Università degli Studi La Sapienza

Homework Sicurezza

Intrusion Detection System: quali tecniche adottare?

Dipartimento di Informatica

Mattia Baldinetti

A.A. 2022/2023

Traccia 3

Le tecniche di Machine Learning (ML) e Intelligenza Artificiale (AI) sono oggi adottate nell'ambito dell'Intrusion Detection, ad esempio con lo scopo di aumentare la capacità degli IDS di identificare intrusioni non note. Si chiede di analizzare la letteratura scientifica e tecnica relativa a soluzioni di intrusion detection al fine di dare risposta alla seguenti domande:

- Quali sono le principali tecniche di ML, deep learning, ed AI in generale, che sono state proposte per risolvere il problema dell'intrusion detection? (dare anche una descrizione del loro funzionamento)
- Quali tipologie di intrusioni è possibile identificare con tali tecniche, e quali tipi di intrusioni invece non possono essere identificate?
- Le soluzioni proposte sono di carattere generale, ovvero permettono di identificare più tipi di attacchi, oppure per ogni tipo di attacco va creata una soluzione dedicata?
- Quali sono i dataset a disposizione per sperimentare (training e testing) le tecniche di intrusion detection precedentemente identificate, e quali sono le principali caratteristiche di questi dataset?

La studentessa / lo studente deve fare riferimento alla letteratura tecnica e scientifica degli ultimi 10 anni (2012 – 2023); le pubblicazioni considerate devono essere state pubblicate su riviste o conferenze IEEE, ACM, Elsevier, Springer; l'analisi si deve basare su almeno 10 pubblicazioni.

Contents

1 Introduzione

1.1 Cos'è un Sistema di rilevamento delle intrusioni (IDS)

Internet è diventato parte integrante della nostra vita mentre svolgiamo attività quotidiane come l'e-Banking, l'e-Education e l'e-Commerce. Con ciò, è aumentata anche la minaccia di aggressori e hacker. Anche se i firewall e i router proteggono la rete, mancano di rilevare dati dannosi e intrusi. Un ruolo cruciale è stato svolto dal sistema di rilevamento delle intrusioni (IDS) che rileva tali atti dannosi [1].

Intrusione è un termine che descrive l'atto dannoso di compromettere il sistema e colpisce l'integrità, la riservatezza e la disponibilità delle risorse. IDS monitora le attività di rete, di sistema e le violazioni delle policy e produce report per l'amministratore. Genera avvisi quando si verifica un'intrusione. Pertanto, IDS consente principalmente ai sistemi informatici di affrontare le minacce di rete [2].

2 Tecniche analizzate

Negli anni passati sono state utilizzate varie tecniche basate su AI (Artificial Intelligence) e ML (Machine Learning) per aumentare la precisione di un IDS grezzo, ma la loro accuratezza rimane lo stesso un problema: la precisione dipende dal rilevamento e dal tasso di falsi allarmi. Il problema dell'accuratezza deve essere affrontato per ridurre il tasso di falsi allarmi e aumentare il tasso di rilevamento. La classificazione e il clustering sono due approcci principalmente utilizzati in questo campo. Il primo si basa su una tecnica di apprendimento automatico supervisionato che utilizza classi predefinite in cui vengono assegnati gli oggetti, mentre il secondo è una tecnica di apprendimento automatico non supervisionato che identifica le somiglianze tra gli oggetti [1].

Esistono numerosi algoritmi basati sulla classificazione come Random Forest, Support Vector Machine (SVM) e Decision Tree (DT). Dal punto di vista del rilevamento delle intrusioni, gli algoritmi di classificazione possono caratterizzare i dati di rete come dannosi, innocui, di scansione o qualsiasi altra categoria di interesse utilizzando informazioni come porte di origine/destinazione, indirizzi IP e il numero di byte inviati durante una connessione. Di questi, SVM è altamente efficace e diventa il miglior algoritmo di apprendimento per la classificazione.

Nel clustering, K-means è estremamente utilizzato in molte applicazioni [1].

2.1 Support Vector Machine (SVM)

Coretes e Vapink hanno proposto per la prima volta il Support Vector Machine (SVM) nel 1995. Mostra molti vantaggi unici in un campione piccolo e non lineare [3].

Questo ordina i dati in due serie, creando un modello così come viene inizialmente addestrato. Viene applicato con successo a varie applicazioni, tra cui la classificazione di testi e ipertesti, l'elaborazione di immagini e le scienze biologiche. Uno dei principali vantaggi dell'utilizzo di questo algoritmo è la sua alta velocità e la capacità di trovare intrusioni in tempo reale. La scalabilità di SVM è altamente adattiva e può espandersi meglio poiché la classificazione non ha alcun problema con la dimensionalità dello spazio delle caratteristiche. Inoltre, ha la capacità di aggiungere e modificare automaticamente le sequenze [1].

Le prestazioni di SVM diminuiscono quando sono coinvolti dati di grandi dimensioni e non è la scelta ideale per analizzare grandi quantità di traffico di rete [2].

Prima dell'avvento del deep learning, SVM era considerato il metodo di apprendimento automatico di maggior successo e più performante negli ultimi decenni [3].

2.2 Random Forest

Leo Breiman ha proposto Random Forest nel 2001. Random Forest è un eccellente algoritmo di apprendimento automatico [3]. Come si può intuire dal suo nome, crea una "foresta", composta da alberi decisionali (Decision Tree), attraverso la suddivisione casuale dei nodi e il ricampionamento casuale di essi [4]. La Figura 1 mostra l'implementazione del modello di classificazione Random Forest. Un campione pre-elaborato di n campioni viene inviato al classificatore di Random Forest. RF crea n alberi diversi utilizzando un numero di sottoinsiemi di funzionalità. Ogni albero produce un risultato di classificazione e il risultato del modello di classificazione dipende dal voto di maggioranza. Il campione viene assegnato alla classe che ottiene il maggior punteggio di voto in modo casuale. La "foresta" che costruisce è un insieme di Decision Trees. RF raggiunge un'elevata precisione di classificazione e può gestire valori anomali e rumore nei dati. Altri vantaggi del RF includono la sua maggiore precisione rispetto ad altri algoritmi e minori possibilità di overfitting, ossia quando un modello si adatta troppo bene ai dati di training e, di conseguenza, non può prevedere in modo accurato i dati di test non visualizzati. Il limite principale dell'algoritmo RF è che molti alberi possono rallentare l'algoritmo per la previsione in tempo reale [2].

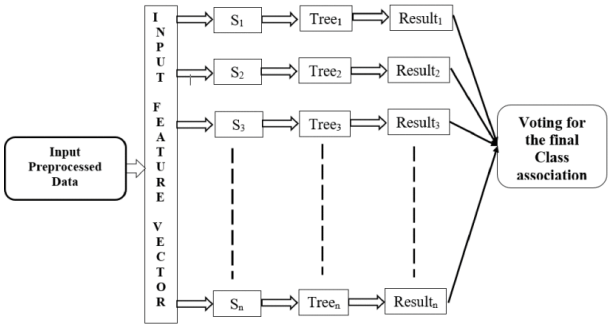

Figure 1

2.3 Decision Tree

L'albero decisionale è inizialmente costruito da un insieme di dati preclassificati. L'approccio principale consiste nel selezionare gli attributi, che suddividono al meglio gli elementi di dati nelle loro classi. In base ai valori di questi attributi gli elementi di dati sono partizionati. Questo processo viene applicato in modo ricorsivo a ciascun sottoinsieme partizionato degli elementi di dati. Il processo termina quando tutti gli elementi di dati nel sottoinsieme corrente appartengono alla stessa classe. Un nodo di un albero decisionale specifica un attributo in base al quale i dati devono essere partizionati. Ogni nodo ha un numero di spigoli, che sono etichettati in base a un possibile valore dell'attributo nel nodo genitore. Un ramo collega due nodi o un nodo e una foglia. Le foglie sono etichettate con un valore decisionale per la categorizzazione dei dati. Gli algoritmi di classificazione creano un albero decisionale come quello presentato nella Figura 1, identificando modelli in un set di dati esistente e utilizzando tali informazioni per creare l'albero. Gli algoritmi accettano dati preclassificati come input. È importante differenziare gli alberi decisionali da altri strumenti in tempo reale che proteggono le reti di interesse. Gli alberi decisionali sono strumenti per l'analisi dei dati e l'identificazione di caratteristiche significative nei dati di rete che indicano attività dannose. Gli alberi decisionali possono aiutare i team a determinare quali firme IDS scrivere, quali regole firewall implementare e quale tipo di attività di rete contrassegnare per ulteriori analisi. Tuttavia, gli alberi decisionali da soli non agiscono per fermare le minacce, come i firewall e i sistemi di prevenzione delle intrusioni (IPS). La loro logica decisionale può essere utilizzata insieme ad altri strumenti in tempo reale. Gli alberi decisionali possono essere utilizzati come rilevamento delle intrusioni per uso improprio in quanto possono apprendere un modello basato sui dati di addestramento e possono prevedere i dati futuri come uno dei tipi di attacco o normali in base al modello appreso. Gli alberi decisionali funzionano bene con set

di dati di grandi dimensioni. L'accuratezza della generalizzazione degli alberi decisionali è un'altra proprietà utile per il modello di rilevamento delle intrusioni. Ci saranno sempre alcuni nuovi attacchi al sistema che sono piccole variazioni di attacchi noti dopo la creazione dei modelli di rilevamento delle intrusioni. La capacità di rilevare queste nuove intrusioni è possibile grazie all'accuratezza della generalizzazione degli alberi decisionali [5].

2.4 K-means

L'algoritmo K-means è un algoritmo iterativo che cerca di partizionare l'insieme di dati in 'k' sottogruppi distinti non sovrapposti (cluster) predefiniti in cui ogni punto dato appartiene a un solo gruppo. Assegna i punti a un cluster in modo tale che la somma della distanza quadrata tra i punti e il centroide del cluster (media aritmetica di tutti i punti che appartengono a quel cluster) sia al minimo. Meno variazioni abbiamo all'interno dei cluster, più omogenei (simili) i punti sono all'interno dello stesso cluster.

Il modo in cui funziona l'algoritmo K-means è il seguente:

- Specificare il numero di cluster K.
- Inizializza i centroidi mescolando prima il set di dati e quindi selezionando in modo casuale K punti dati per i centroidi.
- Continua a ripetere fino a quando non ci sono modifiche ai centroidi, ovvero l'assegnazione dei punti ai cluster non cambia
  - Calcola la somma della distanza al quadrato tra i punti dati e tutti i centroidi.
  - Assegna ciascun punto dati al cluster più vicino (centroide).
  - Calcola i centroidi per i cluster prendendo la media di tutti i punti dati che appartengono a ciascun cluster.

In parole semplici l'algoritmo K-mean calcola la distanza tra il centro del cluster e il punto campione utilizzando la distanza euclidea [4].

Sebbene K-Means sia l'algoritmo di base del clustering e ampiamente utilizzato nel sistema di rilevamento delle intrusioni, questo algoritmo presenta varie limitazioni:

- Ha bisogno di informazioni a priori sui diversi cluster.
- K-Means non è in grado di identificare i due cluster che sono completamente sovrapposti.

[1]

3 Accuratezza nell'indentificare le intrusioni

Prima di procedere con le percentuali di accuratezza di ogni tecnica individuata, è bene evidenziare quelle che sono le formule dietro le variabili su cui basiamo la nostra ricerca:

- Accuratezza = TP + TN / P + N , dove P = TP + FP e N = TN + FN
- P: positive
- N: negative
- TP: true positive (dati di attacco correttamente classificati come attacco)
- TN: true negative (dati normali correttamente classificati come normali)

- FP: fake positive (dati normali erroneamente classificati come attacco)
- FN: fake neagtive (dati di attacco erroneamente classificati come normali)

3.1 Support Vector Machine

Support Vector Machine presenta un accuratezza del rilevamento, con l'utilizzo di un Dataset KDD-99, del 89,02%, mentre utilizzando la nuova versione di questo Dataset, ossia NSL-KDD, la percentuale sale al 92,13% [1].

3.2 Random Forest

Random Forest è stato testato con il Dataset KDD-99 e presenta un accuratezza del 92,48% [1].

3.3 K-means

K-Means utilizza NSL-KDD come Dataset, e la percentuale di accuratezza è del 81,61% [2].

4 Nuovi sistemi proposti

È possibile utilizzare più tecniche contemporaneamente per cercare di aumentare la percentuale di rilevamento delle intrusioni.

4.1 Approccio ibrido di K-Means clustering e adaptive Support Vector Machine

Iniziamo presentando un nuovo sistema proposto che utilizza un approccio ibrido di K-Means clustering e adaptive SVM, utilizzando come Dataset NSL-KDD.

Approccio al rilevamento

La Figura 2 descrive i vari moduli dell'approccio ibrido. L'approccio ibrido di IDS è il più adatto per ottenere il rilevamento desiderato e il tasso di falsi allarmi.

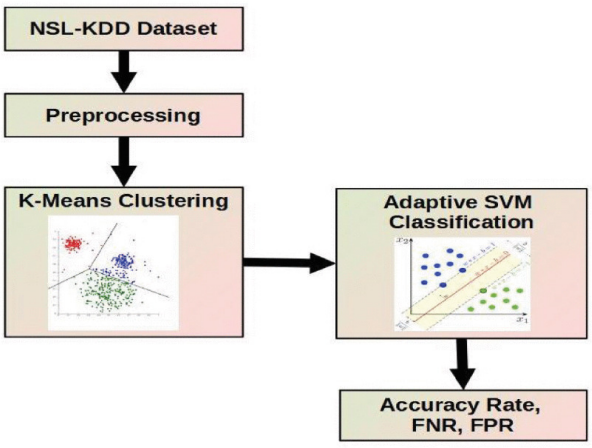

Figure 2

Di seguito sono riportati i passaggi per l'approccio ibrido:

1. Un dato con o senza attacco viene caricato nel set di dati NSL-KDD.

2. Come per il set di dati di addestramento, l'algoritmo K-Means ha preclassificato i dati.
3. I dati di addestramento sono divisi in due gruppi: dati normali e dati di anomalia.
4. Questi gruppi sono ulteriormente suddivisi in dati Train e Test.
5. I dati di addestramento sono raggruppati in due categorie: Train_train e Train_test set.
6. La classificazione SVM viene applicata utilizzando alcuni parametri arbitrari sul set Train_train e Train_test.
7. Trova i parametri effettivi.
8. Rieseguire la classificazione SVM per il set di dati completo.
9. Il tasso di efficienza, il tasso di falsi positivi e il tasso di falsi negativi sono considerati i risultati.

Risultati sperimentali

Per valutare l'approccio ibrido, abbiamo confrontato K-Means e Adaptive SVM come approcci individuali. I parametri di output sono accuratezza nel rilevamento (DA), tasso di falsi positivi (FPR), tasso di falsi negativi (FNR).

Come mostrato in Figura 3, dopo aver eseguito il test 10 volte, si vede chiaramente come l'approccio ibrido sia il migliore.

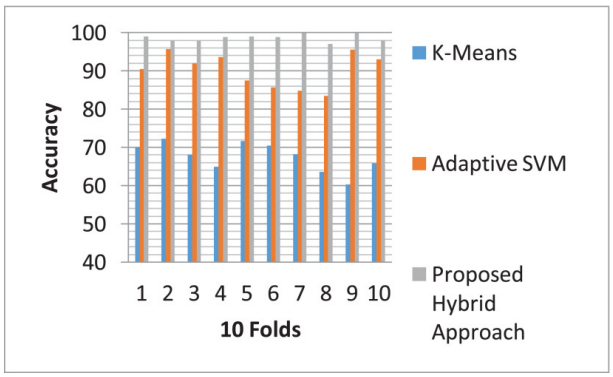

Figure 3

L'accuratezza di rilevamento media delle medie K-Means è dell'81,61%, l'SVM adattivo è del 92,13% e l'approccio ibrido proposto è del 99,54%, come mostrato nella Figura 4. Ciò significa che la più efficiente è la combinazione di entrambi.

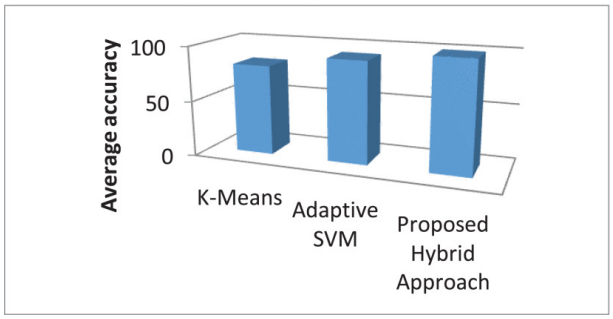

Figure 4

Per quanto riguardi i parametri FPR e FNR, sono riportati rispettivamente nelle due figure successive, 5 e 6.

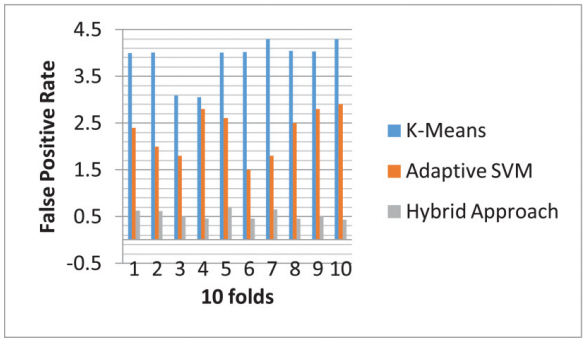

Figure 5

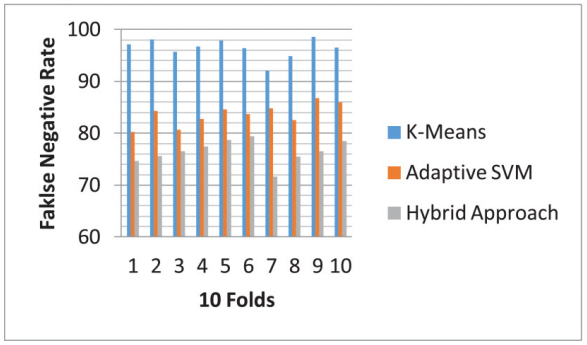

Figure 6

Vediamo sempre come l'approccio ibrido sia il migliore, ancora più evidente nella Figura 7, dove è riportata la media dei risultati ottenuti.

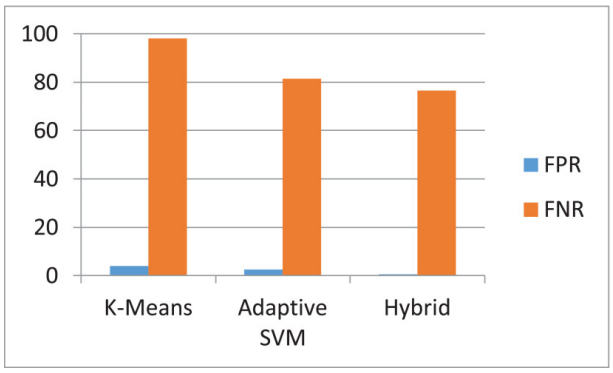

Figure 7

Conclusioni

L'IDS svolge un ruolo di primo piano nell'identificazione delle intrusioni. Questo lavoro presenta un approccio ibrido di K-Means e Adaptive SVM e conclude che questa fusione fornisce risultati migliori rispetto ai singoli risultati di K-Means e Adaptive SVM. Inoltre, questo è un algoritmo estremamente accurato rispetto ad altre tecniche.

La tecnologia ibrida identifica con successo i dati come normali e d'attacco, e questa tecnica risulta essere più accurata del 99,54% rispetto alle tecniche eseguite da soli. Pertanto, concludiamo che l'utilizzo di questo sistema in tempo reale offre un tasso di rilevamento degli attacchi molto elevato. Inoltre, questo approccio è semplice ed efficiente soprattutto per la riduzione del rapporto dei falsi positivi e per l'aumento del rapporto dei falsi negativi [1].

4.2 Random Forest e K-means

Elbasiony et al. ha proposto un modello di rilevamento delle intrusioni basato su RF e K-means e hanno convalidato il loro modello sul set di dati KDD99. Il sistema ha dimostrato risultati con una precisione del 98,3%. La RF non è adatta a prevedere il traffico reale a causa della sua lentezza, dovuta alla formazione di un gran numero di alberi. Inoltre, il set di dati KDD99 presenta alcune limitazioni [2].

4.3 Random Forest

Farnaaz e Jabbar hanno sviluppato un modello per un sistema di rilevamento delle intrusioni basato su RF. Hanno testato l'efficacia del loro modello su un set di dati NSL-KDD e i loro risultati hanno dimostrato un tasso di rilevamento del 99,67%. Il limite principale dell'algoritmo RF è che molti alberi possono rallentare l'algoritmo per la previsione in tempo reale [2].

4.4 Support Vector Machine basato su Decision Tree

Teng et al. ha svolto un lavoro importante. Hanno sviluppato il loro modello basato su alberi decisionali (DT) e SVM e hanno testato il loro modello su un set di dati KDD Cup 1999. I risultati hanno mostrato una precisione che raggiunge l'89,02%. Tuttavia, le SVM non sono preferite per set di dati pesanti a causa dell'elevato costo di calcolo e delle scarse prestazioni [2]. Utilizzando lo stesso approccio, ma usando la versione aggiornata del Dataset, ossia NSL-KDD, si è riusciti a raggiungere una percentuale di accuratezza del 99% [1].

4.5 Framwork basato su Support Vector Machine

Wang et al. ha proposto un framework di rilevamento delle intrusioni basato su SVM e ha convalidato il proprio metodo sul set di dati NSL-KDD. Hanno affermato che il loro metodo, che ha un tasso di efficacia del 99,92%, era superiore ad altri approcci. Nonostante questo, le prestazioni di SVM diminuiscono quando sono coinvolti dati di grandi dimensioni e non è la scelta ideale per analizzare l'enorme traffico di rete per il rilevamento delle intrusioni [2].

5 Intrusioni rilevate

Questo ultimo metodo sembra essere quello con la percentuale di accuratezza più elevata nel rilevamento delle intrusione, tra tutti quelli studiati in questa raccolta. In generale però, tutte le tecniche elencate presentano ottimi risultati per quanto riguarda il rilevamento delle intrusioni. Infatti, tutte queste tecniche riescono a contrastare quelle che sono le 5 classi di attacchi (una normale e 4 classi di intrusione):

- Le connessioni normali sono generate da un comportamento quotidiano simulato dell'utente come il download di file, la visita di pagine web.
- L'attacco Denial of Service (DoS) fa sì che la potenza di calcolo o la memoria di una macchina vittima sia troppo occupata o troppo piena per gestire le richieste legittime.
- Remote to User (R2L) è un attacco in cui un utente remoto ottiene l'accesso a un utente/account locale inviando pacchetti a una macchina tramite una comunicazione di rete, che include sendmail e Xlock.
- User to Root (U2R) è un attacco in cui un intruso inizia con l'accesso a un normale account utente e poi diventa un utente root sfruttando varie vulnerabilità del sistema. Gli exploit più comuni degli attacchi U2R sono buffer overflow regolari, load-module, Fd-format e Ffb-config.
- Probing (Probe) è un attacco che scansiona una rete per raccogliere informazioni o trovare vulnerabilità note. Un intruso con una mappa di macchine e servizi disponibili su una rete può utilizzare le informazioni per cercare exploit.

[5]

6 Intrusioni non rilevate

Nonostante le tecniche fin qui analizzate riescano a intercettare una percentuale altissima di intrusioni, fanno difficoltà a identificare gli attacchi di tipo Zero-Day. Per questo motivo sono state sviluppate nuove tecniche, sempre utilizzando quelle elencate precedentemente, che riescono in parte a minimizzare il danno arrecato dagli Zero-Day.

6.1 Zero-Day

Un attacco zero-day è un attacco a una vulnerabilità sconosciuta alla comunità pubblica. Se un hacker riesce a sfruttare la vulnerabilità informatica prima che gli sviluppatori di software possano trovare una soluzione, tale exploit diventa noto come attacco zero day. Le vulnerabilità zero day possono assumere quasi tutte le forme, perché possono manifestarsi come qualsiasi tipo di vulnerabilità software più ampia. La principale sfida in tali approcci classici è che le nuove varianti di malware utilizzano tecniche di evasione antivirus come l'offuscamento del codice. Questo rende le vulnerabilità zero day difficili da individuare, in quanto neanche tramite aggiornamenti di sicurezza del software è possibile prevenirle [10].

7 Nuovi sistemi proposti per identificare attacchi Zero-Day

7.1 Support Vector Machine

- Insieme di dati: In [11.1], vengono valutate le prestazioni di One-Class SVM. Gli esperimenti utilizzano due set di dati: CIC-IDS2017 e NSL-KDD. Entrambi i set di dati sono ampiamente utilizzati nelle valutazioni di rilevamento delle intrusioni di rete.
- Formazione del modello: Ciascuna delle classi di attacco imita un attacco zero-day e viene utilizzato per valutare la capacità del modello per rilevare la sua anomalia.
- Risultati dell'esperimento: La precisione di rilevamento dell'attacco zero-day varia notevolmente secondo i diversi tipi di attacco. Per esempio, la precisione di rilevamento degli attacchi che sono molto diversi dai benigni (per esempio, Hulk e DDoS), è alto, nel range dal 92% al 99%; mentre il livello di accuratezza di rilevamento, degli attacchi meno distinguibili da quelli benigni come DoS-SlowHTTPTest, è inferiore del 40%. SVM risulta essere adatto per la segnalazione riconoscibile attacchi zero-day, e mostra un basso tasso di fake positive.

[11]

7.2 Random Forest

Lo schema viene valutato utilizzando un set di dati privati da un fornitore di servizi Internet. 38 diversi tipi di flussi dannosi sono stati identificati. I flussi che non sono stati etichettati dal sistema IDS sono etichettati come traffico benigno. I dati sono suddivisi in set di allenamento e set di test. Per simulare malware zero-day, alcune delle classi di malware sono state trattenute dal il set di allenamento e appaiono solo nel set di prova. In particolare, dodici classi di malware più diffusi, ad esempio, Sality, Conficker (Downadup), Tidserv, e Trojan, sono selezionati come malware noti e sono inclusi nel set di allenamento (insieme a un numero uguale di flussi benigni). Il resto dei flussi di rete è stato assegnato al set di test che include tutti i 38 flussi di malware. Random Forest è in grado di rilevare tutti i malware noti e 88.54% del malware zero-day, con un accuratezza del 90,96% sulle classi dannose [11].

7.3 Decision Tree

Il set di dati di allenamento è il Dataset CSE-CIC-IDS2018. I ricercatori hanno selezionato Decision Tree come uno dei modelli più efficaci tra una serie di modelli iniziali. Gli autori applicano il modello dell'albero di decisione all'attacco zero-day. Gli attacchi zero-day sono mescolati con i dati benigni. Il tasso true-positive raggiunge il meglio al 96% quando la massima profondità dell'albero è 5, e si deteriora leggermente al 92% e 90% quando la profondità massima dell'albero si riduce a 3 e 4. Il tasso di falsi positivi è più basso al 5% quando la profondità massima è 3 e 4, e si deteriora a circa il 10% quando la profondità massima è inferiore o superiore a 2 o 5, rispettivamente [11].

7.4 Support Vector Machine e Random Forest con l'utilizzo di K-means

Gli esperimenti utilizzano i dati di CA Tecnologie VET Zoo e altre due fonti di dati pubblicamente disponibili.

Oltre ai file puliti, il set di dati contiene tre tipi di malware, Trojan, worm e virus, e venti famiglie di malware. Il dataset di allenamento contiene tutte le famiglie di malware tranne emerleox, una famiglia di virus che è lasciato fuori come attaccante zero-day. Un totale di 907 esempi sono nel set di allenamento mentre 506 esempi sono nel set di test.

Tutti i dati provenienti da entrambi i set di allenamento e test sono fusi in un unico set per clustering non supervisionato. Un algoritmo globale K-means viene utilizzato per raggruppare i dati in tre gruppi. K-means calcola la distanza tra il centro del cluster e il punto campione utilizzando la distanza euclidea. Le distanze geometriche sono considerate come la conoscenza da dati non etichettati. Random Forest raggiunge la massima accuratezza del 98,5348%, un tasso di vero positivo di 0,985, e un tasso di falso positivo di 0,001. Con l'aumentare delle distanze geometriche, sia SVM sia Random Forest raggiungono risultati di rilevamento perfetti: tasso di vero positivo (1), tasso di falsi positivi (0), e accuratezza del (100%) [11].

8 Dataset utilizzati

8.1 KDD Cup 99 (o KDD-99)

Questo set di dati, derivato dal programma di valutazione IDS DARPA'98, contiene oltre 4 gigabyte di dati compressi risultanti da 7 settimane di traffico di rete [6]. Consiste di circa 4.900.000 singoli vettori di connessione ciascuno dei quali contiene 41 caratteristiche che possono essere classificate come segue; funzionalità di base (ad es. tipo di protocollo, dimensione del pacchetto), funzionalità di conoscenza del dominio (ad es. numero di accessi non riusciti) e funzionalità di osservazione temporizzata (ad es. % di connessioni con errori SYN). Ogni vettore è etichettato come normale o come attacco (di cui esistono 22 tipi di attacco specifici) come delineato nella Figura 8:

		10% KDD '99		NSL-KDD	
Category	**Attack Type**	**Train**	**Test**	**Train**	**Test**
DoS	'back'	2203	1098	956	359
	'land'	21	9	18	7
	'neptune'	107201	58001	41214	4657
	'pod'	264	87	201	41
	'smurf'	280790	164091	2646	665
	'teardrop'	979	12	892	12
Probe	'ipsweep'	1247	306	3599	141
	'nmap'	231	84	1493	73
	'portsweep'	1040	354	2931	157
	'satan'	1589	1633	3633	735
R2L	'ftp_write'	8	3	8	3
	'guess_password'	53	4367	53	1231
	'imap'	12	1	11	1
	'multihop'	7	18	7	18
	'phf'	4	2	4	2
	'spy'	2	0	2	0
	'warezclient'	1020	0	890	0
	'warezmaster'	20	1602	20	944
U2R	'loadmodule'	9	2	9	2
	'buffer_overflow'	30	22	30	20
	'rootkit'	10	13	10	13
	'perl'	3	2	3	2
Normal		97278	60593	67343	9711
Total		494021	292300	125973	18794

Figure 8

È pratica comune utilizzare il 10% del set di dati a dimensione intera, in quanto ciò fornisce una rappresentazione adeguata con requisiti di calcolo ridotti. Questo sottoinsieme del 10% viene prodotto e diffuso insieme al set di dati originale [9].

Le classi nel set di dati KDD99 sono state classificate in cinque classi principali (una classe normale e quattro classi principali di intrusione):

1. Le connessioni normali sono generate da un comportamento quotidiano simulato dell'utente come il download di file, la visita di pagine web.
2. Denial of Service (DoS)
3. Remote to User (R2L)
4. User to Root (U2R)
5. Probing (Probe)

[5]

Per i sistemi IDS, non esiste benchmark migliore di questo set di dati [6].

8.2 NSL-KDD

Il dataset NSL-KDD è la nuova versione del dataset KDD Cup 99. Questo set di dati è stato utilizzato per risolvere i problemi di collegamento nel set di dati KDD Cup 1999 [6]. Nel set NSL-KDD ogni classe è etichettata come normale o d'attacco [8].

Il set di dati NSL-KDD ha fondamentalmente la stessa struttura del set di dati KDD Cup 99 (ovvero ha 22 modelli di attacco o traffico normale e campi per 41 caratteristiche), la cui composizione è mostrata nella Figura 8 [9].

Le classi del dataset NSL-KDD sono raggruppate in cinque classi principali: (a) Normale, (b) Denial of Service (DoS), (c) Remote to User (R2L), (d) User to Root (U2R) e (e) Probing (Probe). Il forte vantaggio del set di dati NSL-KDD è che le istanze di allenamento e test sono ragionevoli, quindi, diventa conveniente eseguire gli esperimenti sul set totale di dati di allenamento e test senza la necessità di selezionare casualmente una piccola

parte del set di dati. Il set di dati NSL-KDD presenta i seguenti vantaggi rispetto al set di dati originale KDD 99:

- Non include record ridondanti nel set di allenamento, quindi i classificatori non saranno sbilanciati verso record più frequenti.
- Non ci sono record duplicati nei set di test proposti; pertanto, le prestazioni non sono influenzate dai metodi che hanno tassi di rilevamento migliori sui record frequenti.
- Il numero di record selezionati da ciascun gruppo di livello di difficoltà è inversamente proporzionale alla percentuale di record nel set di dati KDD originale. Di conseguenza, i tassi di classificazione dei diversi metodi di apprendimento automatico variano in un intervallo più ampio, il che rende più efficiente avere una valutazione accurata delle diverse tecniche di apprendimento.
- Il numero di record nei set di allenamento e nei set di test è ragionevole, il che rende conveniente eseguire gli esperimenti sul set completo senza la necessità di selezionare casualmente una piccola porzione. Di conseguenza, i risultati della valutazione dei diversi lavori di ricerca saranno coerenti e comparabili.

[8]

8.3 CIC-IDS2017

Il dataset è stato creato da Il Canadian Institute for Cybersecurity (CIC), con sede presso l'Università del New Brunswick a Fredericton. Di seguito è riportato l'articolo preso dal loro sito ufficiale.

Il set di dati CIC-IDS2017 contiene gli attacchi comuni benigni e più aggiornati, che assomigliano ai veri dati del mondo reale (PCAP). Include anche i risultati dell'analisi del traffico di rete utilizzando CICFlowMeter con flussi etichettati basati su timestamp, IP di origine e destinazione, porte di origine e destinazione, protocolli e attacco (file CSV). La generazione di un traffico di background realistico è stata la massima priorità nella creazione di questo set di dati. E' stato utilizzato il sistema B-Profile proposto (Sharafaldin, et al. 2016) per profilare il comportamento astratto delle interazioni umane e generare traffico di fondo benigno naturalistico. Per questo set di dati, è stato creato il comportamento astratto di 25 utenti in base ai protocolli HTTP, HTTPS, FTP, SSH ed e-mail.

Nel nostro recente quadro di valutazione del set di dati (Gharib et al., 2016), abbiamo identificato undici criteri necessari per costruire un set di dati di riferimento affidabile. Nessuno dei precedenti set di dati IDS poteva coprire tutti gli 11 criteri. Di seguito, illustriamo brevemente questi criteri:

- Configurazione di rete completa: una topologia di rete completa include modem, firewall, switch, router e presenza di una varietà di sistemi operativi come Windows, Ubuntu e Mac OS X.
- Traffico completo: disponendo di un agente di profilazione utente e di 12 macchine diverse in Victim-Network e attacchi reali da Attack-Network.
- Set di dati etichettati.
- Acquisizione completa: poiché abbiamo utilizzato la porta mirror, come il sistema di intercettazione, tutti i traffici sono stati acquisiti e registrati sul server di archiviazione.
- Protocolli disponibili: Prevista la presenza di tutti i comuni protocolli disponibili, come HTTP, HTTPS, FTP, SSH e protocolli email.
- Diversità degli attacchi: inclusi gli attacchi più comuni basati sul rapporto McAfee 2016, come Web based, Brute force, DoS, DDoS, Infiltration, Heart-bleed, Bot e Scan trattati in questo set di dati.

- Eterogeneità: catturato il traffico di rete dallo Switch principale e il dump della memoria e le chiamate di sistema da tutte le macchine vittime, durante l'esecuzione degli attacchi.
- Set di funzionalità: Estratte più di 80 funzionalità di flusso di rete dal traffico di rete generato utilizzando CICFlowMeter e consegnato il set di dati del flusso di rete come file CSV.
- Metadati: spiegato completamente il set di dati che include il tempo, gli attacchi, i flussi e le etichette.

[14]

Nella Figura 9 [12] è riportata la classificazione dei dati nel set:

	Table Column Head					
	Total	Normal	Dos	Probe	R2L	U2L
CIC-IDS2017 Train	125973	68721	44835	10676	996	53
CIC-IDS2017 Test	22533	9812	7548	2134	2690	203

Figure 9

8.4 CSE-CIC-IDS2018

Questo è il set di dati più recente disponibile nel 2018/2019 dal Canadian Institute for Cybersecurity. Nel set di dati CSE-CIC-IDS2018, utilizziamo la nozione di profili per generare set di dati in modo sistematico, che conterranno descrizioni dettagliate di intrusioni e modelli di distribuzione astratti per applicazioni, protocolli o entità di rete di livello inferiore. Questi profili possono essere utilizzati da agenti o persone per creare eventi su una rete e possono essere applicati a vari protocolli di rete con diverse topologie. Inoltre, il set di dati è stato migliorato tenendo conto degli standard utilizzati durante la creazione di CIC-IDS 2017. Oltre ai criteri di base, offre i seguenti vantaggi:

- Il numero di dati duplicati è molto basso.
- I dati incerti sono quasi assenti.
- Il set di dati è in formato CSV, quindi è pronto per l'uso senza un'elaborazione significativa.

Il set di dati contiene sei diversi tipi di intrusione (Brute-force, Botnet, DoS, DDoS, attacchi Web e infiltrazione della rete dall'interno) con un totale di 14 diverse intrusioni; vale a dire: attacco Botnet, FTP-bruteforce, SSH-bruteforce, BruteForceWeb, bruteforce-XSS, SQL Injection, attacco DDoS-HOIC, attacco DDoS-LOICUDP, attacchi DDoS-LOIC-HTTP, infiltrazione, attacco DoS-Hulk, Attacco DoS-SlowHTTPTest, attacco DoS-GoldenEye e DoS-Slowloris attacco. I dati vengono raccolti da reti reali [13].

9 Bibliografia

[1] J. K. Chahal, V. Gandhi, P. Kaushal, K. R. Ramkumar, A. Kaur and S. Mittal, "KAS-IDS: A Machine Learning based Intrusion Detection System," 2021 6th International Conference on Signal Processing, Computing and Control (ISPCC), Solan, India, 2021, pp. 90-95, doi: 10.1109/ISPCC53510.2021.9609402.

[2] I. Ahmad, M. Basheri, M. J. Iqbal and A. Rahim, "Performance Comparison of Support Vector Machine, Random Forest, and Extreme Learning Machine for Intrusion Detection," in IEEE Access, vol. 6, pp. 33789-33795, 2018, doi: 10.1109/ACCESS.2018.2841987.

[3] L. Liu, P. Wang, J. Lin and L. Liu, "Intrusion Detection of Imbalanced Network Traffic Based on Machine Learning and Deep Learning," in IEEE Access, vol. 9, pp. 7550-7563, 2021, doi: 10.1109/ACCESS.2020.3048198.

[4] C. Liu, Z. Gu and J. Wang, "A Hybrid Intrusion Detection System Based on Scalable K-Means+ Random Forest and Deep Learning," in IEEE Access, vol. 9, pp. 75729-75740, 2021, doi: 10.1109/ACCESS.2021.3082147.

[5] M. Kumar, M. Hanumanthappa and T. V. S. Kumar, "Intrusion Detection System using decision tree algorithm," 2012 IEEE 14th International Conference on Communication Technology, Chengdu, China, 2012, pp. 629-634, doi: 10.1109/ICCT.2012.6511281.

[6] I. Manan, F. Rehman, H. Sharif, C. N. Ali, R. R. Ali and A. Liaqat, "Cyber Security Intrusion Detection Using Deep Learning Approaches, Datasets, Bot-IOT Dataset," 2023 4th International Conference on Advancements in Computational Sciences (ICACS), Lahore, Pakistan, 2023, pp. 1-5, doi: 10.1109/ICACS55311.2023.10089688.

[7] S. M. Sangve and R. Thool, "A formal assessment of anomaly network intrusion detection methods and techniques using various datasets," 2015 International Conference on Applied and Theoretical Computing and Communication Technology (iCATccT), Davangere, India, 2015, pp. 267-272, doi: 10.1109/ICATCCT.2015.7456894.

[8] M. S. Pervez and D. M. Farid, "Feature selection and intrusion classification in NSL-KDD cup 99 dataset employing SVMs," The 8th International Conference on Software, Knowledge, Information Management and Applications (SKIMA 2014), Dhaka, Bangladesh, 2014, pp. 1-6, doi: 10.1109/SKIMA.2014.7083539.

[9] N. Shone, T. N. Ngoc, V. D. Phai and Q. Shi, "A Deep Learning Approach to Network Intrusion Detection," in IEEE Transactions on Emerging Topics in Computational Intelligence, vol. 2, no. 1, pp. 41-50, Feb. 2018, doi: 10.1109/TETCI.2017.2772792.

[10] H. Holm, "Signature Based Intrusion Detection for Zero-Day Attacks: (Not) A Closed Chapter?," 2014 47th Hawaii International Conference on System Sciences, Waikoloa, HI, USA, 2014, pp. 4895-4904, doi: 10.1109/HICSS.2014.600.

[11] Y. Guo, "A review of Machine Learning-based zero-day attack detection: Challenges and future directions", Volume 198, 2023, Pages 175-185, ISSN 0140-3664

[11.1] H. Hindy, R. Atkinson, C. Tachtatzis, J.-N. Colin, E. Bayne, X. Bellekens, Utilising deep learning techniques for effective zero-day attack detection, Electronics 9 (10)(2020)
http://dx.doi.org/10.3390/electronics9101684, URL https://www. mdpi.com/2079-9292/9/10/1684.

[12] R. Singh and G. Srivastav, "Novel Framework for Anomaly Detection Using Machine Learning Technique on CIC-IDS2017 Dataset," 2021 International Conference on Technological Advancements and Innovations (ICTAI), Tashkent, Uzbekistan, 2021, pp. 632-636,
doi: 10.1109/ICTAI53825.2021.9673238.

[13] Y. Ayachi, Y. Mellah, J. Berrich and T. Bouchentouf, "Increasing the Performance of an IDS using ANN model on the realistic cyber dataset CSE-CIC- IDS2018," 2020 International Symposium on Advanced

Electrical and Communication Technologies (ISAECT), Marrakech, Morocco, 2020, pp. 1-4, doi: 10.1109/I-SAECT50560.2020.9523662.

[14] Sito Ufficiale della University of New Brunswick: https://www.unb.ca/cic/datasets/ids-2017.html (Ultima consultazione 29/05/2023)