UNIVERSITÀ DEGLI STUDI DI MILANO
FACOLTÀ DI SCIENZE E TECNOLOGIE

Corso di Laurea in
"Sicurezza dei Sistemi e delle Reti Informatiche"

STUDIO ED ANALISI DI ATTACCHI DI RETE DoS CON PROTOCOLLO NETFLOW

Relatore: Prof. Andrea Lanzi

Tesi di:
Giulio VASSALLO
Matricola: 909212

Anno Accademico 2023/2024

RINGRAZIAMENTI:

Dedico il successo ottenuto alla mia determinazione e resilienza, ai miei figli Davide e Arianna, a mia moglie Marzia che mi ha permesso di studiare, a mio fratello, mia madre, mio padre, agli amici e tutti quelli che hanno creduto in me.

Ringrazio il mio compagno Davide che mi ha accompagnato in questo lungo e faticoso percorso, nei momenti più difficili, la sua forza e l'ottimismo sono stati indispensabili, mettendo a disposizione in maniera generosa le sue conoscenze nel campo IT e non facendomi mai pesare la grande differenza d'età.

Ringrazio Sabrina Papini che in molte occasioni ha fatto da tramite tra me e la segreteria o i professori, aiutandomi a pianificare gli esami e i piani di studio.

Ringrazio Il prof. Andrea Lanzi per il supporto e l'esperienza messa a disposizione durante il periodo di tirocinio.
Ringrazio il prof. Luca Deri e la sua squadra per il supporto al software Ntopng e le relative licenze.

PREFAZIONE

Il mondo è sempre più interconnesso, la proliferazione delle reti digitali porta con sé una serie di sfide significative, tra le quali spicca la crescente minaccia degli attacchi di tipo Denial of Service (DoS). Questa forma di attacco non richiede alcuna raffinatezza, se non l'impiego di grandi risorse per inondare di traffico il bersaglio, con l'unico scopo di sovraccaricare le risorse di rete al fine di renderle inaccessibili agli utenti legittimi, è un pericolo pressante per l'affidabilità e la sicurezza delle infrastrutture digitali.

Questo lavoro di tesi si propone di studiare ed analizzare il fenomeno degli attacchi DoS, sfruttando il protocollo NetFlow per la rilevazione e l'analisi di tali minacce. Il protocollo NetFlow, è stato sviluppato originariamente da Cisco, ed è un ottimo strumento per ottenere una comprensione dettagliata del traffico di rete, senza appesantire troppo le risorse di un sistema.

Lo scopo di questa tesi è realizzare una rete virtuale nella quale vi è un host malevolo che esegue attacchi DoS, si osserveranno per ciascun attacco le caratteristiche servendoci del software di monitoraggio del traffico di rete "Ntopng", che elabora i flussi NetFlow inviati da un Router Cisco e genera allarmi e avvertimenti; infine, Wireshark, un software di analisi di rete open-source ci consentirà di catturare e analizzare il traffico dei pacchetti sulla rete.

A seguito di eventuali allarmi o avvertimenti di Ntopng si eseguiranno delle catture del traffico con Wireshark per capire se il server è sotto attacco o meno.

Hanno una grande importanza gli allarmi di Ntopng perché grazie ad essi è possibile verificare tempestivamente eventuali anomalie nel traffico di rete, inclusi comportamenti non tipici o flussi di dati insoliti, dati su indirizzi IP, protocolli e volumi del traffico, eventuali congestioni e minacce alle prestazioni e sicurezza del sistema.

Sommario

STUDIO ED ANALISI DI ATTACCHI DI RETE DoS CON NETFLOW PROTOCOL

CONTESTO E MOTIVAZIONI

INTRODUZIONE ALLA SICUREZZA INFORMATICA:

Nel contesto della sicurezza informatica, l'analisi del traffico di rete ricopre un ruolo primario. Monitorare e interpretare il flusso di dati è un lavoro complesso e di cruciale importanza per identificare e prevenire potenziali attacchi, e per garantire un ambiente digitale sicuro ed affidabile.
Negli ultimi anni l'importanza dell'analisi del traffico di rete è diventata sempre più importante a causa del preoccupante aumento degli attacchi informatici e delle minacce alla sicurezza online. Le ragioni di questo aumento é determinata da diverse ragioni tra cui:

- CRESCENTE DIPENDENZA DALLA TECNOLOGIA: aumento della dipendenza da device e conseguente connettività ad Internet, questo consente agli attaccanti di avere maggiori opportunità.
- SOFISTICAZIONE DEGLI ATTACCHI: gli attacchi informatici diventano sempre più sofisticati. I criminali informatici sviluppano tecniche avanzate, come attacchi di ingegneria sociale più convincenti, malware più evoluti e attacchi mirati per aggirare le difese di sicurezza.
- MOTIVAZIONI VARIE: furto di dati personali, finanziari o aziendali, spionaggio industriale, il ransomware, il sabotaggio e l'attivismo digitale. Questa diversità di intenti rende gli attacchi più diffusi e difficili da prevenire.
- CRESCITA DELL'INTERNET DELLE COSE (IOT): L'espansione dell'IoT (Internet of things) ha introdotto un grande numero di dispositivi collegati alla rete che spesso mancano di adeguati standard di sicurezza. Questi dispositivi possono essere usati come punto d'accesso per portare a segno attacchi più ampi.
- GLOBALIZZAZIONE DELLE MINACCE: la rete Internet rende le minacce globali, permettendo agli attaccanti di agire da qualsiasi parte del mondo.
- MANCANZA DI CONSAPEVOLEZZA E FORMAZIONE: La mancanza di consapevolezza e formazione sulla sicurezza informatica è un fattore critico. Gli utenti spesso cadono vittima di attacchi di phishing o ignorano le best practices di sicurezza, rendendo più facile per gli attaccanti penetrare nei sistemi.
- COMPLICAZIONI LEGATE ALLA CRITTOGRAFIA: la crittografia è un importante strumento per la sicurezza online, tuttavia, può anche essere utilizzata dai malintenzionati per nascondere le loro attività. La crittografia end-to-end può rendere difficile per le autorità monitorare e prevenire attività illegali online.

MINACCE IN COSTANTE EVOLUZIONE:

La crescente complessità e interconnessione del mondo digitale aumenta il rischio di attacchi per questo è necessario porre sempre maggiore attenzione alla sicurezza informatica, investendo in tecnologie avanzate e formazione degli utenti. La collaborazione tra settore pubblico e privato è importante per affrontare queste sfide in continua evoluzione.

RUOLO DELLE RETI E DEL TRAFFICO DI RETE:

Le reti consentono la trasmissione rapida e affidabile dei dati, facilitano la comunicazione e supportano una vasta gamma di applicazioni, dallo scambio di informazioni personali all'esecuzione di processi aziendali complessi. Tuttavia, questa interconnessione rende anche le reti suscettibili ad attacchi informatici, e il traffico di rete gioca un ruolo chiave nel comprendere e contrastare tali minacce. Ecco come:

- MONITORAGGIO DEL TRAFFICO: Il traffico di rete rappresenta il flusso di dati tra dispositivi e server, monitorando attentamente questo traffico, è possibile individuare anomalie o comportamenti sospetti che potrebbero indicare un potenziale attacco.
- IDENTIFICAZIONE DELLE MINACCE: analizzando il traffico di rete, è possibile identificare attività sospette, come tentativi di accesso non autorizzato, malware o comportamenti anomali. Il traffico insolito può essere un indicatore di un attacco in corso o di un tentativo di violazione della sicurezza.
  Gli attacchi DoS/DDoS (Distributed Denial of Service) possono essere rilevati monitorando i picchi di traffico anomalo che cercano di sopraffare i sistemi di destinazione.
- RISPOSTA AGLI INCIDENTI: la comprensione del traffico di rete è fondamentale per una risposta rapida agli incidenti. Identificare tempestivamente un attacco consente di isolare le minacce, mitigare i danni e ripristinare la sicurezza. Le informazioni sul traffico di rete possono essere utilizzate per tracciare l'origine di un attacco, identificare le vulnerabilità sfruttate e sviluppare contromisure adeguate.
- PROTEZIONE DEI DATI SENSIBILI: Il traffico di rete può contenere informazioni sensibili, monitorare attentamente questo traffico aiuta a identificare e proteggere i dati sensibili contro la perdita, il furto o l'accesso non autorizzato.

In sintesi, il traffico di rete è una fonte preziosa di informazioni per comprendere, rilevare e contrastare gli attacchi informatici. Un'analisi accurata di questo traffico consente alle organizzazioni di rafforzare la loro sicurezza e rispondere efficacemente alle minacce emergenti.

ATTACCHI DOS / DDOS

DoS

DoS acronimo di Denial of Service è un attacco informatico che ha lo scopo di rendere una risorsa non disponibile, o di rendere un servizio gravemente rallentato agli utenti legittimi. L'obiettivo consiste nell'esaurire le risorse di un sistema (server), ovvero la larghezza di banda di rete, la capacità di elaborazione della CPU o la memoria del sistema, limitando o negando l'accesso ai client legittimi.
Gli attacchi DoS possono esse effettuati da un singolo host che è potenzialmente rintracciabile.
Modalità operative degli attacchi DoS:

- FLOOD DI TRAFFICO**:** gli attacchi DoS spesso inondano (flooding) il sistema bersaglio con traffico di rete in eccesso. Ad esempio, un attacco SYN Flood mira a sovraccaricare il sistema con richieste SYN (primo step del three-way handshake necessario per una connessione TCP), lasciando il sistema in attesa di risposte che non arrivano mai.
- ATTACCHI DI ESAURIMENTO DELLE RISORSE**:** gli attacchi DoS possono anche mirare ad esaurire le risorse del sistema, tra cui memoria o CPU, raggiungendo comunque lo scopo di rendere il sistema inaccessibile.
- ATTACCHI DI PROTOCOLLO: alcuni attacchi DoS sfruttano vulnerabilità specifiche nei protocolli di comunicazione. Per esempio, un attacco ICMP flood inonda il sistema target con pacchetti ICMP, consumando la larghezza di banda.
- ATTACCHI DI RALLENTAMENTO**:** Alcuni attacchi DoS non interrompono completamente il servizio, ma cercano di renderlo gravemente rallentato, rendendo così l'accesso ai servizi scadente per gli utenti legittimi.

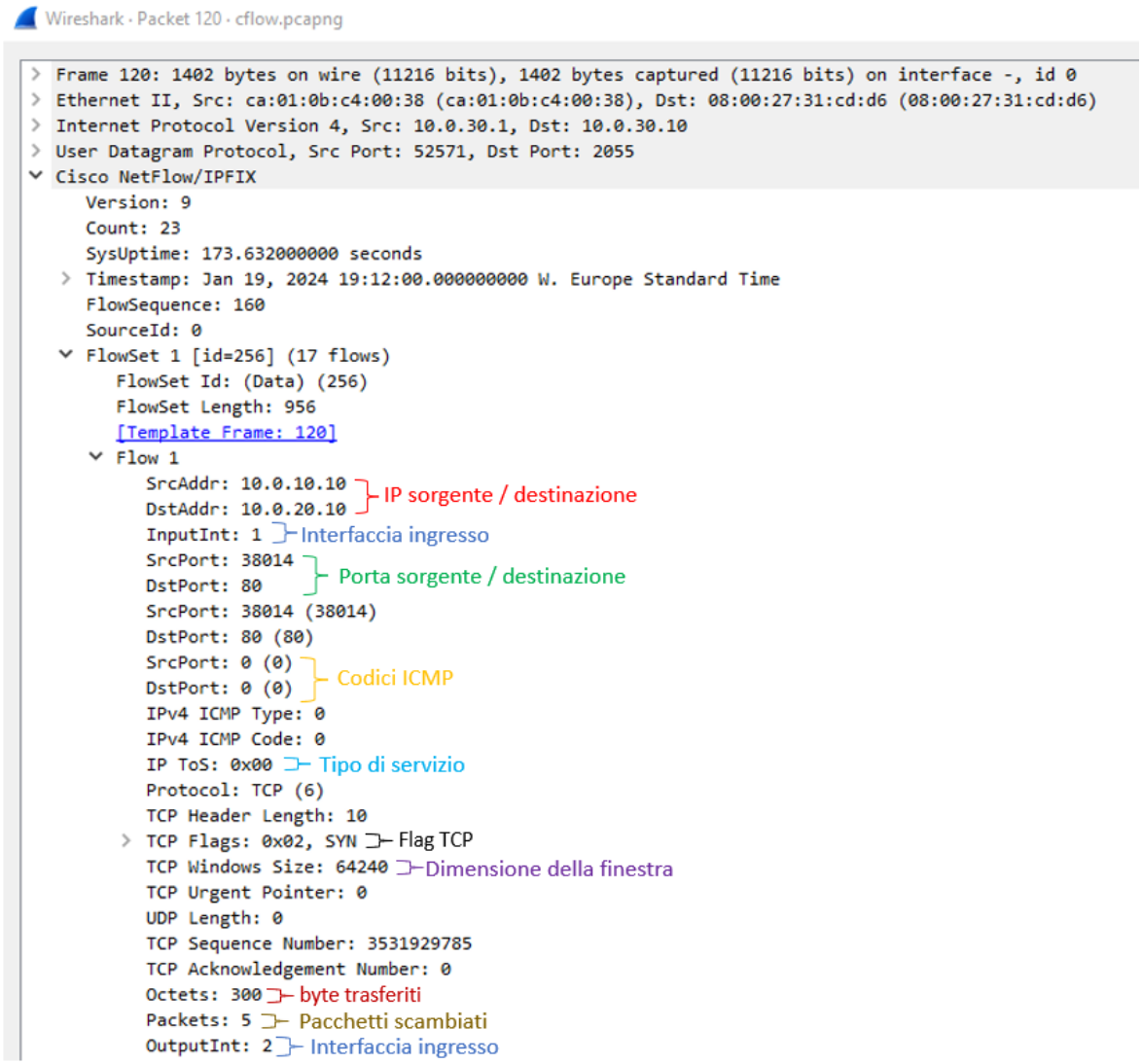
DDoS

DDoS (distribuited denial of service) è un attacco DoS che coinvolge molteplici dispositivi distribuiti in varie posizioni geografiche per aumentare la potenza dell'attacco. Spesso questi sistemi compromessi fanno parte di una BotNet, ossia un gruppo di macchine infettate da malware che rispondono ai comandi dell'attaccante, ma possono anche agire autonomamente. Lo scopo è il medesimo di un attacco DoS ovvero esaurire le risorse di un sistema (server), come la larghezza di banda di rete, la capacità di elaborazione della CPU o la memoria del sistema, limitando o negando l'accesso ai client legittimi.

Per proteggersi da attacchi DoS/DDoS, le organizzazioni spesso implementano misure di sicurezza, come firewall, sistemi di rilevamento e prevenzione delle intrusioni (IDS Intrusion Detection System / IPS Intrusion Prevention System).
Nella mia simulazione utilizzerò NTOPNG un'applicazione per l'analisi e il monitoraggio di rete che è in grado di analizzare in tempo reale, ma anche offline i flussi NetFlow in arrivo da un router Cisco.

IMPORTANZA DEL PROTOCOLLO NETFLOW:

NetFlow è un protocollo sviluppato da Cisco per la raccolta, il monitoraggio e l'analisi del traffico di rete. Netflow viene usato per avere informazioni dettagliate sulle attività di rete, per l'ottimizzazione delle prestazioni, la sicurezza, per risolvere problemi legati all'uso della banda, del traffico e ai rilevamenti di comportamenti anomali. Netflow può inviare molte informazioni, di seguito le principali:



I dati sono inviati dal router o altro dispositivo di rete a un collector tramite datagrammi (UDP).
NetFlow ha un impatto molto ridotto sulla larghezza di banda e in termini di overhead di rete, per queste caratteristiche è molto efficiente. Di seguito alcune caratteristiche di NetFlow:

- **Rappresentazione Aggregata dei Dati:** NetFlow invia informazioni parziali per ogni pacchetto attraverso la rete, questo lo rende leggero ed efficiente. Invia informazioni aggregate sotto forma di "flussi". Ogni flusso è rappresentato da una sequenza di pacchetti che hanno caratteristiche comuni, come protocolli, indirizzi IP e porte sorgente/destinazione.
- **Campionamento:** Il campionamento consente di raccogliere solo una parte dei pacchetti riducendo ulteriormente l'overhead, conservando comunque un campione rappresentativo del traffico.
- **Efficienza del Protocollo:** NetFlow è progettato per essere leggero ed efficiente. Gli header dei pacchetti sono strutturati per avere il minimo overhead, e i dati vengono trasmessi in modo compatto per aumentare l'efficienza della comunicazione.

Queste caratteristiche consentono di monitorare ed analizzare grandi quantità di dati, senza sovraccaricare le risorse del sistema, e questo lo rendono molto funzionale al nostro scopo.

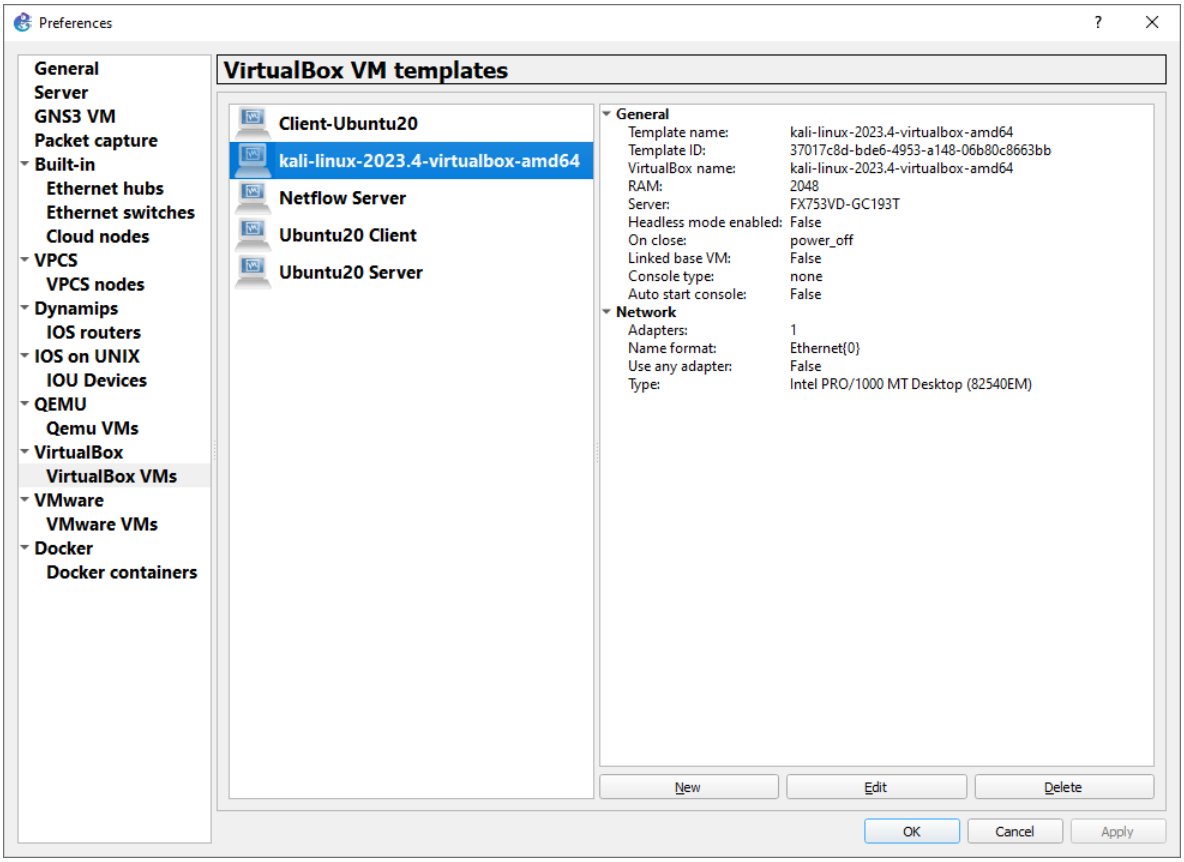
IMPLEMENTAZIONE INFRASTRUTTURA E SOFTWARE UTILIZZATI

GNS3

Per la realizzazione di una rete virtuale, abbiamo utilizzato il software GNS3 (https://www.gns3.com/).
GNS3 è un software open source che simula reti complesse in maniera realistica senza aver bisogno di hardware di rete dedicato. Possiede un'interfaccia grafica e funziona su più sistemi operativi, è basata su altri software:

- Dynamips, un software in grado di emulare il sistema operativo Cisco IOS
- VirtualBox, un software gratuito e open source per l'esecuzione di macchine virtuali (vedi di seguito)

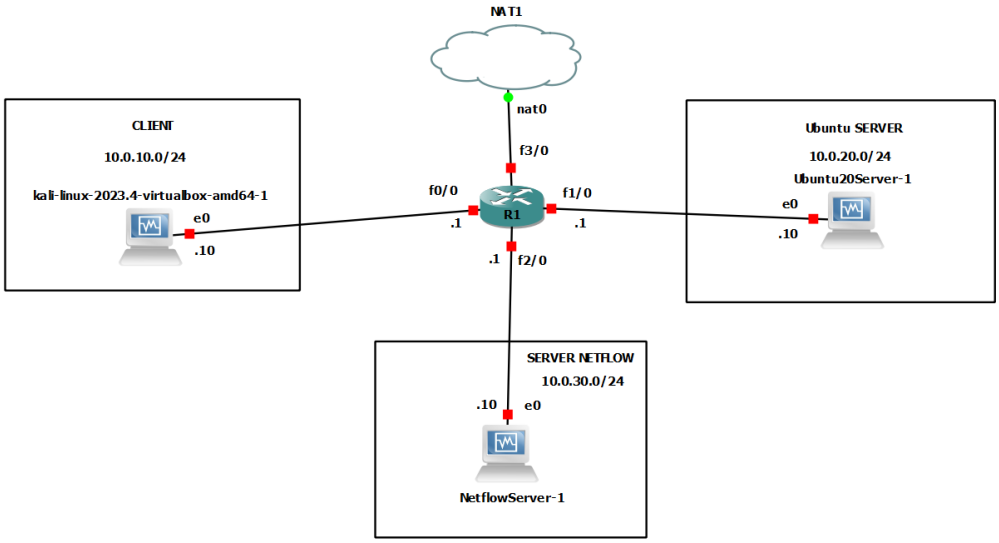
L'integrazione GNS3 – VirtualBox, permette di collegare e collocare le macchine virtuali all'interno di una rete composta da apparati di rete virtuali Cisco (router e switch).



Integrazione GNS3 - VirtualBox

All'interno della rete vi sono:

- Una rete client *(10.0.10.0/24)* nella quale risiede un client avente sistema operativo Kali Linux, distribuzione Unix conosciuta molto per avere built-in diversi strumenti di penetration testing.
- Una rete server *(10.0.20.0/24)* nella quale risiede un server che può offrire servizi Web, questo è il server bersaglio, per semplificare lo scenario, la rete non è protetta da firewall.
- Una rete Server NetFlow *(10.0.30.0/24)* nella quale risiede il nostro server NetFlow che riceve e colleziona i dati NetFlow inviati dal router Cisco.
  Qui abbiamo collegato un server Ntopng *(https://www.Ntopng.org/)* in grado di "leggere" ed elaborare i dati NetFlow inviati dal router Cisco.
- Una rete di NAT *(192.168.56.0/24)* utilizzata dal software GNS3 per realizzare l'uscita e la navigazione in Internet mediante il device "Cloud NAT"
- Un Router Cisco che effettua il routing fra le varie subnet sopra menzionate, campiona i flussi e ne invia il loro dettaglio al server NetFlow.



Layout di rete

NTOPNG

Per l'elaborazione dei dati NetFlow abbiamo deciso di utilizzare il software Ntopng.
Ntopng è un'applicazione di monitoraggio del traffico di rete open-source progettata per fornire informazioni dettagliate sulle attività di rete. La sua principale funzionalità è l'analisi del traffico in tempo reale, consentendo agli amministratori di rete di ottenere una visione approfondita delle attività di rete, delle prestazioni e dei possibili problemi di sicurezza.

Ntopng può leggere i dati NetFlow nelle due seguenti modalità principali:

1. **NetFlow Collector (NetFlow Collector Mode):**
   In questa modalità, Ntopng agisce come un collector NetFlow. Riceve e analizza i flussi NetFlow inviati dai dispositivi di rete, come router e switch, che generano tali dati. Questi flussi NetFlow contengono informazioni sul traffico di rete, come indirizzi IP sorgente e di destinazione, porte, protocolli, durata delle connessioni e quantità di dati trasferiti. Ntopng elabora questi flussi NetFlow per generare rapporti dettagliati sul traffico di rete.
   Questa è la modalità utilizzata nella nostra infrastruttura.

2. **NetFlow Probing (NetFlow Probing Mode):**
   In questa modalità, Ntopng si comporta come un generatore di flussi NetFlow (NetFlow probe). È in grado di inviare pacchetti NetFlow verso il dispositivo di rete da monitorare e analizzarne le risposte per ottenere informazioni sul traffico. Questo approccio è particolarmente utile quando i dispositivi di rete non supportano la generazione di flussi NetFlow in modo nativo, consentendo comunque il monitoraggio del traffico.

Entrambe le modalità consentono a Ntopng di fornire informazioni dettagliate sul traffico di rete, identificando pattern, anomalie, e contribuendo alla gestione efficiente delle risorse di rete. La scelta tra NetFlow Collector Mode e NetFlow Probing Mode dipende dalle esigenze specifiche dell'ambiente di rete e dalla disponibilità di funzionalità NetFlow nei dispositivi di rete coinvolti.

Nella nostra implementazione abbiamo installato il software Ntopng su un server Ubuntu linux 20.04lts *(https://www.Ntopng.org/guides/Ntopng/what_is_Ntopng.html#installing-on-linux)* con licenza Enterprise M gentilmente concessa dal professore e fondatore Luca Deri ed il suo team, la quale ringraziamo moltissimo per aver supportato il progetto.
La licenza indicata ci permette di non avere limiti sul numero dei flussi NetFlow gestibili e di sfruttare alcune funzionalità aggiuntive come la gestione degli Alerts *(https://www.Ntopng.org/products/traffic-analysis/Ntopng/)*.

VIRTUALBOX

Oracle VM VirtualBox è un software gratuito e open source per l'esecuzione di macchine virtuali (con una versione ridotta distribuita secondo i termini della GNU General Public License) per architettura x86 e 64bit che supporta Windows, GNU/Linux e macOS come sistemi operativi host, ed è in grado di eseguire Windows, GNU/Linux, OS/2 Warp, BSD come ad esempio OpenBSD, FreeBSD e infine Solaris e OpenSolaris come sistemi operativi guest.

Nella nostra infrastruttura abbiamo installato Virtualbox su PC host Windows ed abbiamo installato le seguenti macchine virtuali:

- Macchina virtuale Kali Linux che come indicato sopra simula il client malevolo e/o infetto all'interno della rete client
- Macchina virtuale Ubuntu con installati i software Ntopng ed Nprobe utile alla ricezione e gestione dei flussi NetFlow
- Macchina virtuale Ubuntu server che ha il ruolo di target/bersaglio

INDIRIZZAMENTI DI RETE

Descrizione	Network Address	Subnet Mask	Host
Client Network	10.0.10.0	255.255.255.0	Router 10.0.10.1
			Client Kali Linux 10.0.10.10
Server Network	10.0.20.0	255.255.255.0	Router 10.0.20.1
			Server Ubuntu 10.0.20.10
NetFlow Server Network	10.0.30.0	255.255.255.0	Router 10.0.30.1
			NetFlow server Ntopng 10.0.30.10
Internet Network	192.168.56.0	255.255.255.0	Router (DHCP)

ALERT E WARNING RIVELATI DA NTOPNG DURANTE LA SIMULAZIONE

Ntopng andrebbe configurato e ottimizzato per adattarsi al meglio alle proprie esigenze di monitoraggio e risorse disponibili. Durante questo studio sono stati attivati la maggior parte degli allarmi e dei warning con i valori di default. Ovviamente ogni allarme e warning deve essere valutato nel contesto e nella topologia di rete in cui Ntopng lavora.

Elenco di rilevamenti di Ntopng durante le simulazioni:

SYN FLOOD

Un attacco SYN flood sfrutta una debolezza nel processo di connessione tra computer su Internet, noto come "Three-way handshake" nel protocollo TCP. Il funzionamento in breve:

1. Richiesta di connessione: Un computer malintenzionato invia al server una grande quantità di richieste di connessione (pacchetti SYN), ma non completa il processo di connessione.
2. Server in attesa: Il server riceve le richieste di connessione e risponde con un pacchetto di conferma (SYN-ACK), ma il computer malintenzionato non invia mai la conferma finale.
3. Risorse esaurite: Il server rimane in attesa di una conferma che non arriva mai, impegnando le sue risorse. Con un numero sufficiente di richieste di questo tipo, il server si sovraccarica e non riesce a gestire nuove connessioni legittime.

In breve, l'attacco SYN flood satura il server con richieste di connessione false, impedendogli di gestire le connessioni reali e causando un'interruzione del servizio. Esistono 2 tecniche per portare a segno tale attacco:

1. Utilizzando lo spoofing, ossia camuffando il proprio indirizzo IP con uno diverso, così il server invierà la risposta SYN-ACK a un IP falso.
2. Il client non risponde di proposito al server lasciando la connessione aperta

NTOPNG può attivare un avviso quando il numero di SYN/sec inviati/ricevuti supera la soglia > 256 SYN/sec. È possibile modificare i valori di soglia.

TCP SYN SCAN

Il SYN Scan è un tipo di scansione che consiste nell'invio di pacchetti TCP con flag SYN attivo.
Se la porta da controllare è aperta l'attaccante riceverà in risposta un pacchetto TCP con i flag SYN e ACK attivi, al quale si risponderà chiudendo la connessione con un pacchetto TCP con flag RST attivo.
Se la porta da controllare è chiusa, l'attaccante riceverà un pacchetto TCP con flag RST attivo che chiuderà la connessione.

In entrambi i casi, la connessione non verrà mai completata e per questa ragione difficilmente comparirà nei file di log.

Ntopng può attivare un avviso quando il numero di SYN inviati/ricevuti/min (senza risposta) supera la soglia > 256 SYN/min. È possibile modificare i valori di soglia.

TCP FIN SCAN

Il FIN scan è un tipo di scansione che consiste nell'invio di pacchetti TCP alle porte della vittima, aventi il solo flag FIN attivo. Nelle specifiche tecniche della RFC793 un host che riceve un pacchetto con flag FIN attivo, deve rispondere con un pacchetto con flag RST attivo, qualora la porta sia chiusa, mentre se fosse aperta, il pacchetto andrebbe ignorato. Questo consente all'attaccante di sapere quali porte sono aperte. Non tutti i sistemi osservano le specifiche tecniche e restituiscono in ogni caso un pacchetto TCP con flag RST attivo rendendo la scansione inutile.

Ntopng può attivare un avviso quando il numero di FIN inviati/ricevuti/min (senza risposta) supera la soglia > 256 FIN/min. È possibile modificare i valori di soglia.

SCAN DETECTED

Ntopng può attivare un avviso quando viene rilevata una scansione (host/porta) se il numero di flussi TCP/UDP incompleti supera il limite specificato... > 32 Flows (Minute). È possibile modificare i valori di soglia.

FLOWS FLOOD

Ntopng può attivare un avviso quando i nuovi flussi client/server/sec supera la soglia > 256 flussi/sec (minuto). È possibile modificare i valori di soglia.

UNEXPECTED SCORE BEHAVIOUR (COMPORTAMENTO DEL PUNTEGGIO INASPETTATO)

Avviso per il comportamento del punteggio anomalo, al fine di rilevare minacce o difetti.
L'allarme può essere attivato quando un comportamento inaspettato arriva dall'interfaccia.

UNEXPECTED TRAFFIC BEHAVIOUR (COMPORTAMENTO DEL TRAFFICO INASPETTATO)

Controlli per Comportamento Inatteso.
Avviso per il comportamento del traffico anomalo, al fine di rilevare minacce o difetti.
L'allarme può essere attivato quando un comportamento inaspettato arriva dall'interfaccia.

SCORE THRESHOLD EXCEEDED (SOGLIA PUNTEGGIO SUPERATA)

Ogni host ha un valore numerico non negativo utilizzato per memorizzare il valore del punteggio. Questo valore viene calcolato su un intervallo di tempo di 1 un minuto. Quando il punteggio di un host supera la soglia del punteggio > 5000 in un minuto, viene attivato l'avviso. È possibile modificare i valori di soglia.

COME VIENE CALCOLATO LO SCORE IN NTOPNG

Lo score è un indicatore numerico e se diverso da zero è presente qualche tipo di problema, maggiore è il punteggio e peggiore è il problema associato.
Sono tre le fonti principali del punteggio:

1. Lo score del flusso: che indica quanto è grave, ad esempio:
   un flusso con molteplici ritrasmissioni ha un punteggio di flusso non nullo.
2. Lo score dell'host di origine del flusso: un valore numerico associato all'host di origine, ad esempio:
   se nel flusso al punto 1 le ritrasmissioni sono solo destinazione -> client, lo score dell'host sorgente per questo flusso sarà zero, ma se anche la sorgente ha ritrasmissioni il valore sarà positivo.
3. Lo score dell'host di destinazione del flusso: lo stesso di prima ma per l'host di destinazione del flusso.
   Poiché i flussi possono avere più problemi, ogni problema riscontrato su un flusso (ad esempio una ritrasmissione TCP o una versione TLS obsoleta rilevata su un flusso) contribuisce al punteggio.
   Pertanto, lo score è la somma degli score flusso/sorgente/destinazione individuali riscontrati su tale flusso.

WARNING: RECENTLY LIVE FLOWS

Il messaggio "Warning Recently live flows" indica che Ntopng sta osservando un numero significativo di flussi di dati attivi nella rete e quindi un alto utilizzo delle risorse e di conseguenza preoccupazioni per la sicurezza; quindi, è necessario fare indagini a riguardo.
Per affrontare questa avvertenza, potrebbe essere necessario:

Esaminare l'attuale attività di rete per capire se i modelli di traffico sono previsti o se ci sono anomalie. Cercare eventuali minacce alla sicurezza o comportamenti di rete anomali.

SPIEGAZIONE DEGLI ATTACCHI E RILEVAMENTI DI NTOPNG

Teoria di ciascun attacco e sua realizzazione pratica, rilevamenti di Ntopng:

ATTACCHI CON “SLOWHTTPTEST”

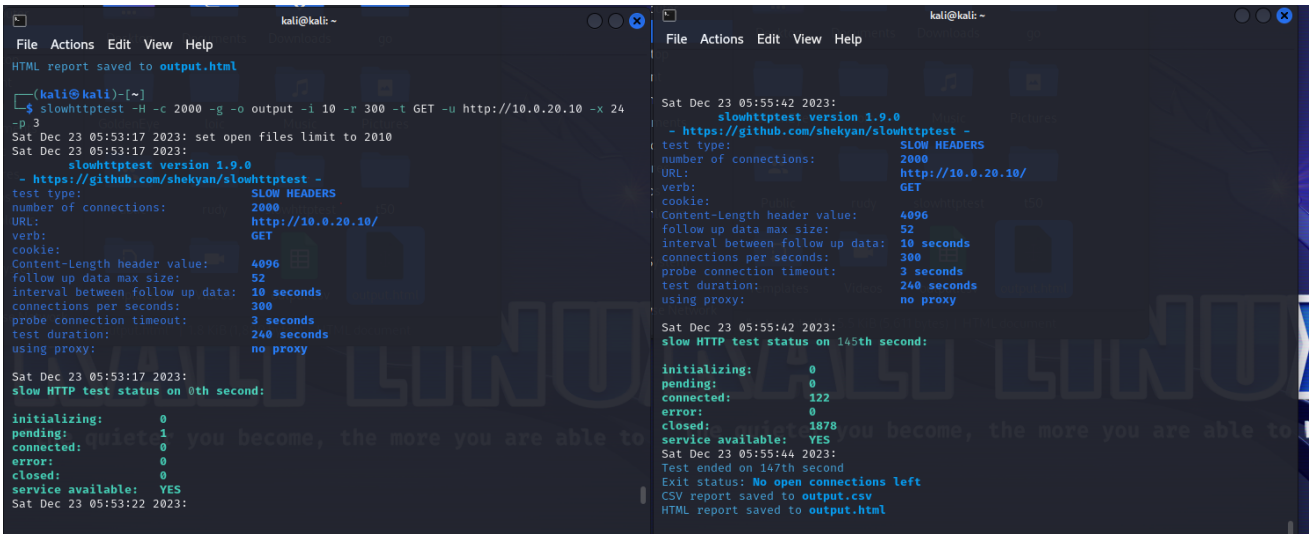
- SLOW HEADERS A.K.A. SLOWLORIS (attacco predefinito)

L’attacco Slow Header consente a un singolo client di saturare le risorse di un server utilizzando una minima ampiezza di banda. Questo è possibile mantenendo le connessioni aperte il più possibile con il server bersaglio, il client malevolo si connette al server target e gli invia richieste parziali, ossia senza terminare l’invio delle intestazioni http; nello specifico quando un client effettua una richiesta HTTP, il server web può chiudere la connessione TCP solo quando l’invio dell’intestazione (header) della richiesta è completa. Questa tecnica obbliga i server a mantenere le connessioni aperte, in questo modo è possibile raggiungere il numero massimo di connessioni disponibili del server e renderlo inaccessibile alle richieste legittime di altri client.

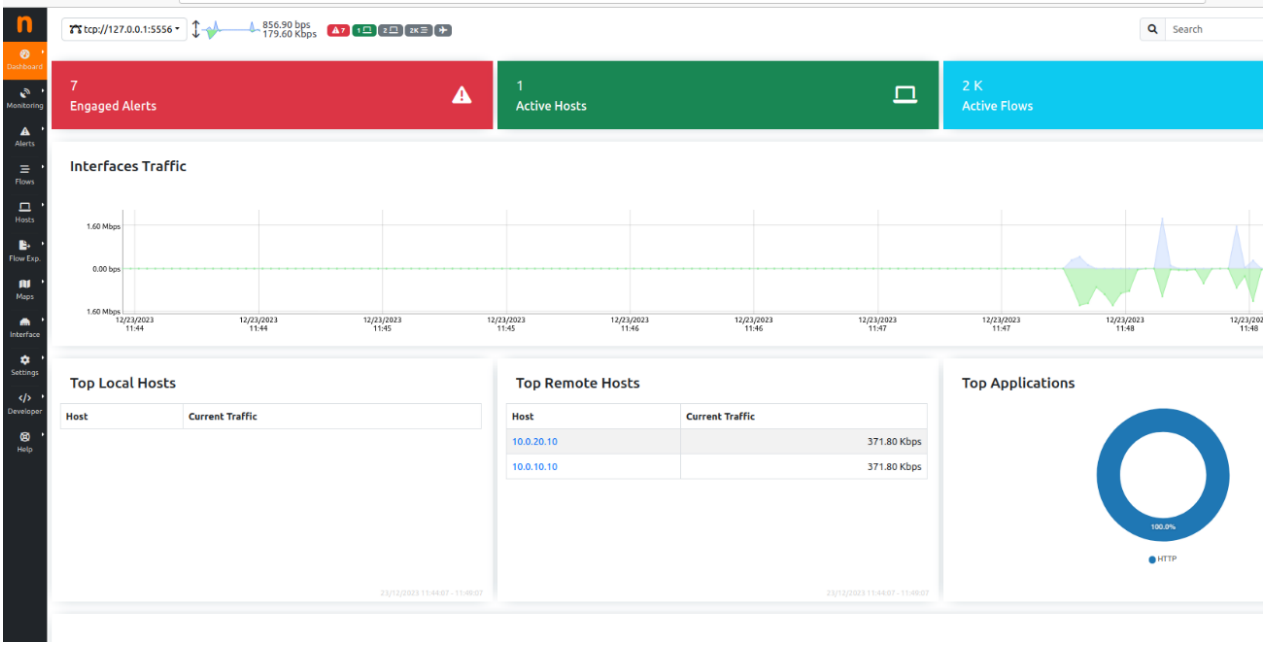
Ecco un esempio del comando Slow Header:

slowhttptest -H -c 2000 -g -o output -i 10 -r 300 -t GET -u http://10.0.20.10 -x 24 -p 3

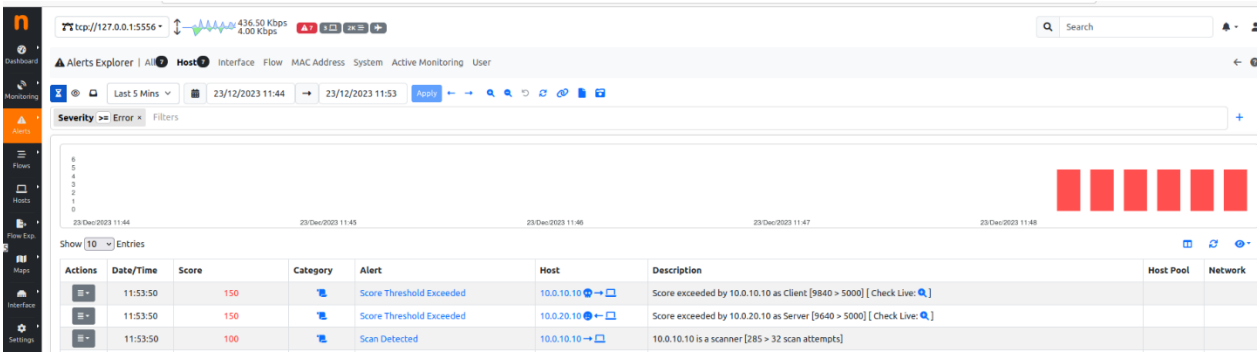
-H Avvia il tipo di attacco predefinito slowloris,
-c con 2000 connessioni,
-o -g vengono create delle statistiche e salvate nell' output specificato “output.csv” e “output.html”,
-i l'intervallo tra i follow-up header è di 10 secondi,
-r la velocità di connessione è di 300 connessioni al secondo,
-t specifica il metodo da utilizzare nella richiesta,
-x massima lunghezza random in byte di ogni coppia nome/valore dei follow-up data, quando SlowHTTPTest esegue un attacco, può inviare dati di follow-up per mantenere le connessioni aperte; questi dati possono essere inclusi nelle intestazioni HTTP o nei dati del corpo della richiesta POST, l'opzione -x controlla la lunghezza massima di ciascuna coppia di dati di follow-up.
-p imposta l'intervallo di attesa per la risposta HTTP sulla connessione sonda, dopodiché il server verrà considerato inaccessibile.



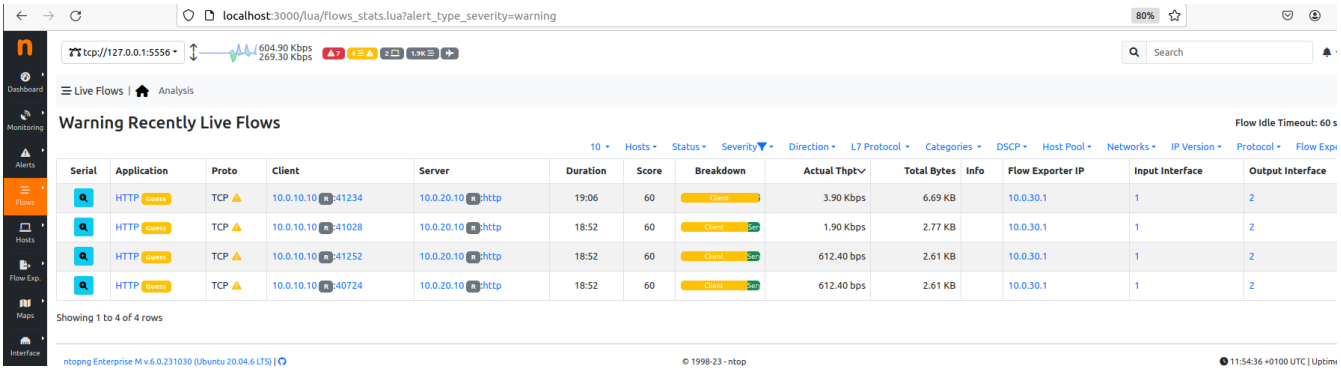
Nella prima immagine è possibile vedere l’inizio dell’attacco e nella seconda la fine, ogni 5 secondi viene aggiornato lo stato con il report in tempo reale delle connessioni stabilite, quelle aperte e quelle chiuse.

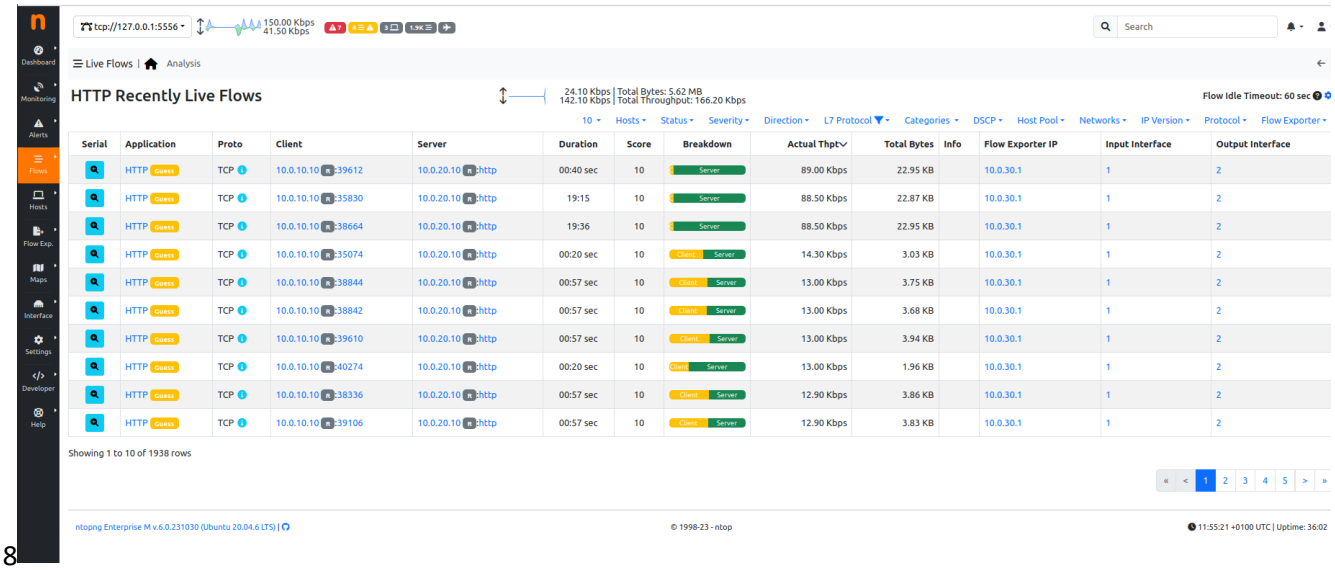


Questa è la schermata principale di Ntopng.



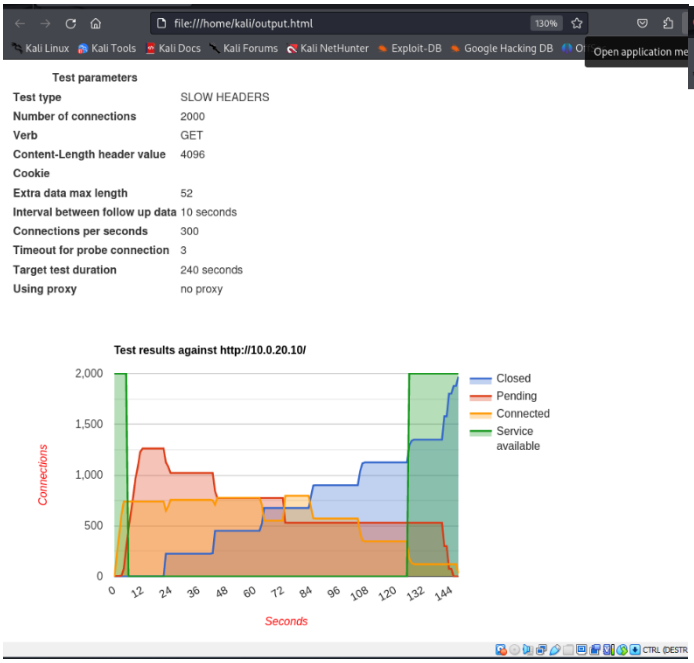
Dopo una trentina di secondi Ntopng rileva uno SCAN DETECTED, nello specifico l'attacco Slow Header non fa una scansione ma Ntopng segnala che tra l'host 10.0.10.10 e il server 10.0.20.10 si sono verificati più di 32 flussi HTTP incompleti. L'alert SCORE THRESHOLD EXCEEDED ovvero "soglia del punteggio superata (spiegato nell'argomento precedente)", viene riportato quando la somma del punteggio flusso / flusso sorgente / flusso destinazione è maggiore del valore impostato in questo caso il valore di default che è 5000.

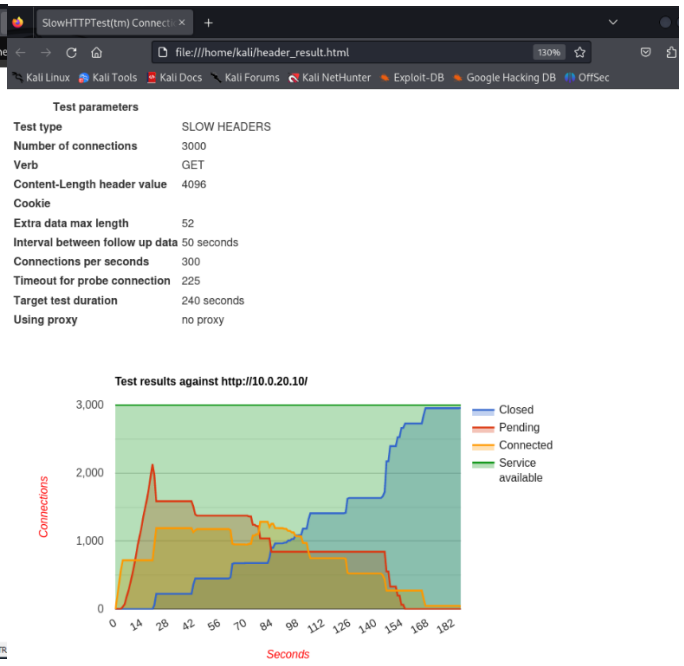




8

Ntopng riporta anche dei warning relativi ai flussi HTTP, questo accade quando è in corso un numero significativo di flussi di dati attivi nella rete e quindi un alto utilizzo delle risorse e di conseguenza preoccupazioni per la sicurezza. Ovviamente è necessaria l'analisi per capire se i modelli di traffico sono previsti o anomali. Nel nostro caso è evidente che c'è un attacco DoS in corso perché i flussi con lo stesso IP sorgente sono centinaia.





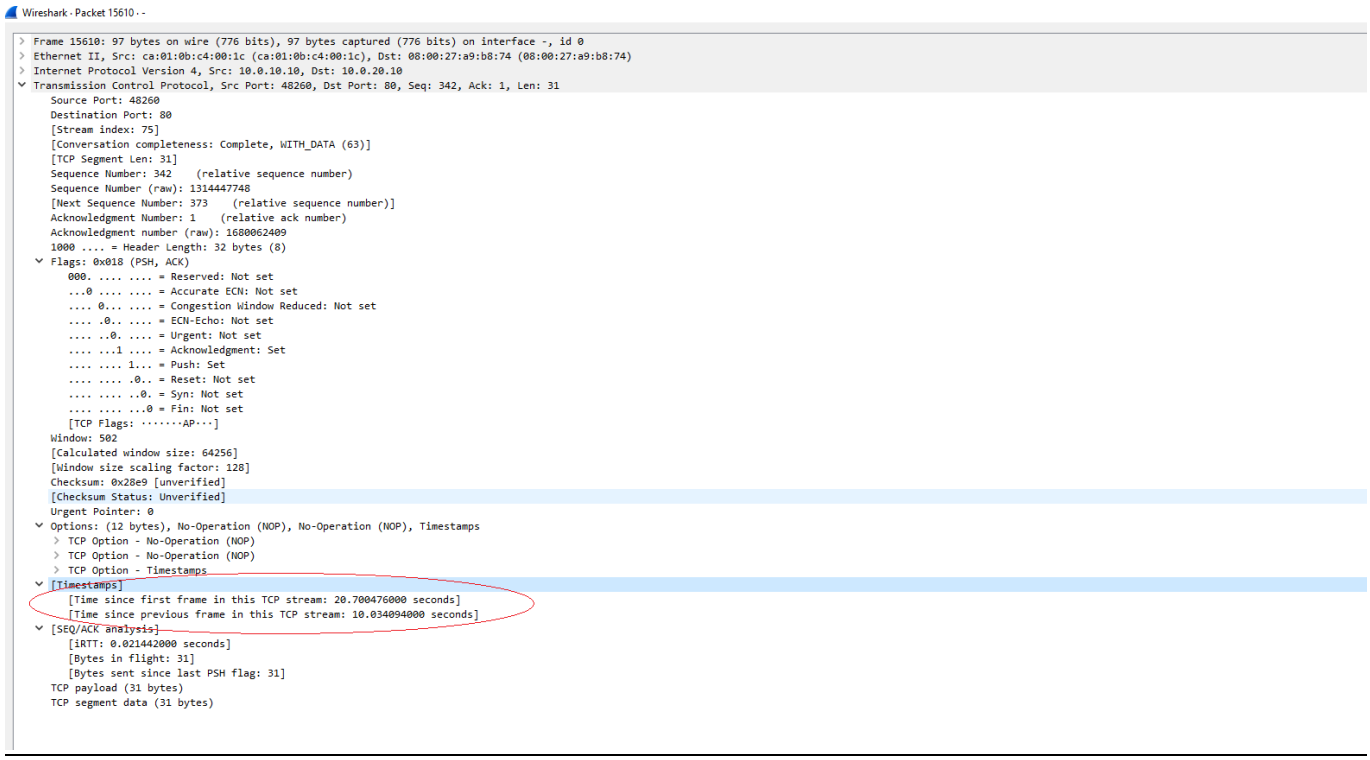
F1 F2

Questi due grafici sono gli output HTML creati grazie alle opzioni -o -g, come si evince da grafico F1 nei primi 10 secondi vengono stabilite 750 connessioni e altre 750 aperte, questo rispecchia le opzioni impostate nel comando, ossia 2000 connessioni a 300 connessioni al secondo. Tuttavia, ben prima delle 2000 connessioni richieste il server risulta NON DISPONIBILE a causa della risposta superiore a 3 secondi, opzione -p 3

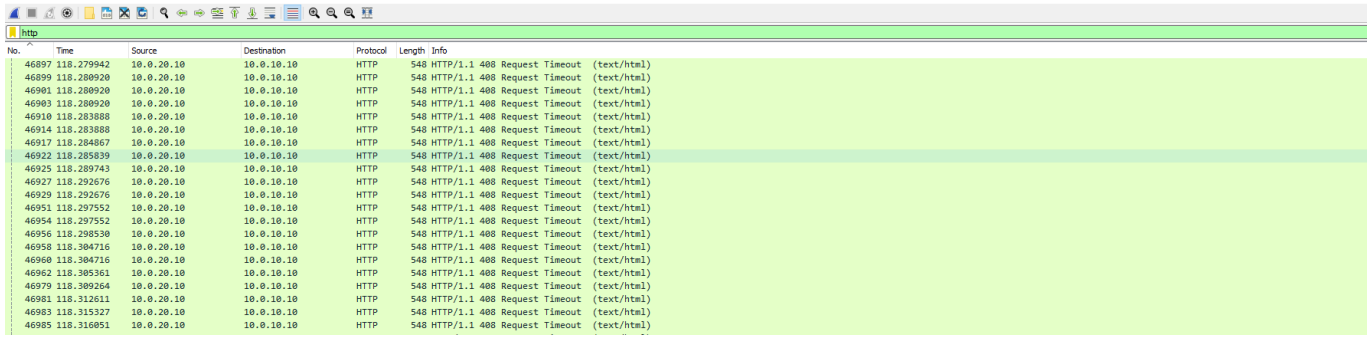
F1→ slowhttptest -H -c 2000 -g -o output -i 10 -r 300 -t GET -u http://10.0.20.10 -x 24 -p 3

Il secondo grafico rappresenta il servizio sempre disponibile (zona verde) questo perché volutamente (a scopo di test) ho impostato -p (l'intervallo di attesa per la risposta http) a 225, nella realtà è impensabile un'attesa così lunga per la risposta da parte di un server.

F2→slowhttptest -H -c 2000 -g -o output -i 10 -r 300 -t GET -u http://10.0.20.10 -x 24 -p 225



I timestamp indicano che tra un frame e l'altro passano 10 secondi, tale intervallo corrisponde all'impostazione -i del comando.



Una piccola cattura ci consente di capire che tutte le richieste HTTP vengono chiuse per timeout e questo significa che le connessioni vengono chiuse forzatamente dal server.
Il codice di stato HTTP 408 "Request Timeout" indica che il server ha terminato la connessione a causa del timeout della richiesta. Questo codice viene restituito quando il server non riceve una richiesta completa dal client entro il tempo massimo consentito. Tutte le richieste finiscono così quindi possiamo dire che il server è sotto attacco DoS a livello di applicazione.

- RANGE ATTACK A.K.A. APACHE KILLER

L'attacco Apache Killer, conosciuto anche come Range Header Attack, è una vulnerabilità che può essere sfruttata contro i server web Apache. L'attacco consiste nell'invio di molteplici richieste a un server web apache contenenti per ognuna un Range Header costruito ad arte.
L'header "Range" in una richiesta HTTP è utilizzato per specificare una determinata gamma o porzione di dati che il client desidera ricevere dal server. Questo può essere utile, ad esempio, per scaricare solo una parte di un file o per riprendere il download da un certo punto.

Range Header malizioso:
l'attaccante specifica un intervallo di byte molto ampio o frammentato, richiedendo una porzione significativa della risorsa. Nel nostro caso è stato impostato il metodo HEAD e le richieste HTTP con tale metodo sono utilizzate per ottenere solo le informazioni di intestazione di una risorsa, senza ricevere la risorsa nel corpo della risposta.

Il server apache, nel tentativo di soddisfare tutte queste richieste, impegna risorse eccessive, portando a un esaurimento delle risorse del server. L'attacco può saturare il server, causando l'interruzione del servizio e di conseguenza negando l'accesso alle risorse ai client legittimi.

di seguito il comando dell'attacco:

slowhttptest -R -g -o output -u http://10.0.20.10 -t HEAD -c 3000 -a 10 -b 10000 -r 500

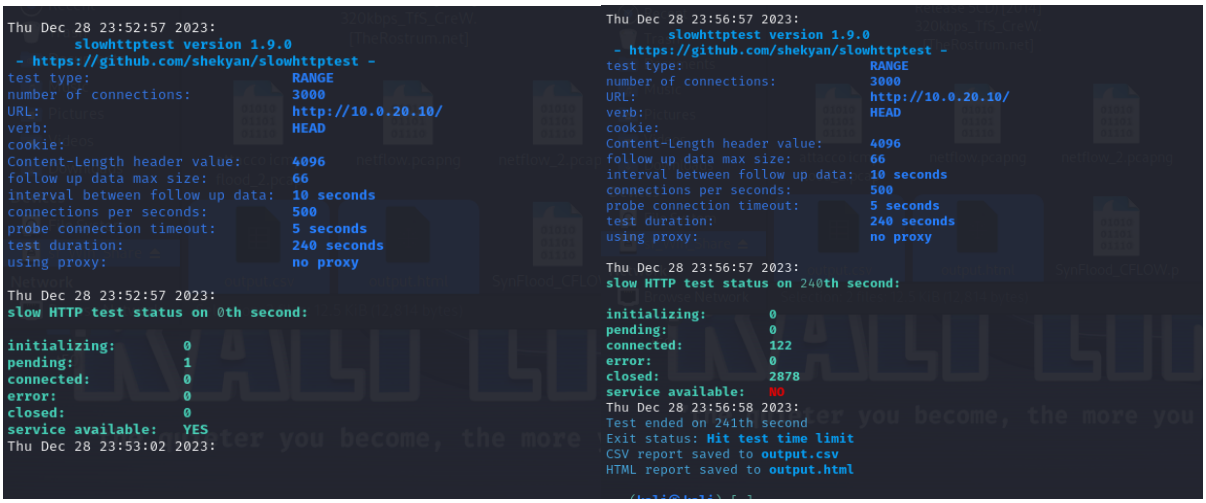
-R attacco di tipo HTTP Keep Alive, il tool invierà dati Range Request header dannosi mantenendo la connessione aperta,

-c avvia il test con 3000 connessioni,

-t utilizza il metodo HEAD per l'invio del Range Header HTTP,

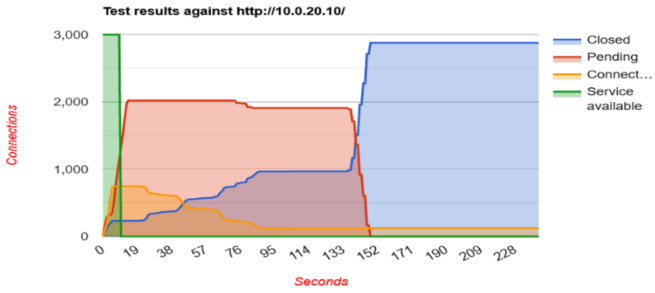
-a -b Intervallo: 0-, x-1, x-2, x-3, … x-y, dove x è il valore di partenza del range (10 nel nostro caso) e y è un numero crescente fino al numero impostato nell'opzione -b (10000 nel nostro caso), con -a si specifica il valore di partenza del range.

-r la velocità di connessione è 500 connessioni per secondo,
l'intervallo tra follow-up data è di 10 secondi.

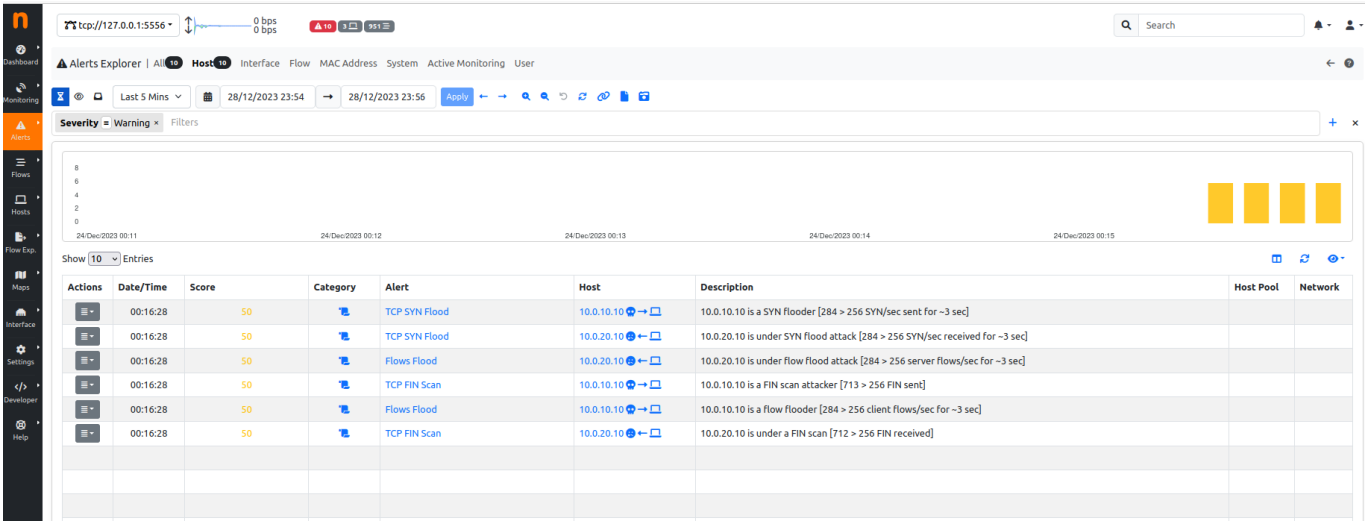


Nelle immagini sopra vediamo la partenza e la fine del test, gli stati vengono restituiti ogni 5 secondi.

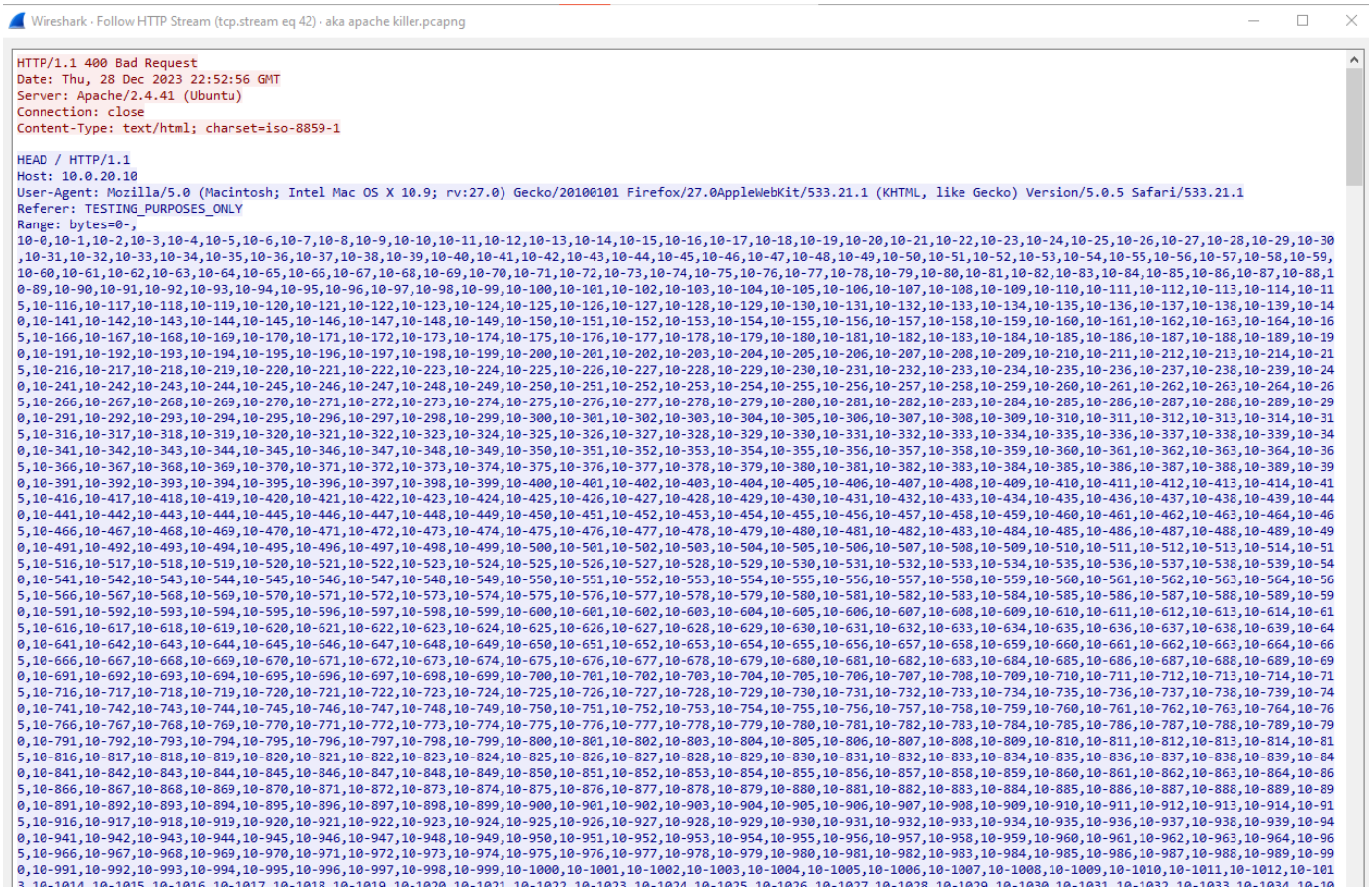
Test parameters	
Test type	RANGE
Number of connections	3000
Verb	HEAD
Content-Length header value	4096
Cookie	
Extra data max length	66
Interval between follow up data	10 seconds
Connections per seconds	500
Timeout for probe connection	5
Target test duration	240 seconds
Using proxy	no proxy



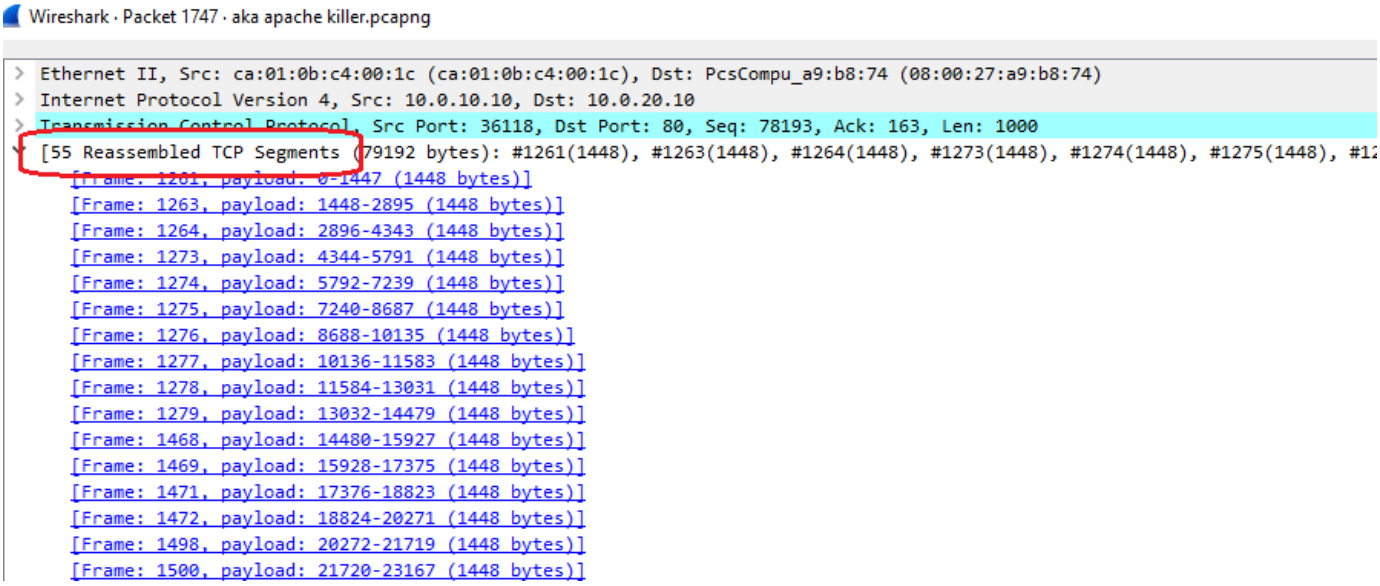
Dal file di output, si vede che dopo una decina di secondi il servizio non è più disponibile in quanto le risposte arrivano oltre la soglia impostata di 5 secondi, inoltre abbiamo 2000 connessioni in attesa, 750 connessioni attive e 250 già chiuse.



Ntopng rileva TCP FIN Scan, questo allarme è notificato quando il flag FIN inviati /ricevuti senza risposta superano la soglia impostata che di default è 256 al minuto. Nello specifico l'attacco Apache Killer non effettua scansioni alle porte del server, ma durante questa simulazione dopo ca 20 secondi il server ha già stabilito 750 connessioni e per ognuna di queste alla fine dell'invio del Range Header viene mandato un pacchetto con flag [FIN, PSH, ACK] a cui il server non risponde mai.
Rilevamento di TCP SYN FLOOD anche in questo caso l'attacco Apache Killer non ha l'intendo di inondare il server di pacchetti SYN ma le richieste HTTP sono molte e per ognuna vi è il Three-way Handshake, Ntopng segnala che per ca 3 secondi i pacchetti contenenti FLAG SYN sono state maggiori di 256 al secondo.
L'alert FLOWS FLOOD segnala che per 3 secondi circa è stata superata la soglia impostata (256/sec) di nuovi flussi client/server, anche in questo è associabile all'alto numero di richieste inviate dall'attante.



Analizzando una delle tante richieste HTTP si vede parte del Range Header della richiesta HTTP.



Sono necessari 55 frame per inviare il Range Header completo da 0-; 10-0; → fino a 10-10000, 79192 bytes. Chiaramente il server è sotto un attacco di tipo DoS Range Header.

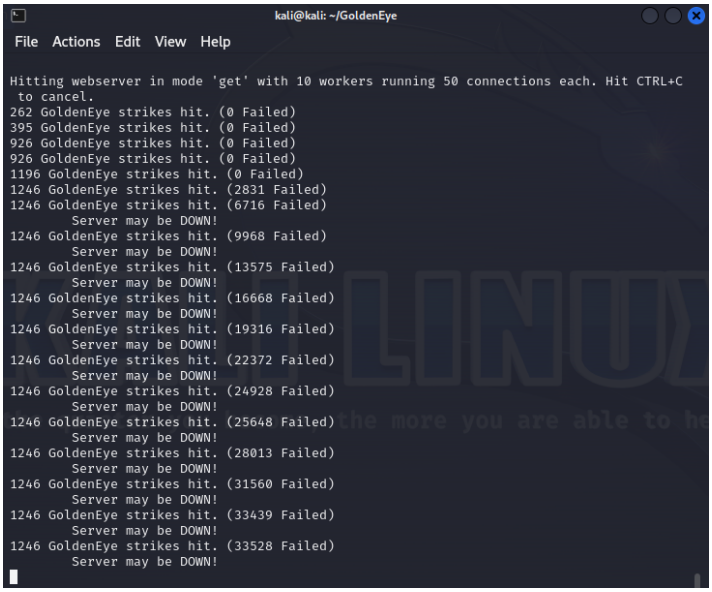
ATTACCHI CON “GOLDENEYE”

Goldeneye è un tool scritto in Python che consente di effettuare attacchi DOS con traffico legittimo in quanto il vettore d’attacco è di tipo HTTP Keep Alive (connessioni persistenti) + NoCache. Questa tecnica invia a intervalli predefiniti molteplici richieste di connessione HTTP con le opzioni Keep Alive e NoCache al fine di rendere indisponibile il server.
Stabilire connessioni HTTP Keep-Alive + NoCache significa che le connessioni tra client e server sono mantenute aperte per consentire l’esecuzione di nuove connessioni HTTP, senza la necessità di rifare il three-way hanshake, migliorando notevolmente le performance. NoCache nel campo “cache control” della richiesta significa che il server non dovrà fornire una risposta dalla cache, ma dovrà verificare la validità della risorsa nel suo sistema di memorizzazione; in altre parole, dovrà fornire la risorsa sempre fresca e non quella eventualmente memorizzata nella cache.
Questo tipo d’attacco non necessita di grandi quantità di banda, perché mira al raggiungimento del limite di connessioni generando così la negazione del servizio.
Grazie alle opzioni addizionali è anche possibile impostare la rotazione degli user-agent (browser web, lettori multimediali) delle richieste **-u**, il metodo con cui verranno inviate **-m**, un debug più approfondito **-d** e la possibilità di saltare il check di validità del certificato ssl **-n**.

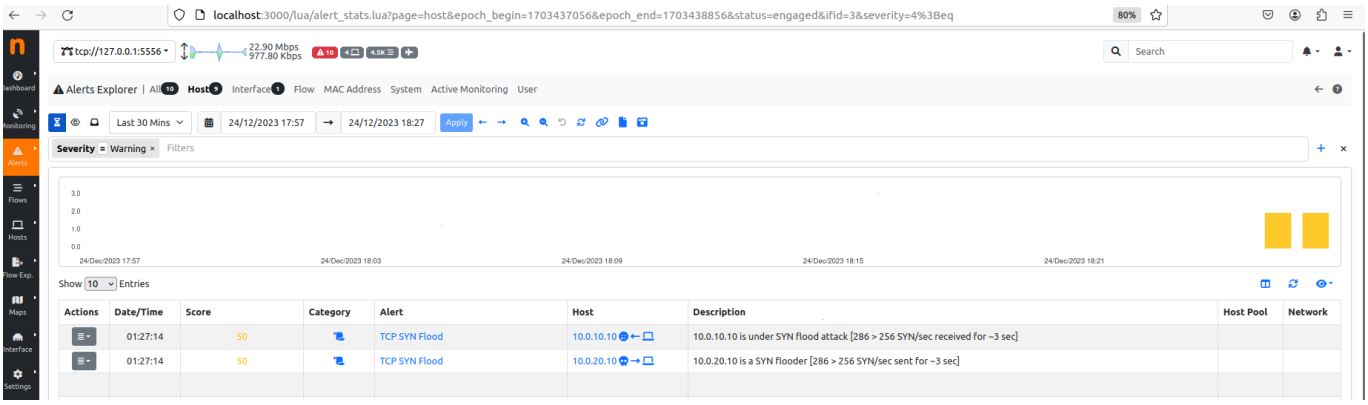
USO: ./goldeneye.py <url> [OPTIONS]

Flag	Description	Default
-u, --useragents	File with user-agents to use (browser/lettori multimediali)	(default: randomly generated)
-w, --workers	Number of concurrent workers, ovvero il numero di thread (processi) che il programma utilizzerà per generare le richieste al server bersaglio durante l'attacco. Ogni worker è responsabile di generare richieste HTTP verso il server.	(default: 10)
-s, --sockets	Number of concurrent sockets (connessioni aperte per ogni worker)	(default: 500)
-m, --method	HTTP Method to use 'get' or 'post' or 'random'	(default: get)
-n, --nosslcheck	Do not verify SSL Certificate	(default: True)
-d, --debug	Enable Debug Mode [more verbose output]	(default: False)
-h, --help	Shows this help	

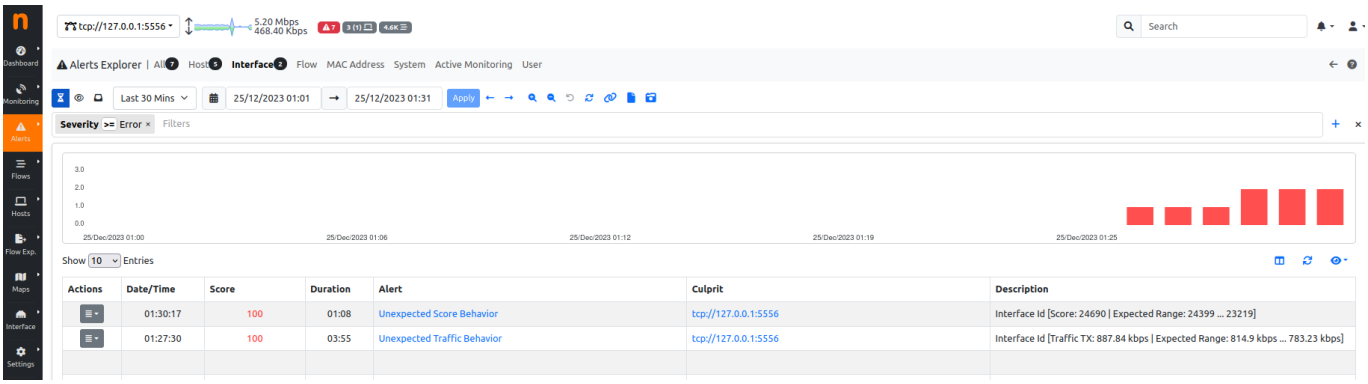
Di seguito un possibile comando d’attacco:
./goldeneye.py http://10.0.20.10 -s 50



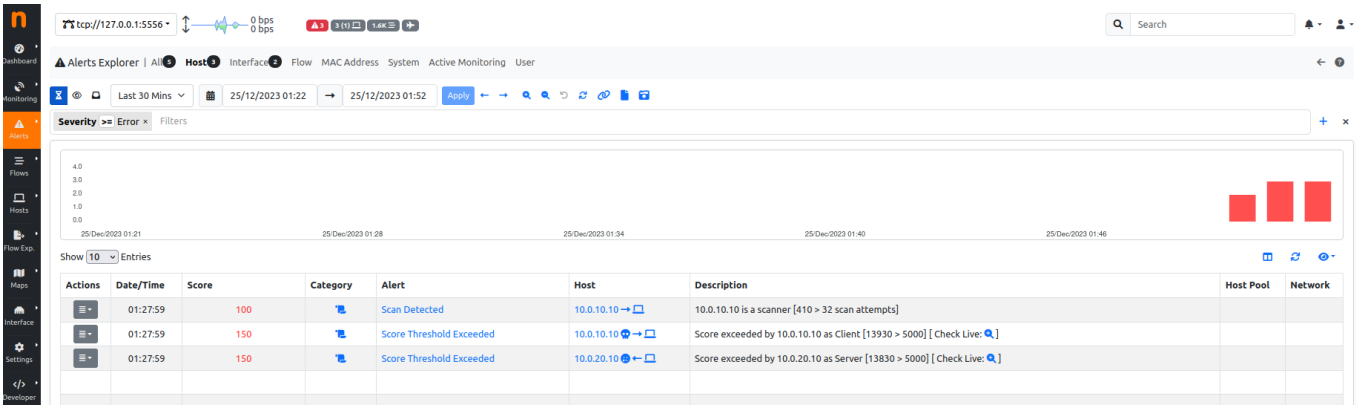
Le seguenti immagini mostrano gli allarmi e i warning riportati da NTOPNG:



Ntopng rileva l'alert TCP SYN FLOOD, anche in questo caso l'attacco inviato da Goldeneye non ha l'intendo di inondare il server di pacchetti SYN; tuttavia, le richieste HTTP sono molte e per ognuna vi è il Three-way Handshake, Ntopng segnala che per ca 3 secondi i pacchetti contenenti FLAG SYN sono state maggiori di 256 al secondo.



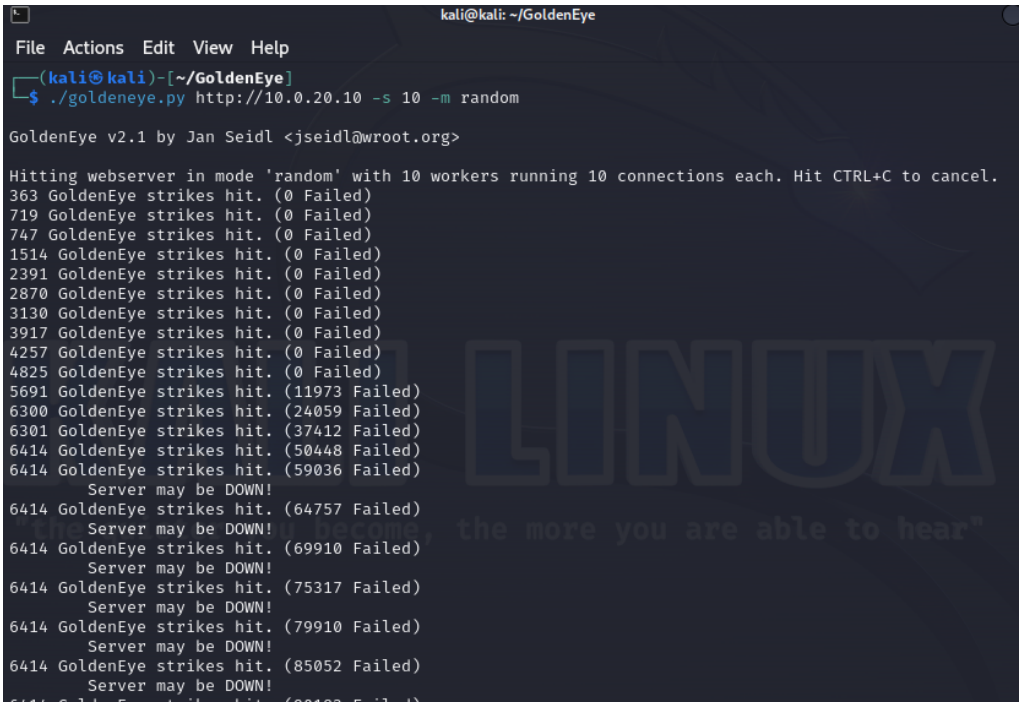
Questi due alert segnalano che c'è un comportamento anomalo dello score e del traffico di rete, questi alert necessito di analisi approfondite al fine di rilevare minacce o difetti.

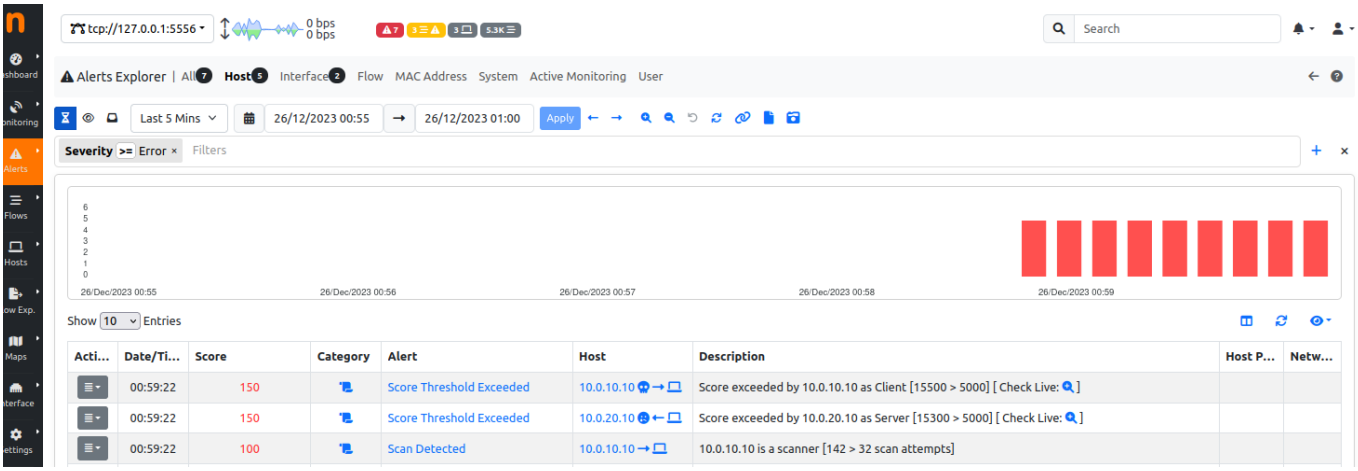


Ntopng rileva uno SCAN DETECTED, nello specifico l'attacco Goldeneye non fa una scansione ma Ntopng segnala che tra l'host 10.0.10.10 e il server 10.0.20.10 si sono verificati più di 32 flussi HTTP incompleti. L'alert SCORE THRESHOLD EXCEEDED ovvero soglia del punteggio superata viene riportato quando la somma del punteggio (flusso/flusso sorgente/flusso destinazione) è maggiore del valore impostato, in questo caso il valore di default che è 5000.

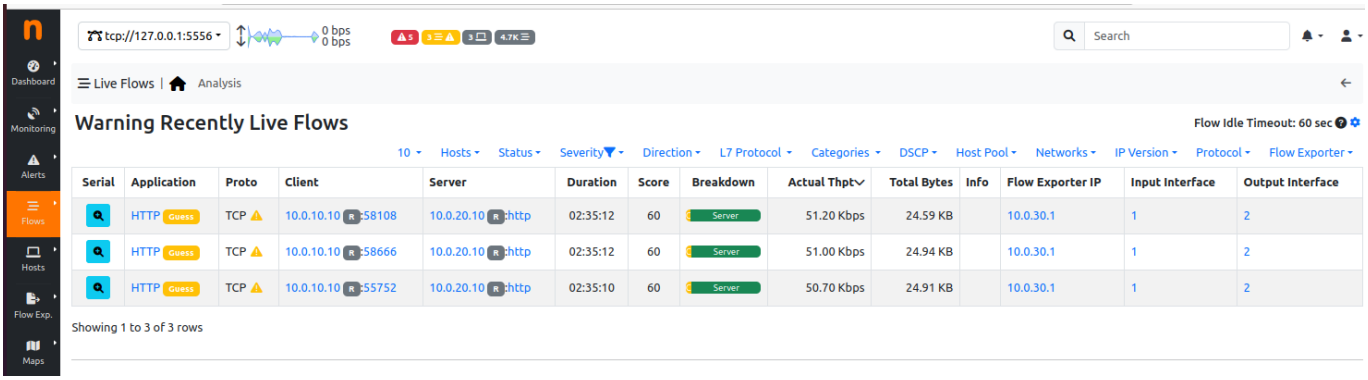
Di seguito un attacco con 10 connessioni per ognuno dei 10 processi di default e con metodo random tra GET e POST

./goldeneye.py http://10.0.20.10 -s 10 -m random





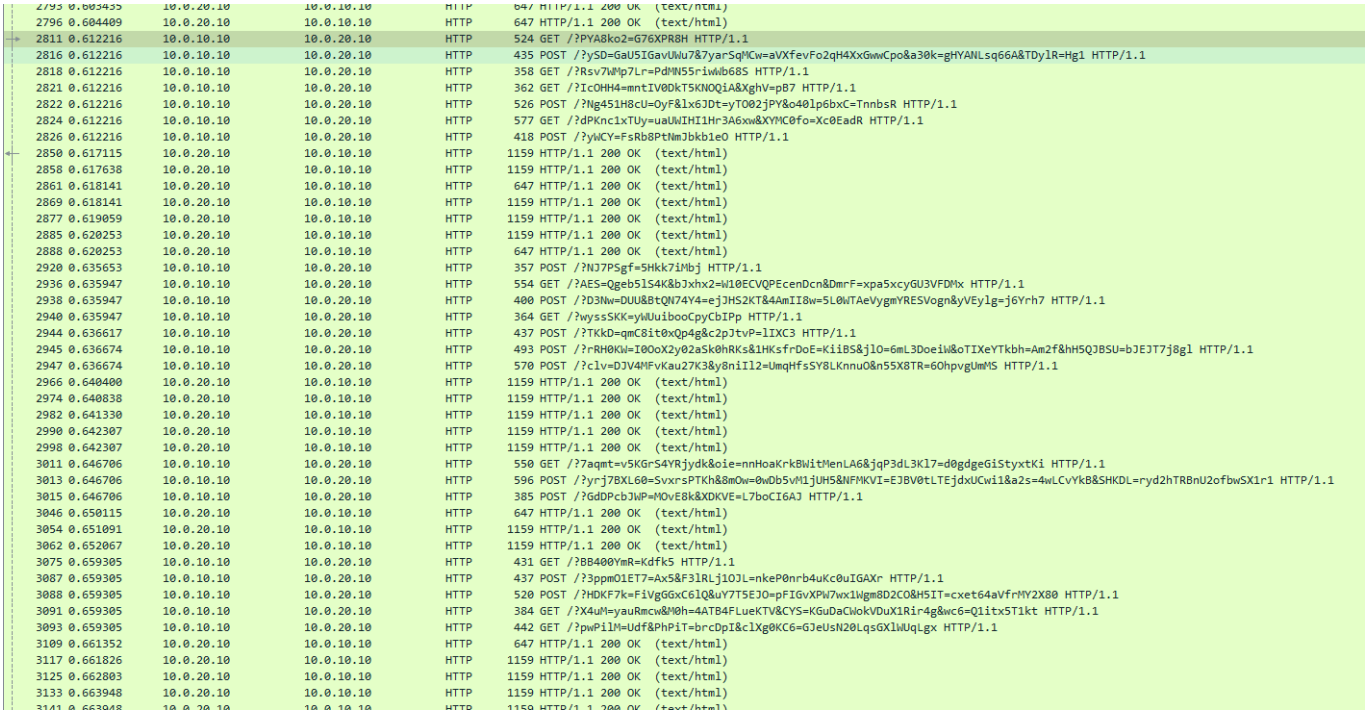
Anche in questo caso sono rilevati gli stessi alert degli altri attacchi.



Ntopng riporta anche dei warning relativi ai flussi http, questo accade quando è in corso un numero significativo di flussi dati attivi nella rete, quindi un alto utilizzo delle risorse e di conseguenza preoccupazioni per la sicurezza. Ovviamente è necessaria un'analisi per capire se i modelli di traffico sono previsti o anomali; nel nostro caso i flussi con IP sorgente uguale sono centinaia, il che ci fa pensare a un attacco di tipo DoS, ma non è possibile

stabilire con Ntopng il tipo di attacco, in realtà non possiamo neanche sapere con sicurezza se i flussi contengono richieste HTTP o solamente pacchetti TCP.

Di seguito una cattura:

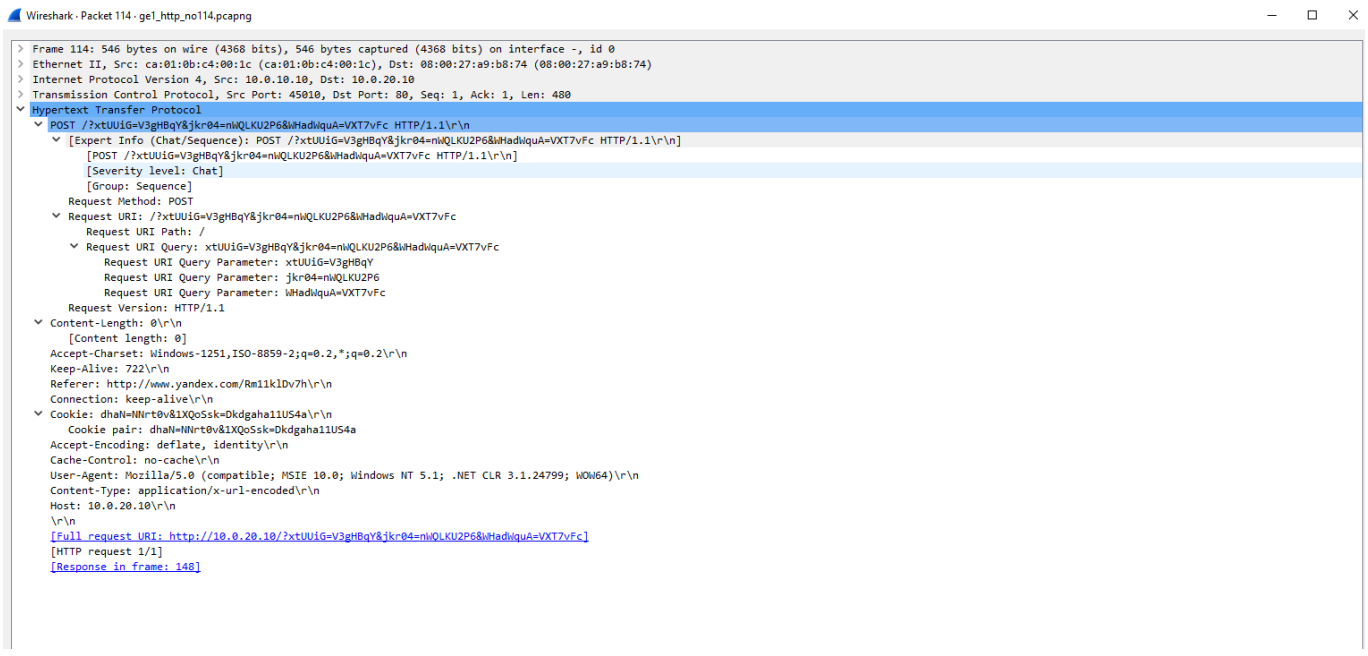


2793	0.603435	10.0.20.10	10.0.10.10	HTTP	647	HTTP/1.1 200 OK (text/html)
2796	0.604409	10.0.20.10	10.0.10.10	HTTP	647	HTTP/1.1 200 OK (text/html)
2811	0.612216	10.0.10.10	10.0.20.10	HTTP	524	GET /?PYA8ko2=G76XPR8H HTTP/1.1
2816	0.612216	10.0.10.10	10.0.20.10	HTTP	435	POST /?ySD=GaU5IGavUWu7&7yarSqMCw=aVXfevFo2qH4XxGwwCpo&a30k=gHYANLsq66A&TDylR=Hg1 HTTP/1.1
2818	0.612216	10.0.10.10	10.0.20.10	HTTP	358	GET /?Rsv7WMp7Lr=PdMN55riwWb68S HTTP/1.1
2821	0.612216	10.0.10.10	10.0.20.10	HTTP	362	GET /?IcOHH4=mntIV0DkT5KNOQiA&XghV=pB7 HTTP/1.1
2822	0.612216	10.0.10.10	10.0.20.10	HTTP	526	POST /?Ng451H8cU=OyF&lx6JDt=yTO02jPY&o40lp6bxC=TnnbsR HTTP/1.1
2824	0.612216	10.0.10.10	10.0.20.10	HTTP	577	GET /?dPKnc1xTUy=uaUWIHI1Hr3A6xw&XYMC0fo=Xc0EadR HTTP/1.1
2826	0.612216	10.0.10.10	10.0.20.10	HTTP	418	POST /?yWCY=FsRb8PtNmJbkb1eO HTTP/1.1
2850	0.617115	10.0.20.10	10.0.10.10	HTTP	1159	HTTP/1.1 200 OK (text/html)
2858	0.617638	10.0.20.10	10.0.10.10	HTTP	1159	HTTP/1.1 200 OK (text/html)
2861	0.618141	10.0.20.10	10.0.10.10	HTTP	647	HTTP/1.1 200 OK (text/html)
2869	0.618141	10.0.20.10	10.0.10.10	HTTP	1159	HTTP/1.1 200 OK (text/html)
2877	0.619059	10.0.20.10	10.0.10.10	HTTP	1159	HTTP/1.1 200 OK (text/html)
2885	0.620253	10.0.20.10	10.0.10.10	HTTP	1159	HTTP/1.1 200 OK (text/html)
2888	0.620253	10.0.20.10	10.0.10.10	HTTP	647	HTTP/1.1 200 OK (text/html)
2920	0.635653	10.0.10.10	10.0.20.10	HTTP	357	POST /?NJ7PSgf=5Hkk7iMbj HTTP/1.1
2936	0.635947	10.0.10.10	10.0.20.10	HTTP	554	GET /?AES=Qgeb5lS4K&bJxhx2=W10ECVQPEcenDcn&DmrF=xpa5xcyGU3VFDMx HTTP/1.1
2938	0.635947	10.0.10.10	10.0.20.10	HTTP	400	POST /?D3Nw=DUU&BtQN74Y4=ejJHS2KT&4AmII8w=5L0WTAeVygmYRESVogn&yVEylg=j6Yrh7 HTTP/1.1
2940	0.635947	10.0.10.10	10.0.20.10	HTTP	364	GET /?wyssSKK=yWUuibooCpyCbIPp HTTP/1.1
2944	0.636617	10.0.10.10	10.0.20.10	HTTP	437	POST /?TkkD=qmC8it0xQp4g&c2pJtvP=lIXC3 HTTP/1.1
2945	0.636674	10.0.10.10	10.0.20.10	HTTP	493	POST /?rRH0kW=I0OoX2y02aSk0hRKs&1HKsfrDoE=KiiBS&jlO=6mL3DoeiW&oTIXeYTkbh=Am2f&hH5QJBSU=bJEJT7j8gl HTTP/1.1
2947	0.636674	10.0.10.10	10.0.20.10	HTTP	570	POST /?clv=DJV4MFvKau27K3&y8niIl2=UmqHfsSY8LKnnuO&n55X8TR=6OhpvgUmMS HTTP/1.1
2966	0.640400	10.0.20.10	10.0.10.10	HTTP	1159	HTTP/1.1 200 OK (text/html)
2974	0.640838	10.0.20.10	10.0.10.10	HTTP	1159	HTTP/1.1 200 OK (text/html)
2982	0.641330	10.0.20.10	10.0.10.10	HTTP	1159	HTTP/1.1 200 OK (text/html)
2990	0.642307	10.0.20.10	10.0.10.10	HTTP	1159	HTTP/1.1 200 OK (text/html)
2998	0.642307	10.0.20.10	10.0.10.10	HTTP	1159	HTTP/1.1 200 OK (text/html)
3011	0.646706	10.0.10.10	10.0.20.10	HTTP	550	GET /?7aqmt=v5KGrS4YRjydk&oie=nnHoaKrkBWitMenLA6&jqP3dL3Kl7=d0gdgeGiStyxtKi HTTP/1.1
3013	0.646706	10.0.10.10	10.0.20.10	HTTP	596	POST /?yrj7BXL60=SvxrsPTKh&8mOw=0wDb5vM1jUH5&NFMKVI=EJBV0tLTEjdxUCwi1&a2s=4wLCvYkB&SHKDL=ryd2hTRBnU2ofbwSX1r1 HTTP/1.1
3015	0.646706	10.0.10.10	10.0.20.10	HTTP	385	POST /?GdDPcbJWP=MOvE8k&XDKVE=L7boCI6AJ HTTP/1.1
3046	0.650115	10.0.20.10	10.0.10.10	HTTP	647	HTTP/1.1 200 OK (text/html)
3054	0.651091	10.0.20.10	10.0.10.10	HTTP	1159	HTTP/1.1 200 OK (text/html)
3062	0.652067	10.0.20.10	10.0.10.10	HTTP	1159	HTTP/1.1 200 OK (text/html)
3075	0.659305	10.0.10.10	10.0.20.10	HTTP	431	GET /?BB400YmR=Kdfk5 HTTP/1.1
3087	0.659305	10.0.10.10	10.0.20.10	HTTP	437	POST /?3ppmO1ET7=Ax5&F3lRLj1OJL=nkeP0nrb4uKc0uIGAXr HTTP/1.1
3088	0.659305	10.0.10.10	10.0.20.10	HTTP	520	POST /?HDKF7k=FiVgGGxC6lQ&uY7T5EJO=pFIGvXPW7wx1Wgm8D2CO&H5IT=cxet64aVfrMY2X80 HTTP/1.1
3091	0.659305	10.0.10.10	10.0.20.10	HTTP	384	GET /?X4uM=yauRmcw&M0h=4ATB4FLueKTV&CYS=KGuDaCWokVDuX1Rir4g&wc6=Q1itx5T1kt HTTP/1.1
3093	0.659305	10.0.10.10	10.0.20.10	HTTP	442	GET /?pwPilM=Udf&PhPiT=brcDpI&clXg0KC6=GJeUsN20LqsGXlWUqLgx HTTP/1.1
3109	0.661352	10.0.20.10	10.0.10.10	HTTP	647	HTTP/1.1 200 OK (text/html)
3117	0.661826	10.0.20.10	10.0.10.10	HTTP	1159	HTTP/1.1 200 OK (text/html)
3125	0.662803	10.0.20.10	10.0.10.10	HTTP	1159	HTTP/1.1 200 OK (text/html)
3133	0.663948	10.0.20.10	10.0.10.10	HTTP	1159	HTTP/1.1 200 OK (text/html)
3141	0.663948	10.0.20.10	10.0.10.10	HTTP	1159	HTTP/1.1 200 OK (text/html)

Una cattura di pochi secondi riporta centinaia di richieste HTTP dallo stesso IP il che fa pensare a un attacco DoS di tipo applicativo.

```
POST /?xtUUiG=V3gHBqY&jkr04=nWQLKU2P6&WHadWquA=VXT7vFc HTTP/1.1
Content-Length: 0
Accept-Charset: Windows-1251,ISO-8859-2;q=0.2,*;q=0.2
Keep-Alive: 722
Referer: http://www.yandex.com/Rm11klDv7h
Connection: keep-alive
Cookie: dhaN=NNrt0v&1XQoSsk=Dkdgaha11US4a
Accept-Encoding: deflate, identity
Cache-Control: no-cache
User-Agent: Mozilla/5.0 (compatible; MSIE 10.0; Windows NT 5.1; .NET CLR 3.1.24799; WOW64)
Content-Type: application/x-url-encoded
Host: 10.0.20.10

HTTP/1.1 200 OK
Date: Mon, 25 Dec 2023 23:58:05 GMT
Server: Apache/2.4.41 (Ubuntu)
Last-Modified: Thu, 30 Nov 2023 21:06:51 GMT
ETag: "2aa6-60b6507da8b01"
Accept-Ranges: bytes
Content-Length: 10918
Vary: Accept-Encoding
Keep-Alive: timeout=5, max=100
Connection: Keep-Alive
Content-Type: text/html


<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
  <!--
    Modified from the Debian original for Ubuntu
    Last updated: 2016-11-16
    See: https://launchpad.net/bugs/1288690
  -->
  <head>
    <meta http-equiv="Content-Type" content="text/html; charset=UTF-8" />
    <title>Apache2 Ubuntu Default Page: It works</title>
    <style type="text/css" media="screen">
  * {
    margin: 0px 0px 0px 0px;
    padding: 0px 0px 0px 0px;
  }

  body, html {
    padding: 3px 3px 3px 3px;

    background-color: #D8DBE2;

    font-family: Verdana, sans-serif;
    font-size: 11pt;
    text-align: center;
```

La risorsa richiesta è strana perché l'uri sembra composto da caratteri casuali. Inoltre, la connessione e di tipo keep/alive NoCache.



Il campo referer è inaspettato perché è come se stesse arrivando da yandex.com (motore di ricerca russo) e il nostro server bersaglio non ha alcuna relazione con yandex.com, inoltre nel valore di Referer di altri pacchetti ci sono motori di ricerca come Bing google ecc.
Nella realtà un IDS potrebbe avere difficoltà nel rilevare questo tipo di attacco, potrebbe essere necessario l'utilizzo WAF Web Application Firewall in modo da filtrare richieste http non veritiere o malformate.

Nella realtà ormai le connessioni HTTP sono tutte sulla porta 443, quindi molte informazioni non sono in chiaro!

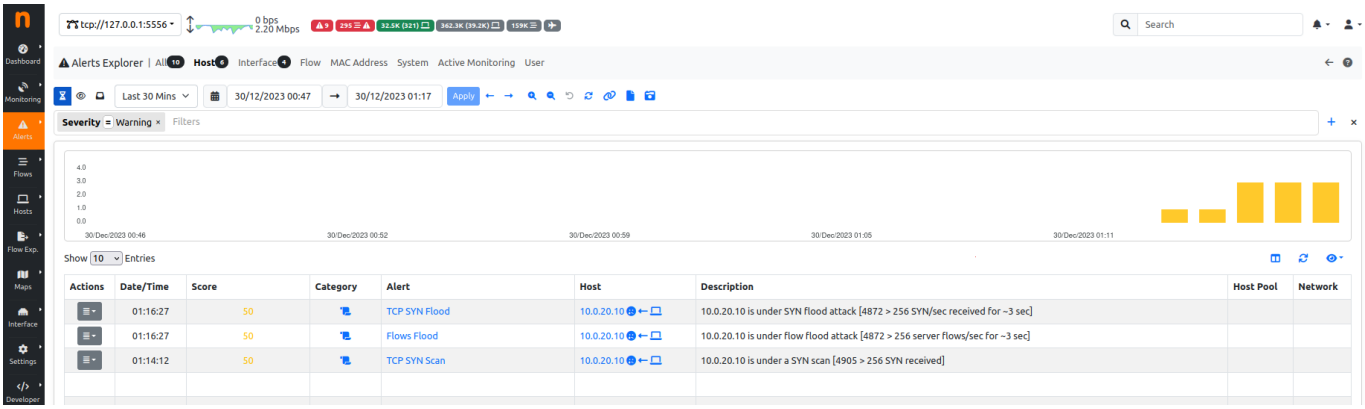
ATTACCHI CON “HPING3”

Il tool hping3 consente di inviare pacchetti manipolati, impostando a piacere dimensioni, quantità e velocità dei pacchetti al fine di sovraccaricare il server bersaglio inondandolo di pacchetti con flag SYN attivo, al fine di aggirare o attaccare i firewall (dovendo bloccare il grande flusso di richieste).
È anche possibile lanciare l’attacco con IP spoofing. Di seguito un attacco SYN FLOOD:

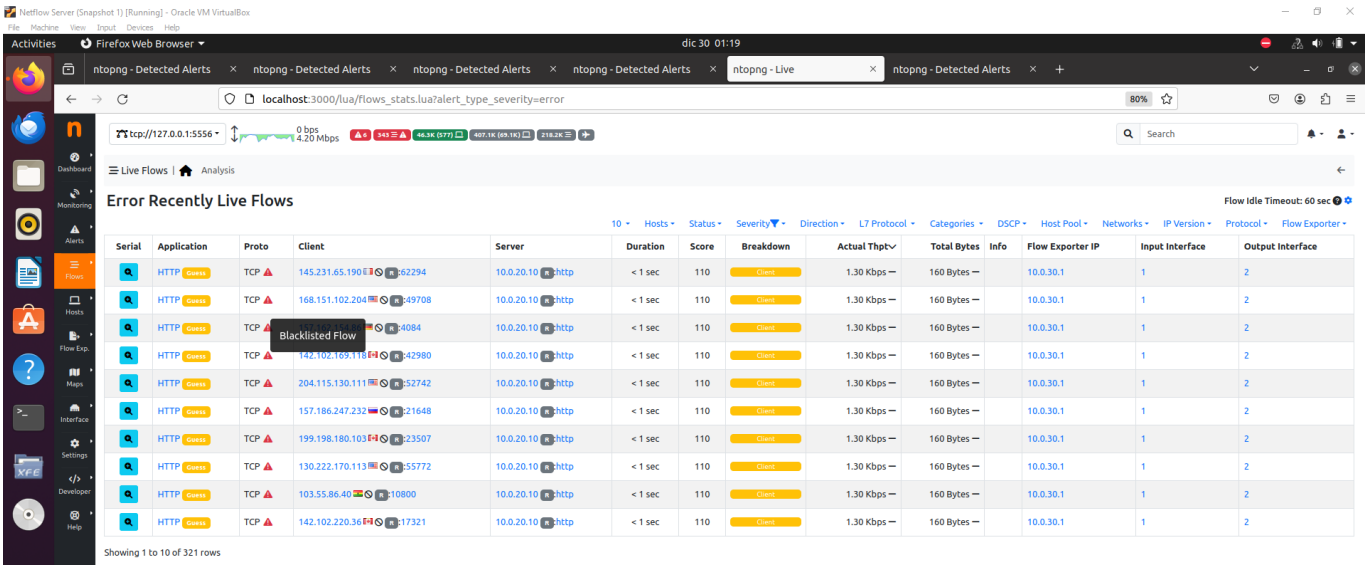
hping3 -c 15000 -d 120 -S -p 80 --flood --rand-source 10.0.20.10

```
┌──(kali㉿kali)-[~]
└─$ sudo hping3 -c 15000 -d 120 -S -p 80 --flood --rand-source 10.0.20.10
[sudo] password for kali:
HPING 10.0.20.10 (eth0 10.0.20.10): S set, 40 headers + 120 data bytes
hping in flood mode, no replies will be shown
^C
--- 10.0.20.10 hping statistic ---
9354535 packets transmitted, 0 packets received, 100% packet loss
round-trip min/avg/max = 0.0/0.0/0.0 ms
```

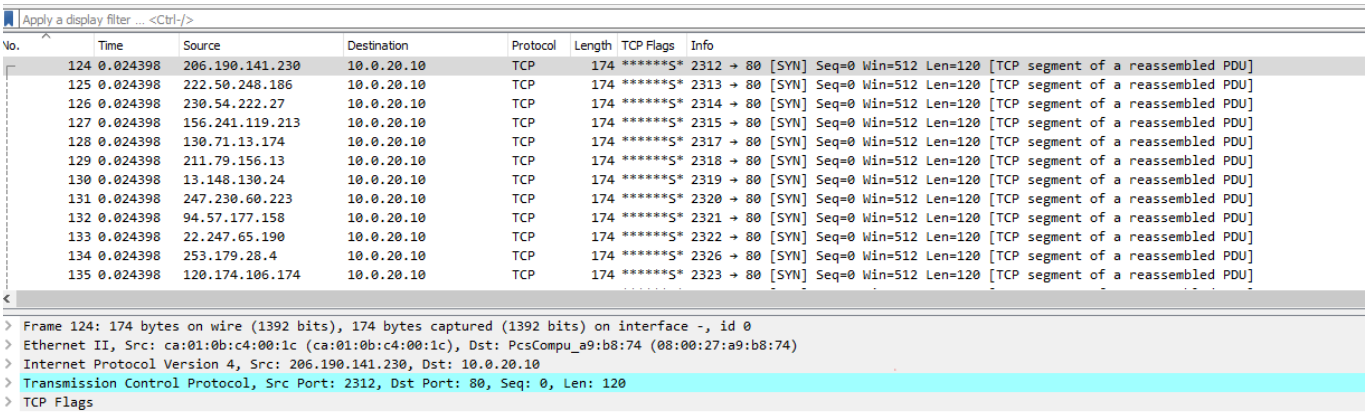
-c invio di 15000 pacchetti
-d di 120 byte di dimensione
-S con impostato flag SYN
-p alla porta 80 (http)
--flood alla massima velocità
--rand-source con IP sorgente random (IP Spoofing)



NTOPNG rileva TCP SYN flood che il tipo di attacco inviato, inoltre l’alert Flows Flood segnala che per 3 secondi circa è stata superata la soglia (256/sec) impostata di nuovi flussi client server, e anche questo è associabile all’alto numero di richieste inviate dall’attacco, TCP SYN Scan perché i pacchetti SYN senza risposta superato la soglia(256/sec) per più di 3 secondi.



Avendo impostato nel comando l'IP sorgente random ogni richiesta ha sorgente casuale, Ntopng notifica che alcuni di questi IP sorgente sono presenti nelle Blacklist al seguente indirizzo https://rules.emergingthreats.net/fwrules/emerging-Block-IPs.txt ". Emerging Threats è un progetto che si occupa di raccogliere e fornire informazioni relative a minacce emergenti e attuali nel panorama della sicurezza informatica. Il progetto fornisce regole di rilevamento delle minacce che possono essere utilizzate con sistemi di prevenzione delle intrusioni (Intrusion Prevention Systems, IPS) e altri strumenti di sicurezza per identificare e mitigare potenziali minacce. Come già detto non è possibile stabilire con Ntopng il tipo di attacco, in realtà non possiamo neanche sapere con sicurezza se i flussi contengono richieste HTTP o solamente pacchetti TCP.



Come è possibile vedere da una cattura wireshark l'attacco ha IP spoofing e ogni pacchetto TCP è di 120 byte (che la lunghezza impostata) e hanno tutti flag SYN attivo, la lunghezza totale del Frame è 174 byte per via dei 20 byte di Header del segmento TCP, 20 byte di Header del pacchetto IP e 14 byte di Header del frame ethernet. L'attacco è palesemente un SYN FLOOD.

MIGLIORAMENTI E SVILUPPI FUTURI

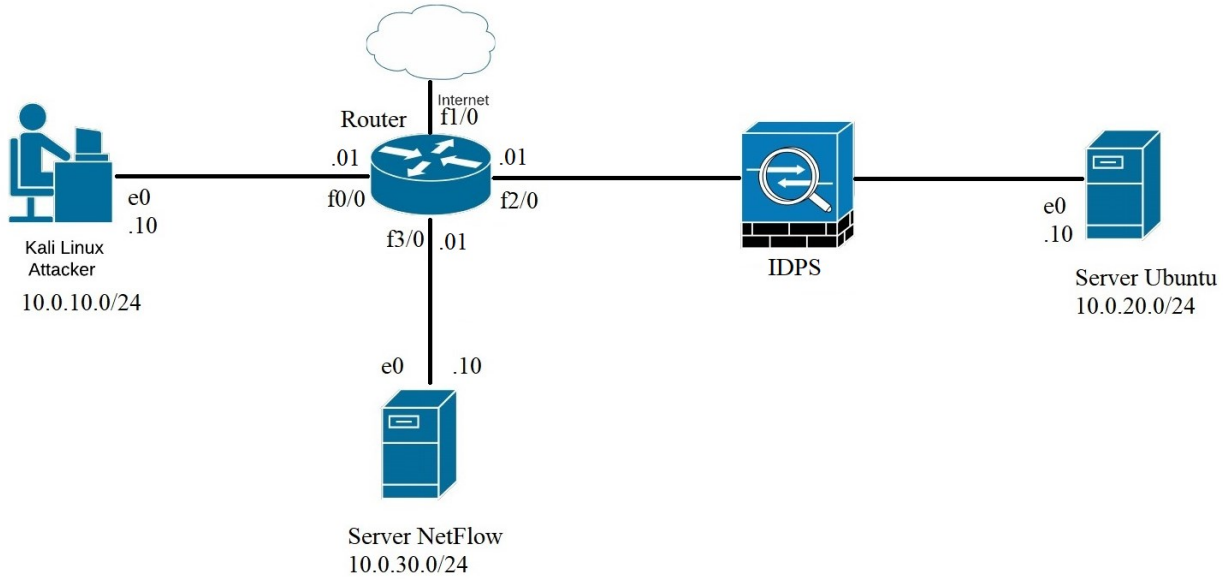
APPLICAZIONE DI TECNICHE DI MACHINE LEARNING

Utilizzando tecniche di machine learning è possibile elaborare i dati Netflow per limitare i danni degli attacchi Denial of Service (DoS). Un ipotetico sistema potrebbe consistere nei seguenti punti:

- Utilizzo dei dati NetFlow come indirizzo e porte sorgente/destinazione, protocolli e volume dei dati.
- Estrazione delle caratteristiche utili allo scopo come, ad esempio, la frequenza delle richieste, dimensione dei pacchetti, durata delle connessioni ecc.
- Definizione dei comportamenti anomali, ad esempio al superamento di un certo numero di connessioni, consumo di banda, tempo di connessione ecc.
- Scelta del modello per la machine learning utilizzando algoritmi di apprendimento automatico:
    - *Alberi decisionali:* un tipo spesso utilizzato per il rilevamento di attacchi informatici è l'Isolation Forest, questo algoritmo consiste in un metodo di apprendimento assimilabile, che è in grado di isolare le anomalie selezionando casualmente una caratteristica e selezionando poi casualmente un valore suddiviso tra i valori massimi e minimi della caratteristica. l'Isolation Forest è in grado di trovare dati insoliti velocemente.
    - *One-Class Support Vector Machine (SVM):* è un algoritmo di classificazione addestrato per il rilevamento di istanze "normali", tutto ciò che non rientra nel rilevamento e considerato anomalo.
- Scelta della tecnica di training del machine learning, utilizzando un set di dati netflow con caratteristiche di traffico normale. Le anomalie di comportamento potrebbero indicare potenziali attacchi DoS.

IMPLEMENTAZIONE DI UN IDPS A INTEGRAZIONE DI NTOPNG

Ntopng è un'applicazione di monitoraggio del traffico di rete versatile, capace di fornire una visione approfondita delle attività di rete, delle prestazioni e dei possibili problemi di sicurezza e della minacce, tuttavia non è progettato per essere principalmente un sistema di rilevamento delle intrusioni e quindi non ha la stessa profondità di analisi dei modelli di attacco di un IDS dedicato; l'integrazione di un IDPS (Intrusion Detection Prevention System) con ntopng potrebbe fornire una visione più approfondita del traffico di rete e migliorare la capacità di rilevare e rispondere alle minacce di sicurezza. Si preferisce utilizzare un IDPS perché l'IDS è uno strumento passivo, la sua funzione è quella di monitorare il traffico di rete e le attività del sistema alla ricerca di comportamenti anomali o firme di attacchi noti senza agire attivamente sul flusso di dati; un IDPS svolge i controlli passivi di un IDS ma anche quelli attivi di un IPS, in particolare può intraprendere azioni per impedire il successo di un attacco; questo può comportare ad esempio la modifica delle regole del firewall o il blocco di indirizzi IP sospetti o altre misure di prevenzione.
Per poter agire attivamente sul traffico in maniera che sia possibile bloccarlo, modificarlo e filtrarlo in base alle policy di sicurezza impostate è necessario collegare l'IDPS in linea, questo comporta il passaggio fisico del traffico di rete attraverso l'IDPS.
Il collegamento in linea di un IDPS:



Bisogna considerare che il passaggio del flusso dei dati traffico dati attraverso l'IDPS può introdurre una certa latenza, inoltre un'azione di blocco basata su un evento che in realtà è un falso positivo può interrompere il traffico legittimo.

Due possibili software IDPS sono Suricata e Snort; l'integrazione tra ntopng e Suricata può essere stabilita attraverso opportune configurazioni tramite l'acquisizione di flussi e avvisi da Suricata utilizzando il formato Eve JSON tramite protocollo syslog.

CONCLUSIONI

Il presente studio mi ha permesso di approfondire la conoscenza degli attacchi di tipo Denial of Service (DoS) attraverso l'utilizzo del protocollo NetFlow, sulla sua efficacia nell'identificare tali minacce.
Durante l'analisi, è emerso che NetFlow, con la sua capacità di catturare e analizzare i flussi di traffico di rete, ha fornito una base solida per la rilevazione precoce degli attacchi.
L'integrazione di NetFlow con ntopng si è dimostrata particolarmente fruttuosa, consentendo la generazione di allarmi e avvisi tempestivi in risposta a anomalie nel traffico. Tuttavia, per una comprensione più approfondita degli attacchi e per l'analisi specifica delle loro caratteristiche, l'utilizzo di Wireshark si è rivelato indispensabile. Wireshark ha offerto una visione più dettagliata e granulare del traffico, consentendo una valutazione più approfondita degli schemi e delle tecniche utilizzate dagli aggressori.

Guardando al futuro, emerge la necessità di una difesa attiva contro gli attacchi DoS. Al fine di prevenire e mitigare efficacemente queste minacce, si raccomanda l'integrazione di un Intrusion Detection and Prevention System (IDPS). Un IDPS aggiunge uno strato di sicurezza attiva alla rete, intervenendo prontamente per bloccare attività sospette e mitigare gli impatti degli attacchi in tempo reale.

In conclusione, questa ricerca ha evidenziato l'importanza di una strategia di difesa completa e integrata per affrontare gli attacchi DoS. NetFlow, ntopng e Wireshark si sono dimostrati strumenti complementari, fornendo una panoramica ampia e dettagliata degli attacchi. L'integrazione di un IDPS completa questa strategia, consentendo una risposta tempestiva e efficace per proteggere l'integrità e la disponibilità delle reti

BIBLIOGRAFIA

Citazioni e riferimenti utilizzati nella tesi:

- Documentazione relativa a Ntopng (Download, licenze, alert, configurazioni ecc.)
  - *https://www.ntop.org*
  - https://www.ntop.org/guides/ntopng/advanced_features/suricata.html
  - *https://www.ntop.org/ntopng/what-is-score-and-how-it-can-drive-you-towards-network-issues/*
  - *https://www.ntop.org/guides/ntopng/alerts/interface_checks.html*

- Ntopng community Italy on Telegram *https://web.telegram.org/k/#@ntop_community_italy*
- Configurazione NetFlow exporter su Cisco router
  - *https://www.cisco.com/c/en/us/td/docs/ios-xml/ios/fnetflow/configuration/xe-16/fnf-xe-16-book/fnf-v9-export.html*
  - *https://documentation.solarwinds.com/en/success_center/nta/content/nta-setting-up-network-devices-to-export-netflow-data-manually-sw75.htm*

- Immagine Router Cisco 7200 *https://cios.dhitechnical.com/7200/*
- Sito GNS3 per download software e IOS Cisco *https://www.gns3.com*
- Community GNS3 per realizzazione architettura di rete *https://www.gns3.com/community*
- VirtualBox software download *https://www.virtualbox.org/wiki/Downloads*
- Risoluzione problemi VirtualBox *https://forums.virtualbox.org/index.php*
- Immagine Kali Linux *https://www.kali.org/get-kali/#kali-platforms*
- *Immagine Ubuntu* *https://www.ubuntu-it.org/download*
- Utilizzo strumenti Kali Linux *https://www.kali.org/tools/*
  - Goldeneye
    - *https://www.kali.org/tools/goldeneye/*
    - https://www.geeksforgeeks.org/goldeneye-ddos-tool-in-kali-linux/
  - SlowHttpTest
    - *https://github.com/shekyan/slowhttptest*
    - *https://medium.com/@4ag2/slowhttptest-simulate-a-dos-attack-69a0d854dba*
    - https://giovannilubrano.blogspot.com/2015/06/kali-linux-attacco-dos-tramite.html
  - Hping3 *https://www.kali.org/tools/hping3/*

- Wireshark *https://www.wireshark.org/*
- Migliori Strumenti DDoS Per Kali Linux https://www.linkedin.com/pulse/best-ddos-tools-kali-linux-ayomide-oluwaga

APPENDICE

CONFIGURAZIONE ROUTER CISCO

```
flow record Netflow-Record
match ipv4 tos
match ipv4 protocol
match ipv4 source address
match ipv4 destination address
match transport source-port
match transport destination-port
match transport icmp ipv4 type
match transport icmp ipv4 code
match transport tcp source-port
match transport tcp destination-port
match transport udp source-port
match transport udp destination-port
match interface input
collect transport tcp sequence-number
collect transport tcp acknowledgement-number
collect transport tcp header-length
collect transport tcp window-size
collect transport tcp urgent-pointer
collect transport tcp flags
collect transport udp message-length
collect interface output
collect counter bytes
collect counter packets
collect application name
!
!
flow exporter Netflow-Server
destination 10.0.30.10
source FastEthernet2/0
transport udp 2055
template data timeout 15
option application-table timeout 60
option application-attributes timeout 300
!
!
flow monitor Netflow-Monitor
exporter Netflow-Server
cache timeout active 15
record Netflow-Record
!
```

Definiamo le informazioni che ciascun record NetFlow deve descrivere

Definiamo il NetFlow server (ntopng) verso la quale esportare i record

Correliamo il Netflow-Record ed il Netflow-Server all'interno di un profilo definito Neflow-Monitor, tale profilo andrà poi legato alle interfacce che dalla quale passa il traffico da

```
!
interface FastEthernet0/0
 description To-Client
 ip address 10.0.10.1 255.255.255.0
 ip nat inside
 ip flow monitor Netflow-Monitor input
 ip flow monitor Netflow-Monitor output
 duplex full
!
interface FastEthernet1/0
 description To-Server
 ip address 10.0.20.1 255.255.255.0
 ip nat inside
 ip flow monitor Netflow-Monitor input
 ip flow monitor Netflow-Monitor output
 duplex full
!
interface FastEthernet2/0
 description To-NetFlow
 ip address 10.0.30.1 255.255.255.0
 ip nat inside
 duplex full
!
interface FastEthernet3/0
 description To-Internet
 ip address dhcp
 ip nat outside
 duplex full
!
!
ip nat inside source list NAT-CLIENT interface FastEthernet3/0 overload
!
!
!
ip route 0.0.0.0 0.0.0.0 FastEthernet3/0 dhcp
!
!
```

Configurazione delle interfacce che fungono da default gateway rispettivamente per le reti di

- ***Client (FastEthernet0/0)***
- ***Server (FastEthernet1/0)***
- ***NetFlow Server (FastEthernet2/0)***
- ***Internet (FastEthernet3/0)****

Le prime due interfacce hanno il profilo Netflow-Monitor definito sopra per collezionare il traffico in input ed output, generare dei record NetFlow ed inviarli al NetFlow Server

****L'interfaccia internet non ha IP statico in quanto riceve la configurazione IP dalla NAT Cloud che funge anche da DHCP Server***

Tramite i comandi "ip nat inside" definiamo le reti che dovranno essere "nattate" con l'indirizzo IP dell'interfaccia FastEthernet3/0

Configurazione NAT in modalità overload, utile alla navigazione Internet.

Default-route statica che utilizza il next-hop fornito dal server DHCP (NAT Cloud)

CONFIGURAZIONE NTOPNG

Il file di configurazione è il seguente "/etc/ntopng/ntopng.conf", mediante la seguente configurazione specifichiamo l'indirizzo e la porta del socket su cui ntopng dovrebbe ascoltare per ricevere i dati. Questi dati sono generati da nProbe, un componente associato a ntopng che può essere utilizzato per monitorare il traffico di rete e inviare i dati a ntopng per l'analisi e la visualizzazione.

```
-i=tcp://127.0.0.1:5556
```

CONFIGURAZIONE NPROBE

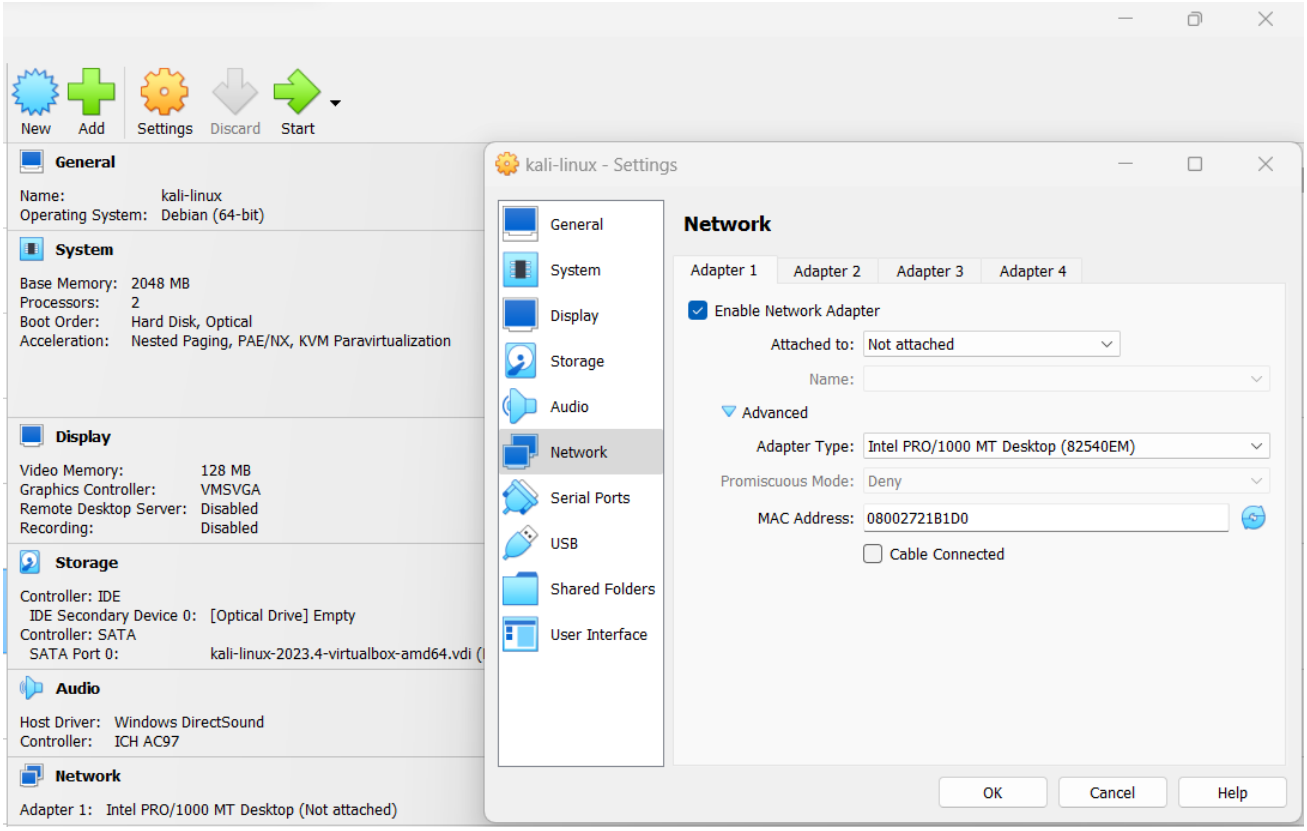
Il file di configurazione è il seguente "/etc/nprobe/nprobe.conf", mediante le seguenti righe di configurazione impostiamo l'endpoint ZeroMQ su cui nProbe invierà i dati e la porta su cui nProbe si metterà in ascolto per ricevere flussi NetFlow (la porta standard per l'invio dei dati NetFlow è la udp2055).

**ZeroMQ è una libreria per la messaggistica asincrona e può essere utilizzata per la comunicazione tra processi*

```
--zmq="tcp://*:5556"

--collector-port=2055
```

CONFIGURAZIONE VM VIRTUALBOX

Per far sì che GNS3 possa gestire la scheda di rete di ciascuna virtual machine, è necessario configurare i settaggi di rete di ciascuna VM come indicato nell'immagine di seguito.
Nel momento in cui avviamo una VM VirtualBox attraverso GNS3, quest'ultimo si occuperà di applicare la configurazione adeguata



Caratteristiche VM Kali Linux