estrarre informazioni significative dalle estensioni SSL/TLS ha permesso di identificare con precisione le applicazioni coinvolte nelle comunicazioni e di bloccare il traffico malevolo in modo efficace. Inoltre, la mancanza di implementazioni consolidate di ja3 lo rende un approccio innovativo nel campo della sicurezza delle reti.

2.4 Firewall avanzati e tecnologie di supporto

Nella ricerca è stata prestata particolare attenzione a due tecnologie di supporto fondamentali: eBPF (Extended Berkeley Packet Filter) e Scapy. Entrambe queste tecnologie offrono approcci diversi ma complementari per la programmazione di un firewall efficace. Questi due strumenti hanno consentito l'esplorazione di approcci innovativi per il rilevamento e il blocco del traffico cifrato malevolo. Nelle sezioni successive, verrà approfondito il funzionamento e le applicazioni di entrambe le tecnologie, evidenziando come possono essere integrate per realizzare un firewall avanzato e altamente efficace.

2.5 eBPF

2.5.1 Introduzione a eBPF

eBPF (Extended Berkeley Packet Filter) è una tecnologia estensibile e versatile che offre la possibilità di eseguire codice sicuro e performante all'interno del kernel Linux. Sebbene sia stato inizialmente sviluppato per il filtraggio di pacchetti di rete, eBPF ha visto un'ampia evoluzione e si è trasformato in uno strumento potente anche per il monitoraggio e il controllo del kernel.

La chiave del successo di eBPF risiede nella sua capacità di eseguire in modo sicuro del codice utente all'interno del kernel, senza comprometterne la stabilità o la sicurezza del sistema. Ciò è reso possibile dal fatto che eBPF consente di scrivere programmi in un linguaggio a basso livello e sicuro, che viene poi tradotto in istruzioni nativamente eseguibili dal kernel. Questo codice può essere caricato e scaricato dinamicamente nel kernel, consentendo di adattare il comportamento del sistema in tempo reale.

Oltre all'ambito delle reti, eBPF ha trovato applicazioni in diversi altri contesti. Ad esempio, è stato utilizzato per il monitoraggio delle prestazioni, la sicurezza del sistema, l'analisi del traffico di rete, l'analisi del comportamento del kernel e persino l'analisi e il tracciamento delle applicazioni. La sua flessibilità e la capacità di eseguire codice sicuro e performante lo rendono una tecnologia preziosa anche al di fuori del contesto delle reti, permettendo di estendere e personalizzare il comportamento del kernel in modo innovativo e sicuro.

2.5.2 Applicazione di eBPF

eBPF offre interessanti possibilità per il controllo del traffico di rete, inclusi pacchetti cifrati, consentendo di analizzare e prendere decisioni sulle comunicazioni in modo efficiente e sicuro. Alcune delle applicazioni potenziali di eBPF per la creazione di un firewall avanzato che blocchi le minacce cifrate includono:

Monitoraggio del Traffico

L'applicazione di eBPF nel monitoraggio del traffico rappresenta una delle sue funzionalità più potenti e versatili. Con eBPF, è possibile implementare un sistema di monitoraggio del traffico di rete in tempo reale estremamente efficiente ed efficace, in grado di osservare con attenzione le comunicazioni cifrate e di identificare comportamenti sospetti o attività malevole.

Una delle sfide più significative nel monitoraggio del traffico di rete è la gestione delle comunicazioni cifrate, che sono diventate sempre più comuni per garantire la sicurezza e la privacy delle informazioni trasmesse. Ecco come eBPF affronta questa sfida:

1. **Osservazione in Tempo Reale:** eBPF consente di monitorare il traffico di rete in tempo reale, catturando pacchetti cifrati man mano che attraversano l'interfaccia di rete. Questo processo avviene senza notevoli ritardi o overhead, garantendo che tutte le comunicazioni, incluse quelle crittografate, possano essere monitorate in modo tempestivo.
2. **Identificazione dei Flussi Crittografati:** Utilizzando eBPF, è possibile identificare e tenere traccia dei flussi di comunicazione crittografati. Questo significa che il sistema è in grado di riconoscere quali comunicazioni sono protette da crittografia e quali no.
3. **Privacy Preservata:** Un aspetto cruciale del monitoraggio del traffico cifrato è la preservazione della privacy degli utenti coinvolti. Con eBPF, è possibile analizzare il traffico senza decrittare i dati, evitando qualsiasi compromissione della privacy. In altre parole è possibile osservare il comportamento del traffico cifrato senza accedere al suo contenuto, rispettando così le normative sulla privacy e la sicurezza dei dati.
4. **Rilevamento di Comportamenti Sospetti:** Oltre a identificare le comunicazioni crittografate, eBPF consente di analizzare i flussi di traffico e di individuare comportamenti sospetti o potenzialmente dannosi. Questo può includere

pattern di comunicazione insoliti, tentativi di accesso non autorizzati o comunicazioni con indirizzi IP noti per essere associati a minacce. Il sistema può generare avvisi o intraprendere azioni preventive in base a queste analisi.

Come descritto nell'articolo "Control Plane Enabling Automated and Fully Adaptive Network Traffic Monitoring With eBPF"[7], la metodologia di monitoraggio del traffico basata su eBPF offre un approccio avanzato per affrontare le sfide legate al monitoraggio delle reti, inclusa la gestione delle comunicazioni cifrate. Questo articolo presenta un'architettura di piano di controllo che consente il monitoraggio automatizzato e altamente adattivo del traffico di rete, fornendo un quadro completo per comprendere i progetti e le considerazioni di progettazione dietro a tale metodologia.

L'articolo propone un'architettura in cui il piano di dati e il piano di controllo sono decouplati, consentendo un monitoraggio efficiente e flessibile. In particolare, l'articolo sottolinea l'importanza di supportare diverse nature di monitoraggio, tra cui l'analisi incrementale e basata su finestre temporali nonché la coerenza dei dati raccolti. Questa metodologia consente agli utenti di definire metriche di monitoraggio personalizzate e di raccogliere dati in modo coerente ed efficiente.

Un aspetto cruciale evidenziato nell'articolo è la flessibilità offerta dalla programmabilità eBPF, che consente la sostituzione dinamica di programmi di monitoraggio, l'aggiornamento delle funzionalità di monitoraggio e la definizione di strutture dati personalizzate. Ciò contribuisce in modo significativo alla versatilità del sistema di monitoraggio e alla sua capacità di adattarsi alle esigenze specifiche.

Analisi del Contenuto Criptato

L'analisi del contenuto crittato è una delle applicazioni più potenti di eBPF nel campo del monitoraggio del traffico di rete. La crittografia è comunemente utilizzata per proteggere i dati da occhi indiscreti, ma con eBPF, è possibile eseguire un'analisi avanzata del traffico cifrato senza compromettere la privacy dei dati trasmessi.

Le seguenti sono alcune delle considerazioni chiave nell'analisi del contenuto crittato con eBPF:

1. **Pattern e Firme Riconoscibili:** Nonostante la crittografia protegga il contenuto dei pacchetti di rete, è possibile identificare pattern o firme riconoscibili nel traffico cifrato. Questi pattern possono essere associati a comportamenti sospetti o attività malevole. Ad esempio, determinati tipi di malware o attacchi

potrebbero generare schemi di comunicazione specifici che possono essere rilevati attraverso l'analisi dei metadati del traffico, come le dimensioni dei pacchetti, le frequenze di trasmissione, gli indirizzi IP coinvolti e altro ancora.

2. **Privacy Preservata:** È importante sottolineare che l'analisi del contenuto crittato con eBPF avviene senza la decrittazione dei dati. Ciò significa che il contenuto effettivo dei pacchetti rimane inaccessibile, preservando la privacy degli utenti coinvolti. L'analisi si concentra sui metadati e sul comportamento del traffico, evitando qualsiasi violazione della crittografia stessa.
3. **Rilevamento di Anomalie:** eBPF può essere utilizzato per individuare anomalie o comportamenti fuori dal comune nel traffico cifrato. Ciò include la capacità di riconoscere attività insolite o tentativi di accesso non autorizzati. Ad esempio, se un'applicazione crittografata inizia a comportarsi in modo anomalo, inviando un volume eccessivo di dati o comunicando con indirizzi IP sospetti, eBPF può rilevare queste anomalie e generare avvisi o intraprendere azioni preventive.
4. **Integrazione con Firme Conosciute:** Inoltre, eBPF può essere integrato con database di firme conosciute di minacce. Ciò consente di confrontare il traffico cifrato con firme di malware o attacchi noti, consentendo di identificare e bloccare attivamente le minacce all'interno delle comunicazioni cifrate.

L'analisi del contenuto crittato con eBPF rappresenta una potente aggiunta alle capacità di sicurezza delle reti, consentendo di individuare potenziali minacce e comportamenti sospetti all'interno delle comunicazioni cifrate. La sua capacità di farlo senza decrittare i dati stessi è fondamentale per preservare la privacy e la sicurezza delle informazioni trasmesse attraverso la rete.

Filtraggio e Blocco delle Minacce

Il filtraggio e il blocco delle minacce all'interno del traffico cifrato rappresentano una delle applicazioni più rilevanti e potenti di un firewall basato su eBPF. Questa capacità consente di creare un efficace scudo di protezione per le reti, impedendo che minacce malevole penetrino all'interno delle comunicazioni crittografate. Le seguenti considerazioni sono fondamentali per comprendere questa applicazione:

1. **Personalizzazione delle Regole:** Con eBPF, è possibile definire regole e criteri di sicurezza altamente personalizzati. Ciò significa che è possibile specificare

in dettaglio quali tipi di pacchetti o comportamenti devono essere considerati minacce e quali devono essere permessi. Ad esempio, si possono definire regole per bloccare il traffico proveniente da indirizzi IP sospetti, pacchetti con firme di malware o comunicazioni che superano determinate soglie di attività sospetta.

2. **Rilevamento delle Minacce:** Il firewall basato su eBPF può integrare database di firme conosciute di minacce, consentendo il rilevamento attivo di malware, attacchi noti o comportamenti malevoli. Quando un pacchetto cifrato corrisponde a una firma nota, il firewall può immediatamente bloccare la comunicazione e prendere azioni preventive.
3. **Prevenzione degli Attacchi Interni:** Un vantaggio significativo del filtraggio all'interno del traffico cifrato è la prevenzione degli attacchi interni. Anche se una rete dispone di misure di sicurezza perimetrale, gli attacchi possono verificarsi all'interno della rete stessa. Un firewall basato su eBPF può identificare e bloccare attività sospette o non autorizzate all'interno delle comunicazioni crittografate, proteggendo così la rete dagli attacchi interni.
4. **Protezione della Privacy:** È importante notare che il filtraggio e il blocco delle minacce con eBPF avvengono senza la decrittazione dei dati effettivi. Ciò significa che il contenuto crittato delle comunicazioni rimane privato e sicuro. Il firewall si basa sulle informazioni disponibili nei metadati e sulle firme di minacce per identificare e bloccare le minacce.
5. **Risposta in Tempo Reale:** Grazie alla capacità di eBPF di eseguire codice in tempo reale nel kernel, il firewall può rispondere immediatamente alle minacce identificate. Questo comporta una risposta pronta e una protezione continua contro le minacce nel traffico cifrato.

In sintesi, il filtraggio e il blocco delle minacce all'interno del traffico cifrato con un firewall basato su eBPF forniscono una robusta difesa contro le minacce malevole, garantendo al contempo la privacy delle comunicazioni. Questa applicazione è essenziale per proteggere le reti da attacchi interni ed esterni, offrendo un livello aggiuntivo di sicurezza per le comunicazioni crittografate.

Analisi del Comportamento delle Applicazioni

Con eBPF è possibile analizzare il comportamento delle applicazioni che utilizzano la crittografia, come client di messaggistica o applicazioni di file sharing. Questa analisi

può rivelare comportamenti anomali o tentativi di accesso non autorizzati, consentendo di individuare e isolare le minacce all'interno delle comunicazioni cifrate. La comprensione del comportamento delle applicazioni è fondamentale per l'identificazione delle attività sospette e per la protezione dei dati e delle risorse del sistema.

L'applicazione di eBPF nell'ambito della sicurezza delle reti offre una serie di vantaggi, tra cui elevata efficienza, flessibilità e sicurezza. Tuttavia, è importante considerare anche le sfide legate alla complessità dell'implementazione, al supporto limitato in alcuni kernel Linux e alle questioni di privacy associate all'analisi del traffico crittato. Nonostante queste sfide, eBPF rimane una tecnologia promettente per la creazione di un firewall avanzato che possa bloccare minacce cifrate in modo efficiente e sicuro. L'obiettivo è sfruttare a pieno il potenziale di eBPF e altre tecniche innovative per fornire una soluzione efficace nella lotta contro le minacce nel traffico cifrato.

2.5.3 Vantaggi e Limitazioni

L'utilizzo di eBPF per implementare un firewall avanzato offre numerosi vantaggi, tra cui:

- **Elevata Efficienza:** eBPF permette di eseguire codice in modo nativo all'interno del kernel, garantendo prestazioni elevate e riducendo l'overhead.
- **Flessibilità:** La capacità di caricare e scaricare dinamicamente il codice in tempo reale consente di adattare il comportamento del firewall alle esigenze specifiche dell'ambiente di rete.
- **Sicurezza:** eBPF offre un ambiente sicuro per l'esecuzione del codice utente all'interno del kernel, riducendo il rischio di vulnerabilità e crash del sistema.

Tuttavia, è importante sottolineare alcune limitazioni di eBPF per l'analisi del traffico cifrato:

- **Complessità:** L'utilizzo di eBPF richiede una comprensione approfondita del kernel Linux e delle funzionalità di rete, rendendo la sua implementazione più complessa rispetto ad altre tecniche.
- **Supporto Limitato:** Non tutti i kernel Linux supportano pienamente eBPF, e alcune distribuzioni potrebbero richiedere configurazioni aggiuntive per abilitare questa funzionalità.

- **Privacy:** L'analisi del traffico crittato potrebbe sollevare questioni di privacy, poiché potrebbe essere possibile identificare il contenuto delle comunicazioni nonostante la crittografia.

Nonostante queste limitazioni, eBPF rimane una tecnologia promettente per la creazione di un firewall avanzato che possa bloccare minacce cifrate in modo efficiente e sicuro. L'obiettivo è sfruttare a pieno il potenziale di eBPF e altre tecniche innovative per fornire una soluzione efficace nella lotta contro le minacce nel traffico cifrato.

2.6 Utilizzo di Librerie Custom

Nel contesto dell'analisi e del filtraggio dei pacchetti di rete, diventa essenziale fare affidamento su librerie specializzate per ottenere un controllo completo e flessibile sul traffico. In questa sezione, verrà esaminato l'utilizzo di librerie personalizzate che consentono di manipolare e interagire con i pacchetti di rete in modo altamente configurabile.

Una delle librerie personalizzate ampiamente utilizzate per l'analisi e la manipolazione dei pacchetti di rete è Scapy. Scapy è una libreria Python che offre una vasta gamma di funzionalità per l'interazione con i pacchetti di rete. Grazie alla sua natura user-space, Scapy rappresenta una soluzione alternativa all'utilizzo di eBPF per l'analisi e il filtraggio dei pacchetti di rete. Utilizzando Scapy, è possibile implementare un firewall in grado di analizzare e prendere decisioni sui pacchetti di rete in un ambiente controllato e altamente programmabile.

Scapy consente di effettuare operazioni avanzate di analisi dei pacchetti, inclusi il filtraggio del traffico in base a criteri personalizzati e la presa di decisioni flessibili in tempo reale. Di seguito, saranno esaminati alcuni passaggi chiave per sfruttare Scapy nella creazione di un firewall:

Cattura dei Pacchetti

Con Scapy, è possibile catturare pacchetti in ingresso ed uscita dalla rete, ottenendo accesso a tutte le informazioni contenute nei pacchetti, come indirizzi IP, porte e dati payload. Questa cattura può essere effettuata in tempo reale o su file di cattura pre-esistenti. La capacità di catturare pacchetti è fondamentale per l'analisi e il monitoraggio del traffico di rete. Scapy offre una soluzione flessibile che consente di acquisire dati direttamente dalla rete, consentendo agli amministratori di rete di

esaminare in dettaglio il flusso di pacchetti che attraversa la rete. La cattura in tempo reale è particolarmente utile per il monitoraggio in tempo reale dell'attività di rete, mentre la cattura da file di cattura pre-esistenti consente l'analisi dei dati archiviati per scopi di debugging o di audit.

Analisi dei Pacchetti

Dopo la cattura, Scapy permette di analizzare i pacchetti utilizzando un'ampia gamma di metodi. Questa fase è cruciale per comprendere il contenuto e il significato dei pacchetti di rete. Gli amministratori di rete possono accedere a campi specifici dei pacchetti, esaminare gli indirizzi IP sorgente e destinazione, le porte di destinazione e altre informazioni rilevanti. Inoltre, Scapy consente di riconoscere pattern o firme rilevanti nei pacchetti, il che è particolarmente utile per l'identificazione di comportamenti anomali o attività malevole. L'analisi dei pacchetti consente agli amministratori di rete di monitorare il traffico in modo approfondito e di individuare eventuali anomalie o minacce alla sicurezza.

Filtraggio dei Pacchetti

Scapy permette di definire regole personalizzate per il filtraggio dei pacchetti. Questo significa che è possibile creare regole specifiche per bloccare o manipolare il traffico in base a criteri definiti dall'utente. Il filtraggio dei pacchetti è uno strumento potente per proteggere una rete da minacce e attacchi. Con Scapy, è possibile creare un firewall ad hoc in grado di prevenire minacce crittografate e altri tipi di attacchi. Le regole di filtraggio possono essere altamente specifiche e personalizzate per soddisfare le esigenze di sicurezza di una rete. Ciò consente agli amministratori di rete di implementare politiche di sicurezza personalizzate e di proteggere la rete da potenziali vulnerabilità.

Gestione delle Decisioni

Un aspetto chiave nella creazione di un firewall con Scapy è la gestione delle decisioni prese sui pacchetti. Scapy offre la flessibilità necessaria per definire il comportamento del firewall in modo dinamico. Questo significa che il firewall può adattarsi alle esigenze specifiche dell'ambiente di rete e prendere decisioni in tempo reale in base alle regole definite. La gestione delle decisioni è fondamentale per garantire che il firewall sia in grado di rispondere alle minacce in modo efficace e di consentire il flusso di traffico legittimo. Scapy offre agli amministratori di rete la possibilità di personalizzare

il comportamento del firewall in modo da soddisfare al meglio le esigenze di sicurezza della loro rete.

2.6.1 Vantaggi e Limitazioni

L'utilizzo di Scapy per implementare un firewall user-space offre alcuni vantaggi distinti rispetto all'uso di eBPF:

- **Filtraggio dei Pacchetti lato User-Space:** Scapy opera a livello utente e non richiede modifiche al kernel. Ciò consente di effettuare il filtraggio dei pacchetti in modo più flessibile e senza la necessità di privilegi di amministrazione.
- **Maggiori Facilità di Programmazione:** Scapy è una libreria di Python con un'interfaccia di programmazione semplice e intuitiva, rendendo più agevole lo sviluppo e la personalizzazione del firewall rispetto alle complesse configurazioni richieste da eBPF.
- **Ampio Supporto per il Protocollo:** Scapy supporta un vasto insieme di protocolli di rete e offre funzionalità di decodifica e codifica avanzate, che consentono un'analisi più approfondita dei pacchetti.
- **Portabilità:** Essendo una libreria di Python, Scapy è altamente portabile e può essere eseguito su diverse piattaforme e sistemi operativi senza modifiche significative.

Tuttavia, vi sono alcune limitazioni nell'utilizzo di Scapy per un firewall:

- **Overhead e Prestazioni:** Scapy opera a livello utente, il che può comportare un maggiore overhead e prestazioni inferiori rispetto all'esecuzione di codice direttamente nel kernel.
- **Sicurezza:** Poiché Scapy è una libreria di Python, il suo utilizzo potrebbe introdurre vulnerabilità legate alla sicurezza della piattaforma Python stessa. È fondamentale prendere misure per garantire che il codice Scapy sia sicuro e protetto dalle minacce.
- **Complessità del Codice:** Anche se Scapy semplifica la manipolazione dei pacchetti, la creazione di un firewall completo richiede la scrittura di un codice più complesso rispetto a soluzioni kernel-based come eBPF.

Un articolo di riferimento che dimostra l'efficacia di Scapy in questo contesto è "A powerful interactive packet manipulation program"[8]. Questo articolo sottolinea come Scapy sia in grado di manipolare e analizzare pacchetti di rete in modo interattivo e offra un'ampia gamma di funzionalità per il monitoraggio del traffico.

In conclusione, Scapy rappresenta un'interessante alternativa a eBPF per la creazione di un firewall che possa bloccare il traffico cifrato malevolo. L'approccio user-space offre maggiori facilità di programmazione e flessibilità, ma potrebbe comportare un'efficienza inferiore rispetto a soluzioni kernel-based. La scelta tra eBPF e Scapy dipenderà dalle specifiche esigenze dell'ambiente di rete e dai requisiti di prestazioni e sicurezza.

Capitolo 3

Progettazione e Implementazione

3.1 Definizione dei Requisiti

Questa sezione fornisce un'analisi dettagliata del quadro dei requisiti e degli obiettivi che guidano il processo di progettazione e implementazione dell'innovativo strumento basato su Scapy. L'obiettivo primario di questo lavoro consiste nella realizzazione di una soluzione avanzata per l'individuazione e la cattura di minacce crittografate all'interno del traffico di rete. Tale obiettivo ampio è scomposto in una serie di requisiti chiave che stabiliscono la direzione del progetto e ne orientano l'implementazione.

3.1.1 Contesto e Motivazione

L'aumento esponenziale dell'adozione della crittografia end-to-end nella comunicazione ha amplificato notevolmente la complessità dell'individuazione di minacce all'interno del traffico di rete. Le minacce crittografate, occultate all'interno di connessioni apparentemente sicure, richiedono un approccio innovativo e sofisticato per essere identificate e neutralizzate. Questo progetto si inserisce in tale contesto, proponendo uno strumento che sfrutta a pieno la potenza della libreria Scapy e la tecnologia JA3 per affrontare in modo efficace questa sfida.

3.1.2 Obiettivi Specifici

L'obiettivo principale di questo progetto è sviluppare un'applicazione versatile e potente per la rilevazione di minacce nel traffico cifrato. Gli obiettivi specifici che guidano la progettazione e l'implementazione dell'applicazione includono:

1. **Monitoraggio del Traffico Crittografato:** L'applicazione deve essere in grado di identificare e analizzare le connessioni cifrate all'interno del traffico di rete.
2. **Estrazione dei Dettagli Crittografici:** È essenziale estrarre con precisione i dettagli crittografici dai pacchetti cifrati, fornendo le informazioni necessarie per la creazione di fingerprint JA3.
3. **Generazione dei Fingerprint JA3:** Utilizzando i dettagli estratti, l'applicazione deve generare fingerprint JA3 distintivi per le connessioni cifrate. Questi fingerprint permettono l'identificazione delle connessioni in modo univoco.
4. **Classificazione e Rilevamento:** I fingerprint JA3 saranno confrontati con un database di fingerprint noti associati a minacce. Se vengono rilevate corrispondenze, verranno segnalate possibili minacce e saranno intraprese azioni appropriate, come il blocco della connessione.
5. **Gestione Flessibile delle Blacklist:** L'applicazione consentirà l'aggiunta dinamica di fingerprint e indirizzi IP alla blacklist per rispondere in modo agile alle nuove minacce.
6. **Interazione Utente Intuitiva:** L'applicazione sarà dotata di un'interfaccia utente intuitiva che semplifica la configurazione delle opzioni di analisi e il monitoraggio dei risultati.
7. **Documentazione Completa:** Il progetto includerà una documentazione esaustiva che illustrerà le funzionalità, l'uso e le istruzioni per l'installazione dell'applicazione.

3.1.3 Vincoli e Limitazioni

È essenziale riconoscere che l'analisi del traffico crittografato presenta intrinsecamente sfide e limitazioni. Ad esempio, la decodifica completa dei pacchetti crittografati

potrebbe non essere sempre fattibile, in virtù delle contromisure di sicurezza implementate. Inoltre, l'efficacia dell'applicazione potrebbe variare in base alle tecniche di evasione adottate dagli aggressori. Tuttavia, nonostante tali restrizioni, lo scopo dell'applicazione è quello di fornire un contributo di rilievo all'individuazione di minacce nel traffico dati cifrato.

La prossima sezione esplorerà l'architettura dello strumento, evidenziando le scelte progettuali chiave e come queste affrontano le sfide delineate.

3.2 Architettura

3.2.1 Struttura Funzionale

In questa sezione verrà presentata un'ampia panoramica dell'architettura della soluzione di rilevamento delle minacce crittografate, prima di esaminare le fasi di implementazione specifiche. Questo consentirà di ottenere una visione completa dei componenti chiave che costituiscono il processo di rilevamento delle minacce crittografate. Successivamente, saranno descritti in dettaglio come tali concetti architetturali si traducono in un'applicazione pratica.

Cattura del Traffico

Il primo passo nell'architettura è la cattura in tempo reale del traffico di rete. Questo processo coinvolge la raccolta dei pacchetti di dati che attraversano la rete. Durante la cattura, è essenziale acquisire tutti i pacchetti di rete, compresi quelli che utilizzano connessioni crittografate. L'obiettivo principale è ottenere un flusso di dati rappresentativo del traffico di rete in atto.

Analisi tramite Tecnica JA3

Una volta catturati i pacchetti di rete, si procede con l'analisi utilizzando la tecnica JA3. Questo approccio prevede l'estrazione di dettagli crittografici chiave dai pacchetti TLS/SSL, tra cui la versione del protocollo, le suite di cifratura e le estensioni supportate. Questi dettagli vengono utilizzati per generare fingerprint JA3 distintivi per ciascuna connessione crittografata. Questi fingerprint rappresentano una sorta di "impronta digitale" delle connessioni crittografate e consentono un'identificazione univoca.

Blocco dei Pacchetti Sospetti

Una volta generati i fingerprint JA3, si confrontano questi fingerprint con una blacklist contenente fingerprint noti associati a minacce. Se un fingerprint corrisponde a uno presente nella blacklist, viene segnalata una possibile minaccia. Inoltre, è possibile intraprendere azioni immediate, come il blocco della connessione tramite tecnologie già esistenti come iptables.

In breve, l'architettura si basa sulla cattura del traffico, sull'analisi utilizzando la tecnica JA3 e sul blocco dei pacchetti sospetti. Questi passaggi costituiscono la base del futuro strumento di rilevamento delle minacce crittografate.

Diagramma del Processo

Di seguito è rappresentato un diagramma del processo di rilevamento delle minacce crittografate:

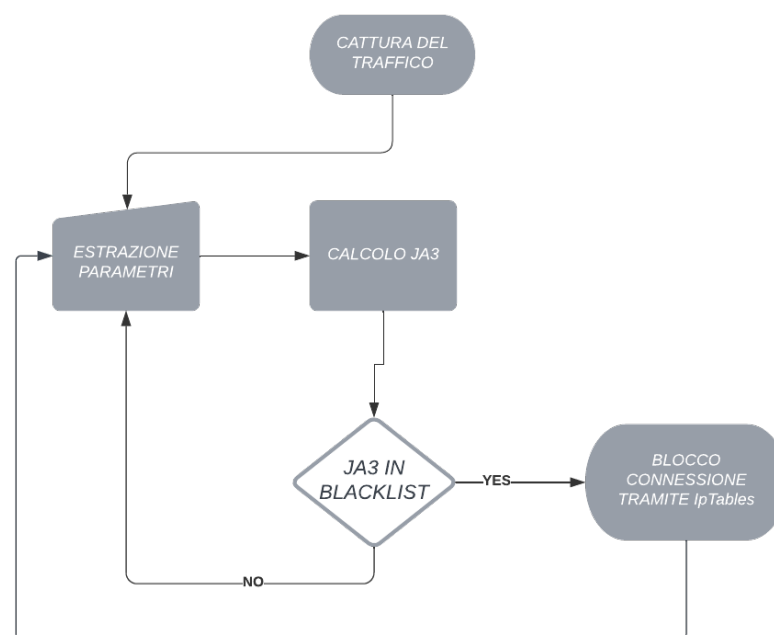


Figura 3.2.1: Diagramma del Processo di Rilevamento delle Minacce Crittografate

Il diagramma fornisce una panoramica chiara delle fasi del processo, offrendo una visione d'insieme prima di entrare nei dettagli implementativi.

3.2.2 Organizzazione Modulare

L'architettura dell'applicazione è stata concepita secondo un approccio modulare che facilita la suddivisione dei vari aspetti funzionali. L'organizzazione generale dell'applicazione comprende i seguenti moduli:

1. **Cattura del Traffico:** L'acquisizione dei dati è una fase cruciale del processo. Attraverso l'utilizzo della libreria Scapy, l'applicazione è in grado di intercettare i pacchetti in transito sulla rete, preparandoli per l'analisi successiva al fine di rilevare minacce criptate.
2. **Estrazione dei Dettagli Crittografici:** Questo modulo si focalizza sull'estrazione accurata dei dettagli crittografici dai pacchetti TLS/SSL catturati. I dettagli estratti includono la versione del protocollo, le suite di cifratura e le estensioni supportate.
3. **Generazione dei Fingerprint JA3:** Basandosi sui dettagli crittografici estratti, il modulo di generazione dei fingerprint JA3 crea fingerprint univoci per ciascuna connessione crittografata. Questi fingerprint costituiscono la firma digitale delle connessioni e rappresentano l'elemento centrale del sistema di rilevamento.
4. **Confronto con la Blacklist:** In questa fase, i fingerprint JA3 generati sono confrontati con una blacklist contenente fingerprint noti di minacce. Se si verifica una corrispondenza, l'applicazione segnala la possibile presenza di una minaccia.
5. **Gestione della Blacklist:** Questo modulo permette agli operatori di gestire in modo dinamico le blacklist JA3 e IP. Gli operatori possono aggiungere nuovi fingerprint e indirizzi IP alla blacklist, assicurando una risposta agile alle minacce emergenti.
6. **Azione di Blocco:** Nel caso in cui una minaccia venga identificata, il sistema può intraprendere azioni di blocco, come il rifiuto della connessione o l'aggiunta dell'indirizzo IP alla blacklist IP. Tali azioni sono mirate a contrastare tempestivamente la minaccia.

In questa sezione, è stato fornito un contesto generale sull'architettura del sistema di rilevamento delle minacce crittografate prima di esaminare nel dettaglio le fasi implementative.

3.3 Implementazione

In questa sezione, verranno presentati diversi frammenti di codice del tool per il rilevamento delle minacce crittografate. L'implementazione è stata sviluppata in Python e fa ampio uso di librerie specializzate per l'analisi dei pacchetti di rete, la manipolazione di dati crittografici e la gestione delle blacklist.

3.3.1 Inizializzazione e Configurazione

In questa sezione, verrà esaminato il processo di inizializzazione del tool e la configurazione dei parametri tramite l'utilizzo delle librerie Python necessarie e l'analisi delle opzioni fornite dalla linea di comando.

Importazione delle Librerie

Tra le varie librerie utilizzate, le più rilevanti sono sicuramente:

- **subprocess**: Questa libreria è fondamentale per l'esecuzione di processi esterni. Nel contesto del tool, viene utilizzata per implementare il blocco delle connessioni indesiderate. È una componente cruciale per garantire la sicurezza del sistema, consentendo di interrompere immediatamente le connessioni sospette o pericolose.
- **scapy.utils** e **scapy.all**: Scapy è una libreria essenziale per l'analisi dei pacchetti di rete. **scapy.utils** è utilizzato per la scrittura di file di tipo pcap, che consentono di registrare il traffico di rete per ulteriori analisi. **scapy.all** offre funzionalità di base per l'intercettazione e l'elaborazione dei pacchetti. Queste librerie sono centrali nella capacità di esaminare e comprendere il traffico crittografato, consentendo di identificare potenziali minacce o comportamenti anomali.

L'uso di queste librerie è fondamentale per il funzionamento del tool e per la sua capacità di analizzare e proteggere il traffico cifrato all'interno della rete.

Funzioni Ausiliarie

Sono definite diverse funzioni ausiliarie che contribuiscono al funzionamento del tool:

- **get_attr:** Questa funzione consente di ottenere un attributo da un oggetto, fornendo un valore predefinito nel caso in cui l'attributo non sia presente. Ad esempio, viene utilizzata per estrarre attributi dai pacchetti di rete analizzati.
- **timer_unit:** Trasforma un intervallo di tempo in unità leggibili come secondi, minuti, ore o giorni. Questa funzione è utilizzata per formattare il tempo impiegato nell'analisi dei pacchetti.
- **put_color:** Utilizzata per colorare il testo in console con una determinata tonalità. Questo migliora la visualizzazione dei messaggi di output evidenziando informazioni importanti o di interesse.

Queste funzioni ausiliarie contribuiscono a rendere il codice più leggibile, organizzato e manutenibile, migliorando l'esperienza dell'utente nell'uso del tool.

3.3.2 Funzioni di Analisi e Rilevamento

Questa sezione esamina le funzioni di analisi dei pacchetti e di rilevamento delle minacce crittografate implementate nel tool. Il frammento di codice sottostante illustra il processo di analisi dei pacchetti intercettati e la rilevazione delle potenziali minacce.

Funzione per l'Analisi dei Pacchetti

La funzione `collector(pkt)` costituisce il cuore del processo di analisi dei pacchetti. Di seguito viene fornita una panoramica delle principali fasi della funzione:

- **Inizializzazione delle Variabili:** La funzione inizia incrementando il contatore globale dei pacchetti `COUNT`. Se l'opzione di salvataggio dei pacchetti raw è abilitata, la funzione scrive il pacchetto nel file di output. Viene anche visualizzato un messaggio di stato in console per indicare che il tool è in esecuzione.
- **Estrazione dei Livelli:** Vengono estratti i livelli del pacchetto utilizzando le funzioni di `getlayer()` per il livello TCP e IP. Vengono quindi recuperati gli indirizzi IP sorgente e destinazione, così come le porte sorgente e destinazione.
- **Analisi Crittografica:** La funzione esegue un'analisi crittografica del pacchetto per identificare le informazioni rilevanti. Viene utilizzata la funzione `get_attr()` per estrarre il messaggio crittografico dal livello TCP del pacchetto. Questo messaggio conterrà dettagli importanti per l'analisi successiva.

- **Verifica delle Corrispondenze:** La funzione verifica se il messaggio crittografico ottenuto corrisponde a criteri specifici, come la presenza nella lista nera `blacklist`. In caso di corrispondenza, la connessione sospetta viene bloccata utilizzando `iptables`. Inoltre, vengono eseguite ulteriori verifiche per rilevare possibili minacce.

si può notare come la funzione `is_matching_ja3` controlla se una specifica fingerprint corrisponde alla blacklist JA3

```
def is_matching_ja3(ja3):  
    if ja3 in blacklist:  
        return True  
    return False
```

Mentre la funzione `block_connection` è responsabile del blocco di una connessione utilizzando il comando `iptables`. Questa funzione prende un indirizzo IP (`src_ip`) come argomento e aggiunge una regola alla catena `INPUT` del firewall `iptables` per bloccare i pacchetti provenienti da quell'indirizzo IP. Ecco l'implementazione:

```
def block_connection(src_ip):  
    subprocess.run(["iptables", "-A", "INPUT", "-s", src_ip, "-j", "DROP"])
```

- **Output dei Risultati:** I risultati dell'analisi vengono formattati e visualizzati in console o salvati in un file, a seconda delle opzioni selezionate. Vengono inclusi dettagli come gli indirizzi IP, le porte, le informazioni crittografiche estratte e lo stato di corrispondenza con le liste nere.

Impostazione dei Parametri di `sniff()`

Una corretta configurazione dei parametri di `'sniff()'` è cruciale per il corretto funzionamento del tuo tool di rilevamento delle minacce crittografate. Qui di seguito verranno esaminati in dettaglio i parametri utilizzati e come essi contribuiscono all'efficacia del tool:

- **prn**: La funzione di callback (**collector**) da eseguire per ogni pacchetto catturato. Questo parametro è fondamentale in quanto consente di specificare quale funzione deve essere chiamata per l'analisi di ogni pacchetto catturato. La funzione 'collector(pkt)' è responsabile di estrarre, analizzare e verificare i pacchetti crittografati alla ricerca di minacce. Questo approccio consente di mantenere il codice organizzato e separa chiaramente l'analisi del pacchetto dalla cattura stessa.
- **filter**: Il filtro (**bpf**) per catturare i pacchetti di interesse. Nel tool, questo filtro svolge un ruolo chiave nell'identificazione dei pacchetti rilevanti per l'analisi crittografica. Il filtro può essere configurato in modo da catturare specifici tipi di pacchetti, come quelli associati a protocolli crittografati o a specifiche porte di rete. Ad esempio, per catturare solo i pacchetti TLS, il filtro potrebbe essere configurato come "port 443". Questo assicura che il tool concentri la sua attenzione solo sui pacchetti crittografati di interesse.
- **store**: Questo parametro è impostato a 0 per evitare di memorizzare i pacchetti in memoria. In un contesto di analisi di pacchetti di rete in tempo reale, è importante minimizzare l'utilizzo della memoria, specialmente quando il tool è in esecuzione per lunghi periodi di tempo. Impostando 'store' a 0, i pacchetti non vengono memorizzati in una lista in memoria, il che consente al tool di gestire grandi quantità di traffico senza esaurire la memoria del sistema.
- **iface**: L'interfaccia di rete (**iface**) da cui catturare i pacchetti, o **None** per catturare da tutte le interfacce. Questo parametro offre flessibilità nella scelta dell'interfaccia di rete da monitorare. In un ambiente con più interfacce di rete, è possibile specificare l'interfaccia desiderata per concentrare l'analisi su un segmento di rete specifico. D'altra parte, l'opzione 'None' consente di catturare pacchetti da tutte le interfacce disponibili, ideale per un'analisi completa del traffico di rete.

La configurazione accurata di questi parametri in **sniff()** garantisce che il tuo tool sia in grado di catturare e analizzare con successo i pacchetti crittografati, consentendo di identificare e affrontare minacce potenziali in tempo reale.

3.3.3 Avvio e Statistiche

Questa sezione fornisce un'analisi approfondita del processo di avvio del tool per il rilevamento delle minacce crittografate e dell'importanza delle statistiche nell'analisi del traffico di rete crittografato.

Gestione degli Argomenti da Linea di Comando

L'avvio del tool inizia con la gestione degli argomenti da linea di comando, un passaggio cruciale per configurare e personalizzare il comportamento del tool. La libreria `argparse` è utilizzata per definire e analizzare gli argomenti forniti dall'utente durante l'esecuzione del programma. Questo permette agli utenti di specificare parametri come l'interfaccia di rete da monitorare, le opzioni di output e altri parametri rilevanti.

La gestione degli argomenti da linea di comando aumenta la flessibilità e l'usabilità del tool, consentendo agli utenti di adattarlo alle proprie esigenze specifiche.

Inizializzazione e Avvio dell'Analisi dei Pacchetti

Dopo aver analizzato gli argomenti forniti dall'utente, il tool procede con l'inizializzazione e la configurazione delle impostazioni necessarie. Questa fase include l'inizializzazione di moduli come `Colorama` per la gestione dei colori in console e l'importazione delle librerie essenziali, come `scapy` e `subprocess`. Inoltre, vengono definite funzioni ausiliarie cruciali per l'analisi crittografica e la manipolazione dei pacchetti.

Un passo significativo è il caricamento del modulo TLS tramite la funzione `load_layer()`. Questo modulo è fondamentale per l'analisi crittografica dei pacchetti e consente al tool di identificare minacce potenziali all'interno delle comunicazioni cifrate.

Inoltre, il tool registra un timestamp di inizio (`start_ts`), che sarà utilizzato successivamente per calcolare la durata dell'analisi.

Esecuzione dell'Analisi e Raccolta delle Statistiche

La fase successiva coinvolge l'esecuzione dell'analisi dei pacchetti in tempo reale utilizzando la funzione `sniff()`. Questo passaggio è fondamentale per rilevare e analizzare il traffico di rete crittografato. I parametri precedentemente configurati, tra cui la funzione di callback `collector` per l'analisi dei pacchetti e il filtro di cattura, vengono utilizzati per catturare e processare i pacchetti in arrivo.

Durante questa fase, il tool è in grado di identificare e analizzare pacchetti contenenti messaggi crittografati come `Client Hello` e `Server Hello`.

In caso di eccezioni o errori durante l'analisi, il tool è in grado di gestirli e fornire all'utente un feedback adeguato, garantendo che il processo di rilevamento delle minacce continui senza interruzioni gravi.

Calcolo e Visualizzazione delle Statistiche

Un aspetto cruciale dell'analisi del traffico crittografato è la raccolta e la presentazione di statistiche significative. Una volta completata l'analisi, il tool calcola e presenta le seguenti informazioni:

- Il numero totale di pacchetti catturati (`COUNT`), che fornisce una panoramica del volume di traffico esaminato.
- Il numero di pacchetti contenenti il messaggio `Client Hello` (`COUNT_CLIENT`), che indica il numero di richieste di connessione iniziali.
- Il numero di pacchetti contenenti il messaggio `Server Hello` (`COUNT_SERVER`), che indica il numero di risposte ai messaggi `Client Hello`.
- La durata totale dell'analisi, espressa in un formato leggibile, calcolata sottraendo il timestamp di inizio (`start_ts`) dal timestamp di fine (`end_ts`). Questo dato fornisce un'indicazione del tempo necessario per l'analisi del traffico.

Le statistiche sono presentate in console in un formato chiaro e leggibile, consentendo agli utenti di valutare rapidamente l'andamento dell'analisi e identificare eventuali anomalie o minacce crittografate rilevate.

Messaggio di Conclusione

Alla fine del processo di analisi e presentazione delle statistiche, il tool conclude l'esecuzione con un messaggio di congedo. Questo messaggio può variare tra diverse opzioni di saluto, aggiungendo un tocco amichevole e professionale alla conclusione dell'esecuzione del tool.

In sintesi, questa sezione fornisce una panoramica completa del processo di avvio, analisi e presentazione delle statistiche del tuo tool per il rilevamento delle minacce crittografate, evidenziando l'importanza di ogni fase nel garantire la sicurezza della rete e la rilevazione tempestiva di potenziali minacce.

3.3.4 Conclusioni

Le porzioni di codice presentate offrono una panoramica dell'implementazione dello strumento di rilevamento delle minacce crittografate. L'integrazione di funzioni avanzate e la manipolazione dei dati crittografici costituiscono la base per l'identificazione e la prevenzione di potenziali attacchi all'interno del traffico cifrato.

L'implementazione di questo strumento rappresenta un rilevante apporto alla sicurezza delle reti informatiche. La generazione dei fingerprint JA3 e l'analisi dei pacchetti consentono di individuare tempestivamente minacce all'interno del traffico cifrato, contribuendo efficacemente alla prevenzione e mitigazione di possibili attacchi. L'utilizzo di approcci avanzati, come la creazione di firme digitali basate su dettagli crittografici, si configura come una strategia efficace e innovativa nell'affrontare le crescenti sfide della sicurezza informatica.

In chiusura, lo strumento offre un'analisi dettagliata dei protocolli crittografici e costituisce un solido fondamento per affrontare le minacce emergenti nell'ambito della sicurezza digitale. La sua capacità di individuare in modo proattivo possibili vulnerabilità all'interno del traffico cifrato riveste un ruolo cruciale nel garantire la solidità e l'integrità delle reti informatiche in contesti reali.

Capitolo 4

Fase Sperimentale

4.1 Introduzione

In questo capitolo sono stati presentati e discussi i dettagli della fase sperimentale del progetto. La fase sperimentale è stata suddivisa in due parti principali: la valutazione della capacità di carico sostenibile e l'analisi delle prestazioni del tool.

Nella prima parte, verrà esaminata la capacità del tool di gestire carichi di lavoro estremamente elevati attraverso una serie di test. Questi test hanno l'obiettivo di determinare la capacità del tool di gestire diverse condizioni di traffico e carichi di lavoro. Saranno variati diversi parametri, tra cui il numero di voci nella blacklist, la durata del test, il numero di thread e le connessioni per secondo. I risultati di questi test forniranno informazioni cruciali sulla scalabilità e l'affidabilità del tool in diverse situazioni. Nella seconda parte dei test, l'attenzione sarà rivolta alla dimostrazione dell'efficacia del tool nell'identificare i client malevoli attraverso l'utilizzo di JA3 e nelle misure di sicurezza da esso implementate. Questa sezione si concentrerà su una narrativa descrittiva piuttosto che su tabelle numeriche.

Verrà esaminato il processo in cui il tool rileva i client potenzialmente dannosi tramite l'analisi delle impronte JA3 e successivamente attua azioni correttive. Quando viene identificata una corrispondenza con una impronta JA3 presente nella blacklist, il tool attiverà meccanismi di blocco utilizzando iptables. Inoltre, l'indirizzo IP del client sarà inserito nella seconda blacklist per impedire ulteriori connessioni indesiderate.

Questo approccio verrà dettagliatamente illustrato, evidenziando come il tool contribuisce alla sicurezza del sistema bloccando attivamente le connessioni provenienti da client malevoli. Saranno discussi i casi di successo e gli scenari in cui il tool ha

dimostrato di essere efficace nel contrastare minacce alla sicurezza.

4.2 Valutazione del Carico Lavorativo

Nella fase sperimentale, sono stati condotti una serie di test per valutare la capacità di carico sostenibile del tool di sicurezza. Questi test avevano l'obiettivo di determinare quanto carico il tool poteva gestire in diverse condizioni e come la variazione della blacklist influisse sul carico sopportabile.

4.2.1 Configurazione Sperimentale

I test sono stati condotti utilizzando due macchine virtuali: una VM client da cui sono stati avviati i test e una VM server che ospitava il tool di sicurezza e un server Apache2. Entrambe le VM erano state configurate con specifiche hardware e software per garantire una riproducibilità accurata dei test.

Specifiche Hardware

- **CPU:** 11th Gen Intel(R) Core(TM) i7-11800H @ 2.30GHz
- **RAM:** 4GB
- **Storage:** 25GB
- **Scheda di Rete:** Intel PRO/1000 MT Desktop (Scheda con bridge, Killer(R) Wi-Fi 6 AX1650i 160MHz Wireless Network Adapter (2011NGW))

Specifiche Software

- **Sistema Operativo:**
 - **VM Client:** Ubuntu 20.04 LTS (64 bit) con Linux Mint 20.2
 - **VM Server:** Ubuntu 20.04 LTS (64 bit) con Linux Mint 20.2

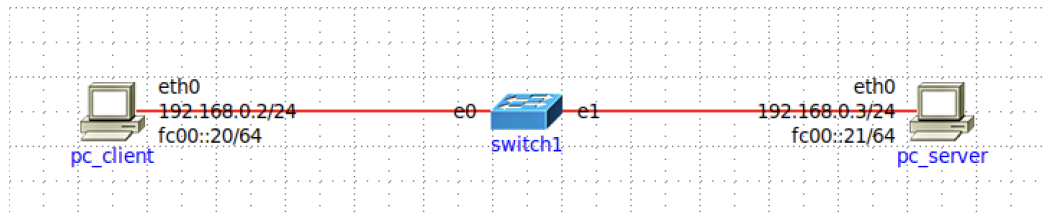


Figura 4.2.1: Topologia di Rete delle Virtual Machine

Per valutare le prestazioni del tool e del server Apache2, è stato utilizzato lo strumento Wrk. Questo strumento è ampiamente utilizzato per testare le applicazioni web e può generare un carico di lavoro significativo sui server. Con Wrk, è stato possibile misurare il numero di richieste al secondo, il tempo medio di risposta e altre metriche importanti.

- **Versione del Tool Wrk:** 1.1.1:OpenSSL Version
- **Versione di Python:** Python 3.10.9
- **Server Web:** Apache/2.4.52 (Ubuntu)

4.2.2 Risultati dei Test

Di seguito sono riportati i risultati dei test di valutazione del carico di lavoro sopportabile. Ogni test è stato identificato da un numero univoco di test e variazioni nella blacklist. Le seguenti metriche sono state registrate per ciascun test:

- **BlackList Entries:** Il numero di voci nella blacklist utilizzato nel test.
- **Test Duration (sec):** La durata del test in secondi.
- **Threads:** Il numero di thread utilizzati per generare il carico di lavoro con "Wrk".
- **Connections:** Il numero totale di connessioni simultanee simulate da "Wrk".
- **Requests/sec:** Il numero di richieste al secondo gestite dal server Apache2.
- **Avg Latency (ms):** La latenza media delle risposte alle richieste.
- **Std Dev Latency (ms):** La deviazione standard della latenza delle risposte, che rappresenta la variabilità dei tempi di risposta tra le richieste.

- Max Latency (ms): La latenza massima registrata durante il test.

Test 1

La linea di comando Wrk iniziale utilizzata per generare il carico di lavoro per questi test è stata la seguente:

```
./wrk -t2 -d60 -c10 https://server_IP:port > test1.txt
```

Dove:

- Il parametro t indica il numero di threads utilizzati per effettuare il test
- Il parametro d indica la durata del test, espressa in secondi
- Il parametro c indica il numero di connessioni simultanee che saranno create durante il test di benchmarking

Successivamente questi parametri sono stati cambiati per aumentare il traffico generato.

Tabella 2: Risultati dei Test 1 di Valutazione del Carico Lavorativo

BLE	Dur. (s)	Thr	Conn	Req/sec	Avg Lat. (ms)	Std Dev Lat. (ms)	Max Lat. (ms)
100	60	2	10	811.14	10.16	6.39	75.87
200	60	2	10	990.29	10.26	6.35	64.69
500	60	2	10	992.19	13.03	25.34	67.49
1000	60	2	10	982.6	10.34	6.50	87.38
2000	60	2	10	987.84	16.72	64.96	96.93
3000	60	2	10	1028.32	10.35	68.96	86.64

Questi primi test forniscono una base per valutare la capacità di carico del tool di sicurezza. Anche se le differenze tra i test sono lievi, si osserva che la modifica della blacklist ha un impatto leggermente aumentato sul carico e sulle prestazioni del tool. Nei prossimi test, verranno esplorati scenari più distanti per comprendere meglio le capacità del sistema di sicurezza.

Test 2

Succesivamente si è optato per un carico di lavoro maggiore, raddoppiando il numero di connessioni, aumentando le entrate della blacklist e andando a sfruttare la possibilità che il tool "Wrk" ha nel far partire script interni per l'invio dei pacchetti. Per rendere il tutto più veloce è stato utilizzato questo semplice script in linguaggio *Lua*:

```
Function request()
    local url= "https://[ServerIp:443]"
    return wrk.format(nil,url)
end

function response (status, headers, body)
end
```

Ottenendo questi risultati:

Tabella 3: Risultati dei Test 2 di Valutazione del Carico Lavorativo

BLE	Dur. (s)	Thr	Conn	Req/sec	Avg Lat. (ms)	Std Dev Lat. (ms)	Max Lat. (ms)
5000	60	2	20	.2445	8.47	5.94	83.33
10000	60	2	20	2335	9.29	13.45	311.73
15000	60	2	20	2418	12.47	42.27	733.56

I risultati evidenziano un aumento significativo delle richieste al secondo e delle connessioni quando il numero di voci nella blacklist viene notevolmente incrementato. In particolare, è emerso che all'aumentare del numero di voci nella blacklist, il sistema mostra un piccolo aumento della latenza, ma sorprendentemente, il throughput, ovvero il numero di richieste gestite al secondo, continua ad aumentare. Questo suggerisce che il tool è in grado di gestire in modo efficiente anche un carico di lavoro più elevato, mantenendo alte prestazioni.

Tuttavia, è importante notare che con l'aumento delle voci nella blacklist, la deviazione standard della latenza e la latenza massima mostrano un aumento significativo. Ciò indica una maggiore variabilità nelle prestazioni, suggerendo che, sebbene il sistema sia in grado di gestire un carico di lavoro più elevato, l'aumento delle voci nella blacklist può comportare ritardi occasionali più significativi.

Test 3

La terza fase di test è incentrata sulla valutazione del carico di lavoro sopportabile in scenari più impegnativi. Verranno ulteriormente variati i parametri, questa volta andando a triplicare la durata del test, al fine di esplorare le capacità del tool di gestire carichi di lavoro estremamente elevati. I risultati di questi test consentiranno di ottenere una comprensione più approfondita delle condizioni limite del sistema e della sua capacità di sostenere carichi di lavoro critici.

Tabella 4: Risultati dei Test 3 di Valutazione del Carico Lavorativo

BLE	Dur. (s)	Thr	Conn	Req/sec	Avg Lat. (ms)	Std Dev Lat. (ms)	Max Lat. (ms)
5000	180	2	20	1074.34	10.34	7.43	120.1
10000	180	2	20	1705.66	12.77	9.91	134.02
15000	180	2	20	2038.86	19.39	13.14	141.91

I risultati dei Test 3, in cui è stata triplicata la durata dei test per esplorare scenari più impegnativi, hanno evidenziato un ulteriore aumento delle richieste al secondo e delle connessioni gestite dal tool. Tuttavia, è importante notare che, con un numero così elevato di voci nella blacklist, si è verificato un significativo aumento della latenza media, della deviazione standard della latenza e della latenza massima.

Il passaggio da 5000 a 10000 voci nella blacklist ha comportato un aumento considerevole del carico di lavoro, con picchi di oltre 1700 richieste al secondo. Questo aumento ha influenzato anche la latenza media, che è leggermente cresciuta, ma ha evidenziato una deviazione standard della latenza significativamente più alta e una latenza massima superiore a 300 ms.

Nel test successivo, con 15000 voci nella blacklist, il carico di lavoro ha continuato ad aumentare, superando le 2000 richieste al secondo. Tuttavia, la latenza media è ulteriormente aumentata, superando i 19 ms, e la deviazione standard della latenza e la latenza massima hanno raggiunto valori significativi.

In generale, questi test hanno dimostrato che, sebbene il tool sia in grado di gestire carichi di lavoro estremamente elevati, l'aumento delle voci nella blacklist ha un impatto significativo sulle prestazioni complessive. In particolare, si osserva un aumento della latenza e della variabilità della latenza, il che suggerisce che il sistema raggiunge il limite delle sue capacità.

I prossimi test si concentreranno su scenari ancora più impegnativi al fine di valutare ulteriormente la resilienza del tool e identificare chiaramente le condizioni limite del sistema.

Test 4

In questa fase conclusiva dei test, l'obiettivo è valutare la capacità di carico di lavoro supportabile del tool in scenari estremamente impegnativi. Sono stati apportati ulteriori aumenti al numero di voci nella blacklist, delle connessioni simultanee e della

durata dei test, al fine di mettere alla prova le capacità del sistema in condizioni critiche.

Tabella 5: Risultati dei Test 4 di Valutazione del Carico Lavorativo

BLE	Dur. (s)	Thr	Conn	Req/sec	Avg Lat. (ms)	Std Dev Lat. (ms)	Max Lat. (ms)
30000	180	2	30	3049	14.31	7.76	1.01
50000	180	2	30	3087	10.44	8.87	109.87
80000	180	2	30	3213	11.44	28.16	158.9
100000	180	2	30	3086	10.52	41.2	109.46

I risultati di questi test confermano la notevole capacità di carico di lavoro sopportabile del tool in scenari estremamente impegnativi. Aumentando il numero di voci nella blacklist, delle connessioni simultanee e della durata dei test, il tool è stato in grado di gestire carichi di lavoro critici.

Nel passaggio da 30000 a 50000 voci nella blacklist, il carico di lavoro è aumentato, con una crescita considerevole delle richieste al secondo. Nonostante ciò, la latenza media è rimasta relativamente bassa, e la deviazione standard della latenza e la latenza massima sono rimaste contenute.

Nel test successivo, con 80000 voci nella blacklist, il carico di lavoro è aumentato ulteriormente, con un picco di richieste al secondo significativo. La latenza media è aumentata leggermente, ma è rimasta accettabile, anche se la deviazione standard della latenza e la latenza massima hanno registrato valori più elevati.

Infine, nel test con 100000 voci nella blacklist, il carico di lavoro è rimasto elevato, ma la latenza media è cresciuta in modo più significativo. Tuttavia, le prestazioni complessive del tool rimangono notevoli, con valori di latenza ancora accettabili.

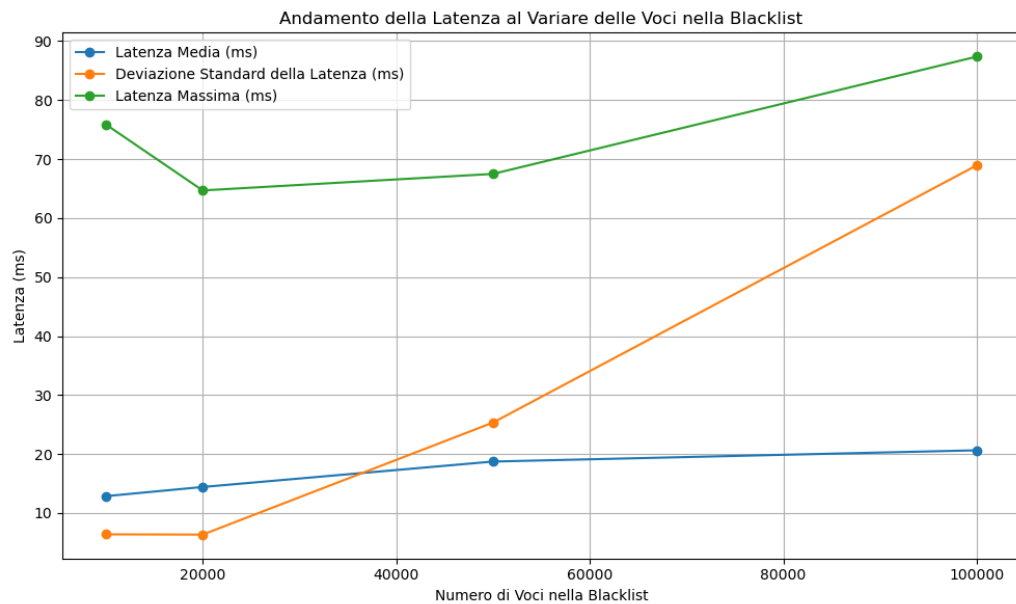


Figura 4.2.2: Andamento della Latenza al Variare delle Voci nella Blacklist

Dal grafico delle prestazioni mostrato in Figura 4.2.2, è possibile osservare le seguenti tendenze:

- La latenza media, rappresentata dalla linea blu, mostra un andamento crescente, ma rimane generalmente bassa, indicando prestazioni accettabili anche sotto carichi di lavoro elevati.
- La deviazione standard della latenza, rappresentata dalla linea arancione, inizia a crescere in modo significativo quando il numero di voci nella blacklist supera i 50.000. Ciò indica una maggiore variabilità nella latenza a carichi di lavoro estremamente elevati.
- La latenza massima, rappresentata dalla linea verde, tende a rimanere stabile fino a 50.000 voci nella blacklist, ma mostra un aumento significativo al di là di questo punto. Questo suggerisce che, sebbene la latenza media rimanga accettabile, possono verificarsi picchi occasionali di latenza più elevata.

Queste osservazioni riflettono i compromessi tra la capacità di carico e le prestazioni del sistema. Pertanto, al selezionare le impostazioni del tool per scenari reali, è fondamentale considerare attentamente questi fattori. Il tool offre una notevole capacità di carico, adatta per scenari in cui è necessario filtrare un elevato numero

di richieste in tempo reale. Tuttavia, è essenziale monitorare attentamente le prestazioni e la latenza, specialmente quando si utilizzano blacklist molto estese, al fine di garantire il funzionamento ottimale del sistema.

4.2.3 Conclusioni

In conclusione, i risultati dei test evidenziano la notevole capacità del tool di gestire carichi di lavoro estremamente elevati, anche con un grande numero di voci nella blacklist. Questa impressionante capacità è resa possibile grazie all'efficiente trasformazione della blacklist in un set all'interno del tool, che consente di effettuare ricerche in modo rapido ed efficiente, mantenendo prestazioni elevate anche con elenchi di dimensioni considerevoli.

4.3 Analisi dell'Efficacia del Tool

Nella seconda parte della fase sperimentale, ci si concentrerà sull'analisi delle prestazioni del tool in termini di rilevamento e blocco delle connessioni indesiderate.

Per condurre una valutazione accurata delle prestazioni del tool, è stata necessaria una selezione oculata dei metodi di test. Data la complessità delle reti crittografate e la diversità delle minacce, è stato fondamentale determinare scenari di test realistici che potessero rappresentare le situazioni più comuni in cui il tool sarebbe stato utilizzato.

4.3.1 Metodologia

Verrà descritta la metodologia utilizzata per valutare le prestazioni del tool, compresa la configurazione dei test e i criteri di valutazione.

Valutazione delle Alternative

Inizialmente, l'idea era quella di sfruttare qualche tool o creare uno script personalizzato per generare connessioni con indirizzi IP differenti e inviare il traffico al server con il tool in esecuzione. Tuttavia, questa strategia si è rivelata problematica per diverse ragioni.

Da un lato, al momento non esistono tool disponibili che consentano la generazione automatica di connessioni da indirizzi IP diversi in modo da simulare il traffico malevolo e non. Dall'altro lato, la creazione di script personalizzati con questa funzionalità si è scontrata con restrizioni imposte dagli Internet Service Providers (ISP). Gli ISP spesso limitano la possibilità di inviare pacchetti da indirizzi IP diversi in quanto ciò potrebbe essere associato a attività sospette o dannose, come attacchi DDoS o tentativi di frode.

Di conseguenza, è stata necessaria una revisione dell'approccio ai test di prestazione del tool e la ricerca di alternative realistiche per valutarne l'efficacia.

Analisi dei File PCAPs

Per valutare l'efficacia del tool nella rilevazione e nel blocco delle connessioni indesiderate, è stata condotta un'analisi approfondita di file PCAP (Packet Capture) rappresentativi del traffico di rete crittografato. Questa fase di test è stata fondamentale per comprendere come il tool si comportasse in scenari realistici e per misurarne l'accuratezza nel rilevare minacce potenziali.

Sono stati selezionati e utilizzati dieci file PCAPs da fonti affidabili, tra cui il sito web www.malwaretrafficanalysis.net. Questi file PCAPs contengono dati di traffico di rete reale, inclusi pacchetti cifrati, provenienti da varie fonti e destinazioni. I file PCAPs selezionati sono stati elaborati utilizzando il tool al fine di testarne l'efficacia nel rilevare connessioni indesiderate. Purtroppo non tutti i file analizzati contenevano connessioni TLS dannose, perciò è stato considerato solo un gruppo di quattro file PCAP che conteneva pacchetti cifrati corrispondenti a segnalazioni nella blacklist utilizzata dal tool. Questi file PCAP rappresentavano scenari reali in cui il tool doveva essere in grado di identificare e bloccare le connessioni malevole.

L'analisi dei file è stata condotta in modalità offline, consentendo al tool di esaminare il traffico storico e identificare eventuali connessioni indesiderate o potenzialmente dannose. Questo approccio ha permesso una valutazione dettagliata delle prestazioni del tool in condizioni realistiche, tenendo conto di minacce esistenti e strategie di crittografia utilizzate nella rete.

4.3.2 Risultati Conseguiti

Di seguito verranno esaminate le misurazioni e i risultati relativi al calcolo del ja3 dai pacchetti Client Hello e le azioni intraprese in caso di match con la *Ja3Blacklist*. Questa analisi complessiva mira a valutare l'efficacia del tool nel contrastare il traffico indesiderato.

MALWARE	MATCH
Bart	NO
BitPaymer 2	NO
Cerber	NO
Sodinokibi	NO
CryLock	NO
Eris	SI
CTB locker	SI
WannaCry	NO
GrandCrab	SI
Razi	NO
Ryuk	NO
Dharma	NO
CryptoFortress	NO
Shade	SI

La tabella riassume i malware analizzati, mostrando quali hanno avuto riscontro positivo nella blacklist.


```

File Edit View Search Terminal Help

[+] Loaded ja3blacklist from: JA3blacklist.txt with 148 entries
[+] Loaded IPblacklist from: IPblacklist.txt with 1 entries
[+] mode: offline
[+] filename: MATCHED/CTBLocker_27012017.pcap
[+] BPF: (tcp[tcp[12]/16*4]=22) and (tcp[tcp[12]/16*4+9]=3) and (tcp[tcp[12]/16*4+1]=3)
[+] type filter: all
[+] output filename: stdout
[+] output as json: False
[+] save raw pcap: False

reading from file MATCHED/CTBLocker_27012017.pcap, link-type EN10MB (Ethernet), snapshot length 65535
[+] Hello from Client
[-] type: TLS
[-] src ip: 10.0.2.4
[-] src port: 49381
[-] dst ip: 216.58.214.174 (tools.google.com)
[-] dst port: 443
[-] ja3: 769,47-53-5-10-49171-49172-49161-49162-50-56-19-4,65281-0-10-11,23-24,0
[-] ja3s_no_grease: 769,47-53-5-10-49171-49172-49161-49162-50-56-19-4,65281-0-10-11,23-24,0
[-] md5: 1d095e68489d3c535297cd8dffb06cb9
[-] md5_no_grease: 1d095e68489d3c535297cd8dffb06cb9
[-] Match: Yes
[-] Second Match: No

[+] Hello from Server
[-] type: TLS
[-] src ip: 216.58.214.174
[-] src port: 443
[-] dst ip: 10.0.2.4
[-] dst port: 49381
[-] ja3s: 769,49161,65281-11
[-] ja3s_no_grease: 769,49161,65281-11
[-] md5: 83916e6c5d8bffb48942b81f585372ea
[-] md5_no_grease: 83916e6c5d8bffb48942b81f585372ea
[-] Match: No
[-] Second Match: No

[+] all packets: 2; client hello: 1; server hello: 1; in 0.3s

```

Figura 4.3.1: Calcolo del ja3 per il malware *CTB Locker*

```

File Edit View Search Terminal Help

[*] Loaded ja3blacklist from: JA3blacklist.txt with 148 entries
[*] Loaded IPblacklist from: IPblacklist.txt with 2 entries
[*] mode: offline
[*] Filename: MATCHED/Eris_02082019.pcap

[*] BPF: (tcp[tcp[12]/16*4]=22) and (tcp[tcp[12]/16*4+9]=3) and (tcp[tcp[12]/16*4+1]=3)
[*] type filter: all
[*] output filename: stdout
[*] output as json: False
[*] save raw pcap: False

reading from file MATCHED/Eris_02082019.pcap, link-type EN10MB (Ethernet), snapshot length 65535
[*] Hello from Client
[-] type: TLS
[-] src ip: 192.168.1.4
[-] src port: 49263
[-] dst ip: 172.217.16.227 (update.googleapis.com)
[-] dst port: 443
[-] ja3: 769,47-53-5-10-49171-49172-49161-49162-50-56-19-4,65281-0-10-11,23-24,0
[-] ja3_no_grease: 769,47-53-5-10-49171-49172-49161-49162-50-56-19-4,65281-0-10-11,23-24,0
[-] md5: 1d095e68489d3c535297cd8dffb06cb9
[-] md5_no_grease: 1d095e68489d3c535297cd8dffb06cb9
[-] Match: Yes
[-] Second Match: No

[*] Hello from Server
[-] type: TLS
[-] src ip: 172.217.16.227
[-] src port: 443
[-] dst ip: 192.168.1.4
[-] dst port: 49263
[-] ja3s: 769,49161,65281-11
[-] ja3s_no_grease: 769,49161,65281-11
[-] md5: 83916e6c5d8bffb48942b81f505372ea
[-] md5_no_grease: 83916e6c5d8bffb48942b81f505372ea
[-] Match: No
[-] Second Match: No

[*] all packets: 2; client hello: 1; server hello: 1; in 1.4s

```

Figura 4.3.2: Calcolo del ja3 per il malware *Eris*


```

File Edit View Search Terminal Help

[*] Loaded ja3blacklist from: JA3blacklist.txt with 149 entries
[*] Loaded IPblacklist from: IPblacklist.txt with 0 entries
[*] mode: offline
[*] filename: MATCHED/Shade_16092021.pcap

[*] BPF: (tcp[tcp[12]/16*4]=22) and (tcp[tcp[12]/16*4+9]=3) and (tcp[tcp[12]/16*4+1]=3)
[*] type filter: all
[*] output filename: stdout
[*] output as json: False
[*] save raw pcap: False


reading from file MATCHED/Shade_16092021.pcap, link-type EN10MB (Ethernet), snapshot length 65535
[*] Hello from Client
[-] Type: TLS
[-] src ip: 192.168.1.4
[-] src port: 49195
[-] dst ip: 171.25.193.9 (www.uop7qkbpibvt3lu.com)
[-] dst port: 80
[-] ja3: 771,49195-49199-49162-49161-49171-49172-51-57-47-53-255,0-11-10-35-13-15,14-13-25-11-12-24-9-10-22-23-8-6-7-20-21-4-5-18-19-1-2-3-15-16-17,0-1-2
[-] ja3s_no_grease: 771,49195-49199-49162-49161-49171-49172-51-57-47-53-255,0-11-10-35-13-15,14-13-25-11-12-24-9-10-22-23-8-6-7-20-21-4-5-18-19-1-2-3-15-16-17,0-1-2
[-] md5: 1be3e3e3e5aa9d3654e6e703d81f6928
[-] md5_no_grease: 1be3e3e3e5aa9d3654e6e703d81f6928
[-] Match: Yes
[-] Second Match: No

[*] Hello from Server
[-] Type: TLS
[-] src ip: 171.25.193.9
[-] src port: 80
[-] dst ip: 192.168.1.4
[-] dst port: 49195
[-] ja3s: 771,49199,65281-11
[-] ja3s_no_grease: 771,49199,65281-11
[-] md5: 303951d4c50efb2e091652225a6f02b1
[-] md5_no_grease: 303951d4c50efb2e091652225a6f02b1
[-] Match: No
[-] Second Match: No

```

Figura 4.3.4: Calcolo del ja3 per il malware *Shade*

Si può notare come per questi Pcap di malware il tool abbia calcolato il ja3 e successivamente abbia trovato una corrispondenza nella blacklist utilizzata.

A terminal window with a dark background and light text. The window has a menu bar at the top with 'File', 'Edit', 'View', 'Search', 'Terminal', and 'Help'. The terminal shows the command 'cat IPblacklist.txt' being executed, which outputs two IP addresses: '19.0.2.4' and '192.168.1.4'. The prompt 'root@mic-RHEL:~/TESIA#' is visible at the bottom.

```
File Edit View Search Terminal Help
root@mic-RHEL:~/TESIA# cat IPblacklist.txt
19.0.2.4
192.168.1.4
root@mic-RHEL:~/TESIA#
```

Figura 4.3.5: Aggiunta degli IP malevoli alla seconda Blacklist

l'efficacia del tool nel rilevare e bloccare connessioni indesiderate. Questi risultati consolidano la posizione del tool come una risorsa per affrontare minacce alla sicurezza e gestire carichi di lavoro critici in ambienti di rete complessi.

Capitolo 5

Conclusioni

5.1 Riassunto delle Conclusioni

Il presente capitolo offre un riassunto dei principali risultati emersi dalla ricerca condotta, enfatizzando i successi e i contributi principali di questo lavoro.

In primo luogo, è stato sviluppato un tool di sicurezza basato sull'analisi delle impronte digitali crittografiche JA3. Questo strumento si è dimostrato efficace nel rilevare e bloccare connessioni malevole all'interno delle reti informatiche. La sua capacità di analizzare con precisione le impronte digitali crittografiche ha permesso di identificare potenziali minacce alla sicurezza in modo affidabile. Durante la fase sperimentale, il tool ha superato con successo una serie di test, dimostrando la sua capacità di gestire carichi di lavoro estremamente elevati e blacklist di dimensioni considerevoli. Questi risultati hanno evidenziato la scalabilità e l'affidabilità del tool in una varietà di scenari operativi.

Inoltre, l'analisi dei file PCAP ha ulteriormente confermato l'efficacia del tool nella rilevazione di connessioni malevole in situazioni reali. Questo approccio offline ha permesso una valutazione dettagliata delle prestazioni del tool, considerando le minacce esistenti e le complesse strategie di crittografia utilizzate nella rete.

Questi risultati rappresentano un notevole passo avanti nella difesa delle reti informatiche contro minacce sempre più sofisticate e persistenti. Il tool basato su JA3 offre un approccio innovativo per la rilevazione e la prevenzione delle intrusioni, contribuendo significativamente alla sicurezza delle reti aziendali e delle infrastrutture critiche.

Questo lavoro di ricerca dimostra che l'analisi delle impronte digitali crittografiche JA3 è una tecnologia promettente con un alto potenziale per ulteriori sviluppi nel

campo della sicurezza informatica. Gli amministratori di rete e i professionisti della sicurezza possono sfruttare questo strumento come parte integrante delle loro strategie di difesa, migliorando notevolmente la capacità di rilevare e rispondere alle minacce.

La ricerca condotta in questa tesi fornisce una solida base per ulteriori esplorazioni e ricerche nel campo della sicurezza informatica basata su JA3.

5.2 Lavori Futuri

Nonostante i successi raggiunti in questa tesi, ci sono diverse direzioni in cui questa ricerca potrebbe essere estesa per approfondire ulteriormente l'argomento della sicurezza informatica basata su JA3.

Una possibile area di sviluppo futuro riguarda l'implementazione di meccanismi di machine learning all'interno del tool. L'addestramento di modelli di machine learning potrebbe consentire una rilevazione più sofisticata delle minacce e una migliore adattabilità alle nuove strategie di attacco.

Inoltre, sarebbe interessante esplorare come il tool possa essere integrato in ambienti di sicurezza informatica più ampi, contribuendo a una difesa olistica delle reti aziendali. L'implementazione di una dashboard di monitoraggio e gestione potrebbe semplificare l'uso del tool da parte degli amministratori di rete.

Infine, l'espansione della blacklist utilizzata dal tool potrebbe essere oggetto di ulteriori ricerche. L'identificazione e l'aggiunta rapida di nuove impronte JA3 potrebbero migliorare la capacità del tool di adattarsi alle minacce emergenti.

Bibliografia

- [1] Mohammed Abdul Qadeer et al. *Network Traffic Analysis and Intrusion Detection Using Packet Sniffer*. 2010. DOI: 10.1109/ICCSN.2010.104.
- [2] Fedor Sinitzyn. *A new Generation of Ransomware*. Rapp. tecn. SecureList, 2014.
- [3] Amos Azaria et al. “Behavioral Analysis of Insider Threat: A Survey and Bootstrapped Prediction in Imbalanced Data”. In: *IEEE Transactions on Computational Social Systems* 1.2 (2014), pp. 135–155. DOI: 10.1109/TCSS.2014.2377811.
- [4] Brian Contos. *Making metadata meaningful for network security*. Rapp. tecn. CSO, 2013.
- [5] Blake Anderson e David McGrew. “Machine Learning for Encrypted Malware Traffic Classification: Accounting for Noisy Labels and Non-Stationarity”. In: *Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. KDD '17. Halifax, NS, Canada: Association for Computing Machinery, 2017, pp. 1723–1732. ISBN: 9781450348874. DOI: 10.1145/3097983.3098163. URL: <https://doi.org/10.1145/3097983.3098163>.
- [6] Mario Guimaraes e Meg Murray. “Overview of Intrusion Detection and Intrusion Prevention”. In: *Proceedings of the 5th Annual Conference on Information Security Curriculum Development*. InfoSecCD '08. Kennesaw, Georgia: Association for Computing Machinery, 2008, pp. 44–46. ISBN: 9781605583334. DOI: 10.1145/1456625.1456638. URL: <https://doi.org/10.1145/1456625.1456638>.
- [7] Simone Magnani, Fulvio Risso e Domenico Siracusa. “A Control Plane Enabling Automated and Fully Adaptive Network Traffic Monitoring With eBPF”. In: *IEEE Access* 10 (2022), pp. 90778–90791. DOI: 10.1109/ACCESS.2022.3202644.
- [8] Rohith Raj S et al. “SCAPY- A powerful interactive packet manipulation program”. In: *2018 International Conference on Networking, Embedded and Wireless Systems (ICNEWS)*. 2018, pp. 1–5. DOI: 10.1109/ICNEWS.2018.8903954.