



**UNIVERSITATEA DE VEST DIN TIMIȘOARA
FACULTATEA DE MATEMATICĂ ȘI INFORMATICĂ
PROGRAMUL DE STUDII DE LICENȚĂ: INFORMATICĂ**

LUCRARE DE LICENȚĂ

COORDONATOR:
Asistent Dr. Florin Roșu

ABSOLVENT:
Stoie Vlad-Florin

**TIMIȘOARA
2025**

UNIVERSITATEA DE VEST DIN TIMIȘOARA
FACULTATEA DE MATEMATICĂ ȘI INFORMATICĂ
PROGRAMUL DE STUDII DE LICENȚĂ: INFORMATICĂ

Rețele Informatice: Configurare, Vulnerabilități și Securizare împotriva Atacurilor Ciberneticе

COORDONATOR:
Asistent Dr. Florin Roșu

ABSOLVENT:
Stoie Vlad-Florin

TIMIȘOARA
2025

Cuprins

Abrevieri

Rezumat

1	Introducere	1
2	Descrierea problemei	3
2.1	Riscuri de securitate în rețelele informatice	3
2.2	Probleme de performanță și scalabilitate	3
2.3	Lipsa autentificării și a controlului identității în rețea	4
2.4	Riscurile asociate firmware-ului învechit și lipsa actualizărilor	4
2.5	Necesitatea unei abordări proactive	5
3	Abordări existente	6
4	Fundamente teoretice	8
4.1	Echipamente de rețea	8
4.1.1	Routeri Cisco	8
4.1.2	Switch-uri Cisco (Layer 2 și Layer 3)	9
4.2	Protocoale de rutare și segmentare a rețelei	9
4.2.1	Protocolul OSPF	9
4.2.2	Protocolul EIGRP	9
4.2.3	Comparație între OSPF și EIGRP	10
4.2.4	Segmentarea rețelei prin VLAN	10
4.2.5	Prevenirea buclelor cu STP	10
4.3	Tehnologii de securitate și control al accesului	10
4.3.1	Listele de control al accesului (ACL)	10
4.4	Sisteme de operare utilizate în infrastructura simulată	11
4.4.1	Stații de lucru Windows 10	11
4.4.2	Stații de lucru Ubuntu	11
4.5	Platforme și medii de simulare	12
4.5.1	Cisco Packet Tracer	12
4.5.2	GNS3 (Graphical Network Simulator 3)	12
4.5.3	VMware Workstation	13
4.5.4	Wireshark	13
4.6	Tipuri de atacuri cibernetice simulate	14
4.6.1	ARP Spoofing	14
4.6.2	DHCP Starvation	14

4.6.3	ICMP Flood	14
4.6.4	Unelte Kali Linux utilizate	14
5	Metodologia de realizare	16
5.1	Zona CORE	16
5.1.1	Routerul central – CORE-ROUTER	17
5.1.2	Switch-ul central – CORE-SW	18
5.1.3	Switch-ul de distribuție – SW-1-15	19
5.1.4	Switch-ul de distribuție – SW-16-31	20
5.2	Noduri de tip Cloud și configurarea tunelurilor UDP	21
5.3	Zona ISP și conectarea la rețeaua externă	23
5.3.1	Conectivitate cu Internetul	23
5.3.2	Conectarea cu infrastructura internă (CORE)	23
5.3.3	Configurația completă a routerului ISP	23
5.3.4	Limitări și observații	24
5.4	Structura sălilor individuale	24
5.5	Simularea atacurilor cibernetice	25
5.5.1	Scanarea rețelei folosind <code>nmap</code>	25
5.5.2	Simularea atacului ARP Spoofing	28
5.5.3	Simularea atacului DHCP Starvation	29
5.5.4	Simularea atacului ICMP Flood	30
5.6	Măsuri de securitate	30
5.6.1	Măsuri de protecție împotriva atacului ARP Spoofing	30
5.6.2	Măsuri de protecție împotriva atacului DHCP Starvation	31
5.6.3	Măsuri de protecție împotriva atacului ICMP Flood	32
6	Rezultate și discuții	33
7	Concluzii și direcții viitoare	35

Bibliografie

Anexe

Abrevieri

ACL *Access Control List*

BPDU *Bridge Protocol Data Unit*

CLI *Command Line Interface*

CVE *Common Vulnerabilities and Exposures*

DDoS *Distributed Denial of Service*

DHCP *Dynamic Host Configuration Protocol*

DMZ *Demilitarized Zone*

DNS *Domain Name System*

EAP *Extensible Authentication Protocol*

EIGRP *Enhanced Interior Gateway Routing Protocol*

FTP *File Transfer Protocol*

GUI *Graphical User Interface*

hping3 *High-performance Packet Generator (instrument de testare pentru generare de pachete personalizate)*

HTTP *Hypertext Transfer Protocol*

HTTPS *Hypertext Transfer Protocol Secure*

ICMP *Internet Control Message Protocol*

IDS *Intrusion Detection System*

IEEE *Institute of Electrical and Electronics Engineers*

IETF *Internet Engineering Task Force*

IOS *Internetwork Operating System*

IOSv *IOS Virtual (versiune virtualizată a sistemului de operare Cisco IOS)*

IP *Internet Protocol*

IPS *Intrusion Prevention System*

IPsec *Internet Protocol Security*

IPv4 *Internet Protocol version 4*

IPv6 *Internet Protocol version 6*

IPX *Internetwork Packet Exchange*

LAN *Local Area Network*

Linux *Linux (sistem de operare open-source)*

LSA *Link-State Advertisement*

LTS *Long Term Support*

MAC *Media Access Control*

MitM *Man-in-the-Middle*

NAT *Network Address Translation*

OSI *Open Systems Interconnection*

OSPF *Open Shortest Path First*

PC *Personal Computer*

QoS *Quality of Service*

RADIUS *Remote Authentication Dial-In User Service*

SNMP *Simple Network Management Protocol*

STP *Spanning Tree Protocol*

SVI *Switched Virtual Interface*

Telnet *Telecommunication Network Protocol*

TLS *Transport Layer Security*

UDP *User Datagram Protocol*

VLAN *Virtual Local Area Network*

VoIP *Voice over IP*

VPN *Virtual Private Network*

Rezumat

În această lucrare de licență îmi propun să proiectez și să simulez o rețea de calculatoare utilizând platformele GNS3 și Cisco Packet Tracer. Voi începe prin configurarea detaliată a protocoalelor de rețea și construirea unei infrastructuri funcționale, urmând ca ulterior să simulez atacuri cibernetice asupra acesteia pentru a identifica posibile vulnerabilități și riscuri de securitate. Pe baza acestor simulări, îmi propun să analizez în detaliu impactul acestor atacuri asupra funcționării rețelei și să propun soluții de securizare eficiente, menite să prevină astfel de amenințări. Lucrarea va îmbina atât aspectele teoretice, cât și cele practice ale procesului de configurare și securizare a rețelelor, oferind o descriere detaliată a întregului proces, de la implementare până la testare și securizare. În acest fel, îmi propun să ofer o perspectivă valoroasă asupra securității rețelelor, cu aplicabilitate practică și relevanță academică.

1 Introducere

În contextul actual al digitalizării accelerate, securitatea rețelelor informatice reprezintă un subiect de interes major atât pentru organizații, cât și pentru cercetători. Dezvoltarea rapidă a infrastructurilor de rețea, alături de creșterea exponențială a numărului de dispozitive conectate la internet, a condus la o expunere semnificativă la riscurile asociate atacurilor cibernetice. Aceste atacuri, precum sniffing, spoofing, ransomware sau atacurile de tip *DDoS*, amenință confidențialitatea integritatea și disponibilitatea datelor, având consecințe grave asupra funcționării organizațiilor și asupra utilizatorilor individuali. În acest caz, apariția unor soluții robuste și eficiente pentru protecția rețelelor informatice devine o necesitate, mai ales în condițiile în care se cere un echilibru constant între performanța rețelei și securitatea acesteia. Lucrarea propune o metodă pentru securizarea rețelelor informatice, având ca obiectiv proiectarea, simularea și analiza unei rețele virtuale care reflectă scenarii reale întâlnite în organizații. Infrastructura proiectată va permite identificarea vulnerabilităților și testarea unor soluții eficiente pentru protecție. Procesul include configurarea unei rețele funcționale, care să suporte operațiuni obișnuite și simularea unor atacuri cibernetice precum accesul neautorizat sau interceptarea traficului. Pe baza rezultatelor obținute, vor fi implementate măsuri care să protejeze resursele critice și să minimizeze riscurile asociate. Pentru atingerea acestor obiective, vor fi utilizate aplicațiile GNS3 și Cisco Packet Tracer. GNS3 va facilita simularea unor scenarii care includ mașini virtuale, iar Cisco Packet Tracer va fi utilizat pentru o vizualizare mai bună a structurii rețelei și verificarea funcționalităților de bază. Această soluție pune accent pe identificarea vulnerabilităților în infrastructura rețelelor informatice și pe aplicarea unor măsuri concrete de securizare care să sporească protecția împotriva amenințărilor cibernetice. Contribuția personală a constat în proiectarea unei rețele informatice virtuale, inspirată din structura reală a sălilor de la parterul Universității de Vest din Timișoara, simularea unor atacuri cibernetice și formularea de măsuri de securizare împotriva acestor atacuri. Activitatea desfășurată a acoperit întregul proces, de la implementarea infrastructurii până la testare și analiză, cu scopul de a evidenția vulnerabilitățile rețelelor informatice și metodele eficiente de protecție. Lucrarea este organizată în mai multe capitole, fiecare abordând un aspect specific al temei propuse. Primul capitol introduce contextul general al problemicii, prezentând motivația alegerii temei, obiectivele stabilite și soluția propusă. Acesta oferă o bază teoretică pentru înțelegerea provocărilor legate de securitatea rețelelor informatice. Capitolul al doilea este dedicat descrierii problemei. Aici sunt detaliate vulnerabilitățile comune ale rețelelor informatice și amenințările cibernetice care le pot afecta. Totodată, este prezentată importanța identificării acestor probleme pentru a dezvolta soluții eficiente de protecție. Capitolul al treilea tratează abordările existente, explorând soluții similare din literatură de specialitate. Acest capitol include exemple de implementări relevante care au fost uti-

lizate ca punct de plecare pentru proiectarea rețelei simulate în această lucrare. Cel de-al patrulea capitol include detaliile teoretice despre toate tehnologiile, protocoalele, aparatura și software-urile utilizate în realizarea întregii părți aplicative a lucrării. Astfel sprijinim mai buna înțelegere a tuturor conceptelor legate de întregul proces de realizare a rețelei, a atacurilor cibernetice și a măsurilor de securitate propuse. În capitolul al cincilea este prezentată metodologia de realizare a proiectului propus, care constituie componenta principală a contribuțiilor personale din această lucrare. Sunt descrise etapele parcurse pentru proiectarea, configurarea și testarea infrastructurii de rețea. Vom detalia arhitectura generală a rețelei, organizată logic pe mai multe VLAN-uri interconectate printr-un nucleu central (CORE), precum și modul în care au fost configurate echipamentele de rețea folosind platformele Cisco Packet Tracer și GNS3. De asemenea, vor fi prezentate simulări ale unor atacuri informatice relevante, precum și soluțiile de protecție aplicate, cu scopul de a valida funcționalitatea și securitatea rețelei. Capitolul al șaselea include analiza rezultatelor obținute. Sunt discutate performanța rețelei în urma simulărilor, impactul atacurilor simulate și eficiența măsurilor de securizare implementate. Ultimul capitol conține concluziile și direcțiile viitoare. Aici sunt sintetizate rezultatele lucrării, subliniind contribuția acesteia la domeniul securității rețelelor informatice. De asemenea, sunt propuse direcții pentru cercetări ulterioare, care ar putea extinde și aprofunda soluțiile dezvoltate în această lucrare.

2 Descrierea problemei

În era digitalizării accelerate, rețelele informatice reprezintă infrastructura esențială care susține funcționarea instituțiilor, companiilor și a întregii societăți moderne. Complexitatea din ce în ce mai mare a acestora aduce însă și un set considerabil de provocări, în special din punct de vedere al securității, performanței și scalabilității. În acest context, identificarea riscurilor și a limitărilor cu care se confruntă rețelele actuale este o etapă esențială pentru construirea unui mediu de comunicații sigur și eficient.

2.1 Riscuri de securitate în rețelele informatice

Printre cele mai importante probleme cu care se confruntă administratorii de rețea se află riscurile asociate atacurilor cibernetice. Rețelele moderne sunt frecvent vizate de atacuri precum:

- atacuri de tip *DDoS*, care blochează resursele sistemului prin suprasolicitare;
- spoofing, prin care un atacator își asumă identitatea unui dispozitiv legitim;
- interceptarea comunicațiilor în rețele wireless nesecurizate.

Aceste amenințări pot compromite confidențialitatea, integritatea și disponibilitatea serviciilor de rețea. Tanenbaum detaliază aceste scenarii în cadrul capitolului 8.6 din lucrarea *Computer Networks*, prezentând metode de apărare precum *IPsec*, firewall-uri și rețele virtuale private (*VPN*) [7, cap. 8.6, pp. 814–822]. Un aspect esențial subliniat de autor este lipsa criptării ca subiect principal de risc: datele transmise necriptat pot fi capturate și manipulate cu ușurință. În plus, multe rețele wireless nu implementează standarde de protecție moderne (precum WPA2), ceea ce facilitează accesul neautorizat. De asemenea, în lipsa unor politici riguroase de autentificare și filtrare, firewall-urile devin ineficiente în protejarea resurselor interne.

2.2 Probleme de performanță și scalabilitate

Pe lângă riscurile de securitate, performanța rețelei este adesea afectată de factori precum traficul mare de date care suprasolicită rețeaua, lipsa rutării optimizate sau absența mecanismelor de priorizare a pachetelor. În special în cazul aplicațiilor sensibile la latență cum ar fi comunicațiile *VoIP*, serviciile video sau bazele de date distribuite chiar și o mică întrerupere poate duce la probleme semnificative.

Aceste probleme pot apărea din:

- lipsa mecanismelor de *QoS* (Quality of Service),
- configurări incorecte ale protocoalelor de rutare,
- lipsa redundanței și a arhitecturilor de tip failover.

În subcapitolele unde sunt discutate aspecte despre performanță (5.3, 5.4, 6.6)[7, pp. 393,404,582], Tanenbaum explică modul în care aceste limitări pot fi reduse prin planificare atentă a topologiei rețelei și prin implementarea unor protocoale adaptative.

2.3 Lipsa autentificării și a controlului identității în rețea

Unul dintre cele mai subestimate riscuri în infrastructurile de rețea este lipsa controlului identității utilizatorilor și a dispozitivelor conectate. În lipsa unui mecanism de autentificare riguros, orice echipament care se conectează fizic la rețea poate avea acces imediat la resursele interne, ceea ce creează un punct critic de vulnerabilitate.

Standardul IEEE 802.1X oferă o metodă eficientă de autentificare la nivel de port, permițând acceptarea sau respingerea conexiunii în funcție de identitatea utilizatorului sau a dispozitivului. Implementarea acestui mecanism presupune integrarea cu un server *RADIUS* și configurarea corectă a switch-urilor. În lipsa acestuia, atacatorii pot introduce dispozitive malițioase în rețea, imitând echipamente legitime, fără a fi detectate. Mai mult, fără un sistem centralizat de gestionare a identității, este dificil de aplicat politici de acces sau de a urmări activitatea fiecărui utilizator.

2.4 Riscurile asociate firmware-ului învechit și lipsa actualizărilor

Un aspect adesea ignorat în managementul rețelelor informatice este legat de lipsa actualizărilor periodice ale sistemelor de operare și firmware-ului echipamentelor. Routerele, switch-urile, serverele și sistemele endpoint funcționează deseori ani întregi fără intervenții de mentenanță, ceea ce expune întreaga infrastructură la vulnerabilități critice.

Producătorii de echipamente de rețea, inclusiv Cisco, publică frecvent actualizări care corectează probleme de stabilitate și securitate. În lipsa acestor actualizări, rețeaua rămâne expusă la atacuri bine documentate, multe dintre ele înregistrate în bazele de date *CVE*. De exemplu, un atacator poate exploata o vulnerabilitate cunoscută dintr-o versiune veche de *IOS* sau firmware pentru a obține acces complet la dispozitiv sau pentru a redirecționa traficul în mod malițios.

Aceeași logică se aplică și în cazul serverelor și sistemelor de operare utilizate în rețea. Lipsa actualizărilor în distribuțiile de Linux, precum Ubuntu, sau în sistemele Windows, poate conduce la compromiterea completă a acestora prin exploit-uri active. Patch-urile de securitate publicate periodic de producători sunt esențiale pentru protecția împotriva atacurilor zero-day și pentru închiderea breșelor cunoscute. Prin

urmare, o strategie eficientă de protejare a rețelei trebuie să includă, pe lângă configurarea inițială corectă, și un plan de mentenanță continuă, actualizare periodică și audit al versiunilor de software utilizate.

2.5 Necesitatea unei abordări proactive

Tendințele actuale impun o schimbare de paradigmă: de la reacție, la prevenție. Abordarea „security by design” presupune integrarea măsurilor de protecție încă din etapa de proiectare a rețelei. Aceasta înseamnă implementarea combinată a următoarelor principii:

- segmentare a traficului în zone de securitate diferite (*VLAN*, *DMZ*),
- criptare a comunicațiilor prin *IPsec* sau *TLS*,
- utilizarea *VPN*-urilor pentru accesul de la distanță,
- autentificare strictă, bazată pe certificate sau standarde precum IEEE 802.1X,
- monitorizarea traficului și alertarea automată la anomalii, prin soluții de tip *IDS* și *IPS*.

Aceste aspecte nu doar că sporesc securitatea, dar contribuie și la un management mai ușor al rețelei.

3 Abordări existente

Lucrarea [3] analizează procesul de proiectare și implementare a unei rețele *VLAN* securizate, creată pentru a facilita schimbul de date în siguranță. Autorii s-au concentrat pe separarea logică a traficului utilizând *VLAN*-uri pentru izolarea diferitelor segmente ale rețelei, fiecare cu roluri și funcționalități distincte. Pentru gestionarea traficului între *VLAN*-uri și pentru asigurarea conectivității prin mai multe switch-uri, s-a utilizat standardul IEEE 802.1Q. Acest standard definește mecanismul de etichetare a cadrelor Ethernet pentru identificarea apartenenței lor la *VLAN*-uri, oferind astfel o metodă eficientă de segmentare și management al traficului. În plus, pentru a consolida securitatea rețelei, au fost configurate *ACL*-uri care limitează accesul între segmentele *VLAN*, protejând astfel resursele critice ale organizației. Lucrarea subliniază importanța implementării corecte a standardelor și politicilor de acces în rețele, oferind un exemplu clar de aplicare practică a soluțiilor de securitate avansate.

Un alt studiu relevant este [5] care detaliază procesul de proiectare și implementare a unei rețele *LAN* care să asigure o comunicare eficientă și securitate sporită. Autorii au propus o structură segmentată, în care traficul rețelei este organizat pe departamente, pentru a preveni suprasolicitarea și pentru a facilita managementul resurselor. Pentru a optimiza funcționarea, s-au utilizat protocoale precum *STP*, care elimină riscul buclelor în rețea și *EIGRP* pentru rutare rapidă și adaptabilă. Testele au fost realizate în GNS3, simulând diferite scenarii pentru a evalua viteza de transfer, stabilitatea rețelei și eficiența rutării. Această lucrare oferă un exemplu clar despre cum un design bine planificat și protocoale dinamice pot contribui la crearea unei rețele performante și sigure.

O altă contribuție semnificativă este lucrarea [2] care prezintă procesul de construire a unei rețele *LAN* robuste și sigure pentru un institut de cercetare cu cerințe stricte de securitate. Rețeaua a fost organizată în mai multe segmente dedicate activităților administrative, de cercetare și pentru utilizatorii finali, optimizând comunicarea internă și protejând resursele critice. Pentru rutare, s-a utilizat protocolul *OSPF*, iar securitatea suplimentară a fost asigurată prin autentificare la nivel de port folosind standardul 802.1X. Configurațiile și testele simulate în GNS3 au permis verificarea designului înainte de implementarea practică. Lucrarea evidențiază importanța utilizării unor politici stricte de control al accesului pentru protejarea datelor sensibile, demonstrând că designul propus îndeplinește cerințele atât de performanță, cât și cele de securitate.

Încă o lucrare importantă [1] abordează în profunzime procesul de dezvoltare a unei rețele pentru un campus al unei corporații, având ca scop configurarea unei infrastructuri complexe, dar eficiente. Proiectul propune un design care integrează elemente esențiale precum conectivitatea sigură, redundanța operațională și gestionarea traficului, asigurând totodată protecția resurselor interne ale organizației.

Un aspect central al studiului constă în utilizarea platformei GNS3 pentru a emula o rețea completă bazată pe dispozitive Cisco, permițând simularea funcționalităților rețelei în diferite scenarii. Autorul detaliază configurarea straturilor de rețea, precum stratul de acces, stratul de distribuție și stratul core, evidențiind cum acestea interacționează pentru a oferi performanță optimă. Prin utilizarea stratului DMZ și a conexiunilor VPN de tip site-to-site, s-a realizat o separare clară între traficul intern și cel extern, reducând expunerea infrastructurii la riscuri externe. Pe lângă dezvoltarea designului, autorul testează comportamentul rețelei în scenarii care includ utilizarea intensivă, pierderea conexiunilor și simularea unor atacuri cibernetice. Aceste teste au permis identificarea unor puncte critice de îmbunătățit și au oferit oportunitatea de a ajusta configurațiile pentru a spori reziliența în fața problemelor. Concluziile lucrării subliniază faptul că o rețea pentru un campus bine proiectată poate asigura nu doar o performanță tehnică ridicată, ci și un grad avansat de protecție pentru datele organizaționale. Simularea completă a designului propus în GNS3 a demonstrat utilitatea acestei platforme ca instrument pentru validarea unor soluții complexe, oferind un model demn de urmat pentru inginerii de rețele care se confruntă cu cerințe similare.

4 Fundamente teoretice

Acest capitol prezintă conceptele esențiale care stau la baza proiectării, configurării și securizării rețelei informatice simulate în cadrul lucrării. Sunt abordate echipamentele utilizate în infrastructura de rețea, protocoalele de rutare și segmentare, tehnologiile de securitate și control al accesului, precum și platformele software implicate în procesul de simulare. Prin înțelegerea acestor fundamente, se asigură o bază solidă pentru implementarea scenariilor practice analizate în capitolul următor.

4.1 Echipamente de rețea

O infrastructură de rețea modernă este construită dintr-o serie de echipamente specializate, fiecare având un rol bine definit în gestionarea traficului de date. În această lucrare, echipamentele cheie sunt simulate prin intermediul platformelor *Cisco Packet Tracer* și *GNS3*, incluzând componente precum **router**e, **switch-uri**, **firewall-uri**, precum și mașini virtuale care utilizează sistemele de operare Windows și Linux.

4.1.1 Routere Cisco

Routerele sunt dispozitive esențiale care operează la nivelul 3 al modelului *OSI* și sunt responsabile pentru direcționarea pachetelor între rețele distincte, pe baza adreselor *IP*. Acestea utilizează tabele de rutare statice sau dinamice (populate de protocoale precum *OSPF* sau *EIGRP*) pentru determinarea traseului optim.

Routerele Cisco folosesc sistemul de operare *IOS* (Internetwork Operating System), care oferă posibilitatea de a simula pe calculator aparatura Cisco similar cu aparatura fizică permițând configurarea din linia de comandă a dispozitivelor exact ca în scenarii reale.

Avantaje:

- Interoperabilitate extinsă cu alte protocoale și echipamente;
- Stabilitate și suport profesional pentru rețele enterprise;
- Posibilitatea de automatizare și securizare;

Dezavantaje:

- Necesită cunoștințe avansate de rețelistică pentru configurare eficientă;
- Costuri ridicate comparativ cu soluțiile open-source;
- Necesitatea licențierii pentru anumite funcționalități.

4.1.2 Switch-uri Cisco (Layer 2 și Layer 3)

Switch-urile Cisco sunt echipamente de rețea de nivel 2 sau 3, utilizate pentru conectarea dispozitivelor într-un *LAN* și, respectiv, pentru direcționarea traficului între segmente. Cele de nivel 2 funcționează pe baza adreselor *MAC* pentru a transmite pachetele de date, în timp ce modelele multilayer (Layer 3) adaugă funcționalități de rutare acestea combinând funcționalitățile unui router și ale unui switch, astfel putând ajuta la reducerea de exemplu a numărului de dispozitive necesare într-o rețea.

Funcționalități importante prezente pe switch-uri Cisco includ:

- *VLAN* și inter-*VLAN* routing;
- *STP* pentru prevenirea buclelor;
- Politici de securitate per port, inclusiv autentificare 802.1X;
- Suport pentru *QoS*, crucial în aplicații *VoIP* sau multimedia.

4.2 Protocoale de rutare și segmentare a rețelei

Funcționarea eficientă a unei rețele informatice presupune mai mult decât conectarea fizică a echipamentelor. Direcționarea traficului și organizarea logică a rețelei se bazează pe protocoale robuste, care permit atât rutarea pachetelor între subrețele distincte, cât și segmentarea internă pentru optimizarea performanței și a securității. Dintre cele mai importante tehnologii utilizate în acest scop, se remarcă protocoalele de rutare *OSPF* și *EIGRP*, metodele de segmentare logică prin *VLAN*, și mecanismele de redundanță precum *STP*.

4.2.1 Protocolul OSPF

OSPF (*Open Shortest Path First*) este un protocol de rutare de tip *link-state*, definit în *RFC 2328* de către *IETF* [4]. Utilizează algoritmul lui Dijkstra pentru a construi o topologie completă a rețelei și a calcula cea mai scurtă cale între noduri [7, p. 474].

OSPF organizează rețeaua în zone ierarhice, ceea ce reduce volumul de informații și îmbunătățește performanța. Routerule pe care este configurat *OSPF* fac schimb de informații de stare a legăturii folosind mesaje *LSA*, iar orice schimbare în topologie se propagă rapid în rețea [6, p. 443].

4.2.2 Protocolul EIGRP

EIGRP (*Enhanced Interior Gateway Routing Protocol*) este un protocol de rutare hibrid, dezvoltat de Cisco, care combină caracteristici ale protocoalelor de tip *distance vector* și *link-state*. Utilizează algoritmul *DUAL* (*Diffusing Update Algorithm*) pentru calculul rutei optime [6, p. 733].

Spre deosebire de *OSPF*, *EIGRP* nu este un protocol deschis, dar oferă convergență rapidă și un mecanism robust de actualizare a rutei. De asemenea, permite rutarea atât în rețele *IPv4*, cât și *IPv6*, păstrând mai multe baze de date interne.

4.2.3 Comparație între OSPF și EIGRP

În Tabelul 4.1 este prezentată o comparație între caracteristicile celor două protocoale de rutare.

Caracteristică	<i>OSPF</i>	<i>EIGRP</i>
Tip protocol	Link-State	Hibrid
Algoritm	Dijkstra	DUAL
Ierarhie în rețea	Da (Zone)	Nu
Convergență	Rapidă	Foarte rapidă
Standardizare	Deschis (RFC)	Cisco Proprietar
Suport pentru IPv6	Da	Da

Tabela 4.1: Comparație între protocoalele *OSPF* și *EIGRP*

4.2.4 Segmentarea rețelei prin VLAN

VLAN este o tehnologie de segmentare logică ce permite separarea traficului în cadrul aceleiași rețele fizice. Astfel, dispozitivele sunt grupate pe criterii funcționale, indiferent de locația fizică [6, p. 179]. Etichetarea cadrelor Ethernet este realizată prin protocolul IEEE 802.1Q, care adaugă un câmp de identificare VLAN în header-ul pachetului. Comunicarea între *VLAN*-uri se face prin rutare inter-*VLAN*, de obicei folosind interfețe *SVI* pe switch-uri Layer 3.[6, p. 392]

4.2.5 Prevenirea buclelor cu STP

STP (*Spanning Tree Protocol*) este un protocol de nivel 2 care previne apariția ciclurilor în rețelele redundante. Acesta dezactivează automat porturile neesențiale pentru a forma o topologie fără bucle [6, p. 210].

Funcționează pe baza mesajelor *BPDU* și stabilește un „root bridge”, cu ajutorul căruia se calculează căi optime. Variante moderne precum RSTP (*Rapid Spanning Tree Protocol*) reduc semnificativ timpul de convergență în cazul modificărilor rețelei.

4.3 Tehnologii de securitate și control al accesului

Într-un context în care rețelele informatice sunt tot mai expuse la riscuri, securitatea devine un proces activ și continuu. Nu mai este suficientă izolarea fizică sau simpla utilizare a unui firewall controlul eficient al accesului presupune identificarea, autentificarea și autorizarea fiecărui dispozitiv și utilizator. Acest lucru este realizat printr-o combinație de tehnologii precum listele de control al accesului (*ACL*), autentificarea la nivel de port prin IEEE 802.1X, utilizarea de servere *RADIUS* și politici de acces distribuite.

4.3.1 Listele de control al accesului (ACL)

ACL-urile sunt instrumente esențiale în filtrarea traficului de rețea, fiind implementate pe interfețele routerelor sau switch-urilor pentru a permite sau bloca pachetele în

funcție de anumite criterii. Aceste criterii pot include adrese *IP*, protocoale, numere de port sau direcția traficului (*inbound* sau *outbound*).

Există două tipuri principale: *ACL* standard, care operează pe baza adresei *IP* sursă, și *ACL* extins, care permite un control mai detaliat, inclusiv după *IP* destinație, tip de protocol și porturi specifice. În mediile enterprise, *ACL*-urile sunt folosite nu doar pentru securitate, ci și pentru segmentare logică și controlul amănunțit al fluxurilor de date.

Deși sunt eficiente și ușor de implementat, *ACL*-urile nu oferă inspecție la nivel de aplicație, ceea ce le limitează în contextul unor amenințări sofisticate. În plus, întreținerea lor în rețele mari devine complicată, necesitând o bună planificare și documentare.

4.4 Sisteme de operare utilizate în infrastructura simulată

În cadrul infrastructurii simulate din această lucrare, vor fi utilizate două sisteme de operare principale instalate pe mașini virtuale: **Windows 10** și **Ubuntu**. Alegerea acestor sisteme reflectă realitatea din majoritatea rețelelor educaționale sau organizaționale, unde utilizatorii finali accesează resursele rețelei prin astfel de medii.

4.4.1 Stații de lucru Windows 10

Windows 10 este un sistem de operare larg răspândit în mediul instituțional, fiind utilizat în această simulare pentru testarea funcționalităților de bază precum:

- conectarea la rețea și obținerea configurației *IP*;
- interacțiunea cu diverse servicii de rețea (ex. *HTTP*, *DNS*);
- reacția la politici de rețea sau scenarii de securitate simulate.

4.4.2 Stații de lucru Ubuntu

Ubuntu (varianta desktop) este utilizat ca alternativă open-source pentru testarea comportamentului într-un mediu Linux. Sistemul oferă acces facil la unelte de analiză, monitorizare și configurare, fiind util în scenariile de simulare a accesului și depanării la nivel de client.

Considerații finale

Utilizarea acestor două tipuri de stații de lucru virtuale permite observarea diferențelor de comportament între sisteme Windows și Linux în aceeași infrastructură de rețea. Acest aspect este esențial pentru evaluarea compatibilității, funcționalității și reacției în fața unor configurații și evenimente simulate în scop didactic sau de testare.

4.5 Platforme și medii de simulare

Simularea unei rețele informatice este esențială pentru testarea funcționalității, comportamentului și securității infrastructurii, fără a fi necesare echipamente fizice reale. În această lucrare au fost utilizate două platforme complementare: *Cisco Packet Tracer* și *GNS3*. Ambele au fost folosite pentru construirea aceleiași rețele, cu aceleași configurații pe dispozitivele de rețea (routere și switch-uri). Diferența principală constă în scopul utilizării fiecăreia: Packet Tracer pentru o vizualizare clară a topologiei, iar GNS3 pentru integrarea mașinilor virtuale și simularea atacurilor.

4.5.1 Cisco Packet Tracer

Cisco Packet Tracer este un simulator dezvoltat de Cisco, utilizat frecvent în medii educaționale pentru proiectarea și testarea rețelelor. Platforma permite conectarea și configurarea de routere, switch-uri, stații de lucru, servere sau alte dispozitive, într-o interfață grafică simplă.

În această lucrare, Packet Tracer a fost utilizat pentru:

- construirea topologiei complete a rețelei simulate;
- vizualizarea structurii fizice și logice a rețelei;
- testarea configurațiilor de bază (VLAN, ACL, rutare).

Principalul motiv pentru alegerea acestei platforme a fost posibilitatea de a urmări rapid și intuitiv modul în care dispozitivele sunt conectate și interacționează între ele. De asemenea, este utilă pentru prezentări și demonstrații.

4.5.2 GNS3 (Graphical Network Simulator 3)

GNS3 este un instrument de simulare care permite rularea de imagini reale Cisco IOS (precum IOSv) și integrarea acestora cu mașini virtuale (Windows, Linux, etc.), conectate direct în rețea. Configurațiile realizate în GNS3 sunt identice cu cele din Packet Tracer, dar platforma oferă funcționalități suplimentare esențiale pentru scopurile acestei lucrări.

În cadrul lucrării, GNS3 a fost utilizat pentru:

- emularea realistă a routerelor și switch-urilor cu IOSv;
- integrarea de mașini virtuale cu Ubuntu și Windows 10;
- simularea scenariilor de atac cibernetic (ARP Spoofing, DHCP starvation, ICMP flood);

GNS3 a fost ales deoarece permite conectarea rețelei simulate la stații reale sau virtuale, oferind astfel un mediu complet pentru testarea securității și comportamentului la nivel de client. Acest aspect nu poate fi realizat în Cisco Packet Tracer.

Considerații finale

Atât *Cisco Packet Tracer*, cât și *GNS3* au fost utilizate pentru construirea aceleiași infrastructuri simulate. Packet Tracer a fost preferat pentru claritatea vizuală și simplitatea interfeței, fiind potrivit pentru prezentarea generală a topologiei. GNS3 a fost folosit pentru testarea realistă a scenariilor de securitate, datorită posibilității de a include mașini virtuale și de a rula imagini IOS reale. Combinația acestor platforme permite o analiză completă în ceea ce privește funcționalitatea rețelei și vulnerabilitățile acesteia.

4.5.3 VMware Workstation

Pentru rularea mașinilor virtuale utilizate în simulare, lucrarea folosește platforma *VMware Workstation*. Aceasta permite crearea și gestionarea de mașini virtuale Windows 10 și Ubuntu, folosite ca stații de lucru în rețeaua proiectată în GNS3.

În plus, VMware Workstation este utilizat pentru rularea componentei *GNS3 VM*, necesară pentru funcționarea optimă a platformei GNS3, inclusiv pentru emularea routerelor cu imagini IOSv.

- asigură rularea mașinilor virtuale Windows și Linux conectate la topologia GNS3;
- permite testarea scenariilor de rețea din perspectiva utilizatorului final;
- oferă suport pentru integrarea GNS3 VM cu infrastructura emulată.

Utilizarea VMware Workstation contribuie astfel la simularea realistă a comportamentului stațiilor de lucru și la testarea interacțiunii acestora cu restul infrastructurii de rețea.

4.5.4 Wireshark

Wireshark este cel mai popular instrument de analiză a traficului de rețea, utilizat pentru capturarea și inspectarea pachetelor de date în timp real. Acesta este esențial în etapa de evaluare a comportamentului rețelei, permițând identificarea anomaliilor, erorilor de configurare sau a tentativelor de atac.

Funcționalități cheie:

- captură de pachete pe interfețele virtuale GNS3/VMware;
- analiză a protocoalelor (ARP, ICMP, HTTP, DNS, etc.);
- identificare atacuri de tip spoofing, sniffing, flood.

Wireshark permite observarea directă a modului în care datele circulă într-o rețea. Este folosit în această lucrare pentru a valida comportamentul infrastructurii în condiții normale și sub atac.

4.6 Tipuri de atacuri cibernetice simulate

Simularea unor atacuri cibernetice simple, dar relevante, permite identificarea vulnerabilităților rețelei și familiarizarea cu comportamentul infrastructurii în condiții ostile. În cadrul acestei lucrări au fost selectate patru scenarii de atac ușor de implementat în GNS3, utilizând o mașină virtuală Kali Linux conectată la rețeaua construită.

Atacurile alese se concentrează pe vulnerabilitățile frecvent întâlnite în rețelele informatice, cu impact direct asupra disponibilității, autenticității sau confidențialității datelor.

4.6.1 ARP Spoofing

ARP Spoofing este o tehnică de atac de nivel 2 prin care un dispozitiv malițios trimite pachete ARP falsificate în rețea, asociind adresa MAC a atacatorului cu adresa IP a unei alte gazde. Scopul este redirectionarea traficului prin dispozitivul atacator, facilitând interceptarea sau manipularea comunicațiilor.

Tool-uri utilizate: arpspoof

4.6.2 DHCP Starvation

DHCP Starvation este un atac de tip denial-of-service (DoS), prin care un atacator trimite un număr mare de cereri DHCP false, folosind adrese MAC diferite, cu scopul de a epuiza pool-ul de adrese al serverului DHCP. Astfel, clienții legitimi nu mai pot obține o adresă IP validă.

Tool-uri utilizate: Yersinia

4.6.3 ICMP Flood

ICMP Flood este un atac simplu de tip DoS, care implică trimiterea unui volum mare de pachete ICMP (ping) către o gazdă sau interfață de rețea, cu scopul de a supra-solicita dispozitivele, afectând disponibilitatea serviciilor de care dispozitivele atacate se ocupă.

Tool-uri utilizate: hping3

4.6.4 Unelte Kali Linux utilizate

Distribuția Kali Linux oferă un set variat de unelte specializate pentru testarea securității, utilizate frecvent în scopuri educaționale sau de simulare controlată. Pentru atacurile cibernetice prezentate în această lucrare, au fost selectate următoarele aplicații:

- **arpspoof** – parte din pachetul **dsniff**, utilizat pentru trimiterea de pachete ARP falsificate;
- **Yersinia** – aplicație pentru testarea protocoalelor de nivel 2, inclusiv DHCP;
- **hping3** – generator de trafic pentru ICMP Flood;
- **nmap** – scanare de porturi, detecție de servicii, identificare de sisteme de operare.

Utilizarea acestor instrumente are un rol exclusiv academic, pentru observarea comportamentului infrastructurii în fața unor atacuri simulate, în condiții controlate. Aceste scenarii vor fi detaliate în capitolul practic al lucrării, unde se va urmări reproducerea efectivă a atacurilor și impactul acestora asupra rețelei.

5 Metodologia de realizare

În acest capitol este prezentat, pas cu pas, modul de realizare a infrastructurii simulate în GNS3 și în Cisco Packet Tracer. Sunt detaliate configurările efectuate pe fiecare echipament din rețea, organizarea pe VLAN-uri, rutarea, alocarea adreselor IP, precum și testarea conectivității. De asemenea, este descrisă simularea a trei atacuri cibernetice (ARP Spoofing, DHCP Starvation și ICMP Flood), urmată de implementarea măsurilor de protecție necesare pentru prevenirea acestor atacuri.

5.1 Zona CORE

Zona CORE reprezintă nucleul infrastructurii de rețea simulate. Această zonă are rolul de a interconecta toate celelalte componente ale rețelei: switch-urile care deservesc sălile de laborator, zona administrativă și legătura cu rețeaua externă (ISP). Din punct de vedere logic, aici se realizează rutarea traficului între VLAN-uri, distribuirea adreselor IP prin DHCP și translatarea adreselor IP pentru accesul la Internet.

Topologia zonei CORE include un router central (CORE-ROUTER), un switch principal (CORE-SW) și două switch-uri (SW-1-15 și SW-16-31). Acestea sunt conectate prin legături de tip trunk, astfel încât toate VLAN-urile definite pot fi propagate de la switch-uri către router și invers. Toate serviciile fundamentale sunt administrate din această zonă.

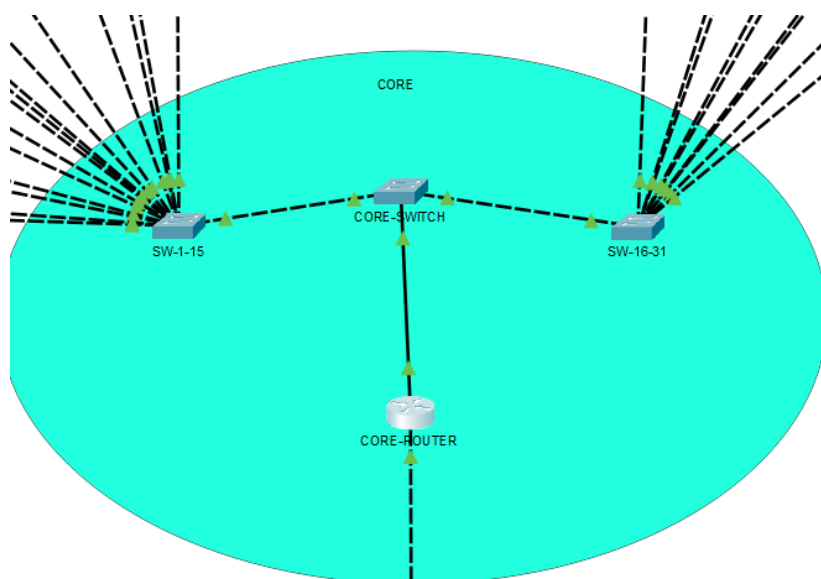


Figura 5.1: Structura zonei CORE realizată în Cisco Packet Tracer

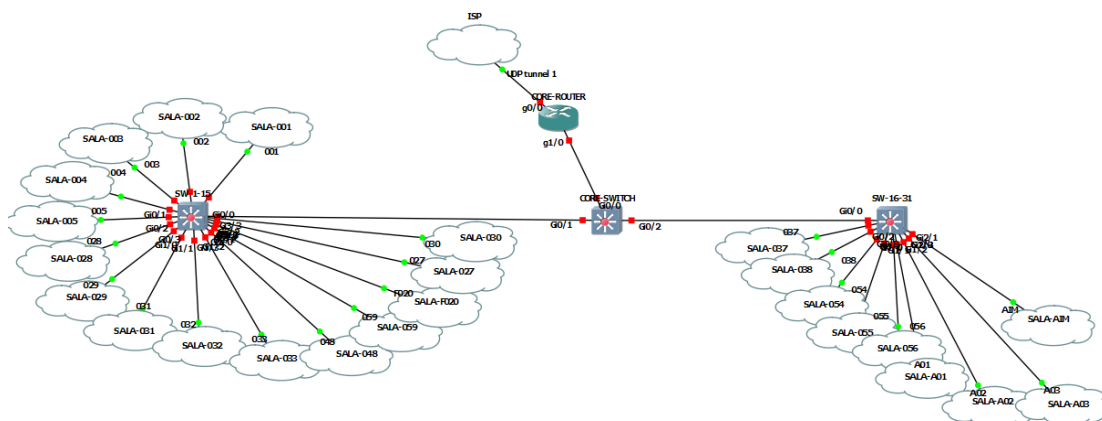


Figura 5.2: Structura zonei CORE implementată în GNS3

5.1.1 Routerul central – CORE-ROUTER

Dispozitivul **CORE-ROUTER** are rolul de nod central de rutare în infrastructura proiectată. Acesta este responsabil pentru realizarea comunicației între toate VLAN-urile definite în rețea, pentru alocarea automată a adreselor IP către dispozitivele conectate, precum și pentru asigurarea conectivității cu rețeaua externă. Configurația realizată pe acest echipament reflectă un model clasic de rețea segmentată logic, cu rutare inter-VLAN și acces la Internet prin NAT (Network Address Translation).

Conectarea la zona internă a rețelei se face prin interfața **GigabitEthernet0/0/0**, care este împărțită logic în multiple subinterfețe câte una pentru fiecare VLAN existent. Fiecare subinterfață este configurată cu encapsulare 802.1Q (**dot1q**), utilizând ID-ul corespunzător VLAN-ului respectiv, și o adresă IP care acționează ca default gateway pentru acel segment. De exemplu, pentru VLAN 10, interfața este definită astfel:

```
interface GigabitEthernet0/0/0.10
  encapsulation dot1q 10
  ip address 192.168.10.1 255.255.255.0
  ip nat inside
```

Această abordare permite segmentarea logică a traficului în rețea și asigură rutarea locală între toate subrețelele. Subinterfețele sunt corespunzătoare diferitelor săli, departamente sau din rețea. Marcarea cu **ip nat inside** indică faptul că traficul din aceste rețele va fi supus translării NAT atunci când părăsește rețeaua locală.

Pentru alocarea dinamică a adreselor IP către stațiile de lucru, routerul conține câte un pool DHCP pentru fiecare rețea locală. Fiecare pool specifică adresa de rețea, masca de subrețea, gateway-ul (care este adresa subinterfeței asociate VLAN-ului) și serverul DNS, în acest caz setat la 8.8.8.8. De asemenea, adresele IP atribuite interfețelor routerului sunt excluse explicit pentru a evita conflictele:

```
ip dhcp excluded-address 192.168.10.1
ip dhcp pool VLAN001
  network 192.168.10.0 255.255.255.0
  default-router 192.168.10.1
  dns-server 8.8.8.8
```


Această metodă permite configurarea automată a clienților din fiecare VLAN, fără intervenție manuală, și menține o structură administrativă clară.

Accesul către rețeaua externă (Internet) este asigurat prin interfața `GigabitEthernet0/0/1`, configurată cu o adresă IP publică alocată static. Pentru a permite comunicația între dispozitivele din rețea (cu adrese IP private) și Internet, routerul aplică mecanismul de traducere a adreselor IP folosind *Port Address Translation* (PAT). Aceasta permite mai multor dispozitive din rețea să partajeze o singură adresă IP publică prin folosirea unor porturi diferite:

```
interface GigabitEthernet0/0/1
  ip address 203.10.10.1 255.255.255.252
  ip nat outside

access-list 1 permit any
ip nat inside source list 1 interface GigabitEthernet0/0/1 overload
```

Regula de NAT asigură traducerea tuturor adreselor sursă interne definite în `access-list 1`, utilizând adresa interfeței externe. Configurația este standard pentru rețele enterprise cu adresare privată și permite conectivitate completă la Internet pentru toate stațiile din VLAN-uri.

Pentru rutarea între subrețele și pentru viitoare scenarii de extindere, a fost activat protocolul de rutare OSPF (Open Shortest Path First). Toate rețelele locale sunt anunțate dinamic în area 0, acoperind întreaga clasă de adrese 192.168.0.0/16, ceea ce simplifică adăugarea de noi rețele fără a modifica configurări statice:

```
router ospf 1
  network 192.168.0.0 0.0.255.255 area 0
```

În topologie `CORE-ROUTER` este conectat direct la `CORE-SWITCH` și la echipamentul desemnat ca furnizor ISP. Interfața fizică către zona internă (`G0/0/0`) preia traficul de la switch, iar interfața externă (`G0/0/1`) asigură conectivitatea cu exteriorul.

Prin centralizarea rutării, a serviciilor DHCP și NAT, `CORE-ROUTER` este componenta esențială care asigură funcționarea întregii rețele simulate. Configurația sa este scalabilă, modulară și ușor de adaptat pentru scenarii suplimentare de testare și analiză.

5.1.2 Switch-ul central – CORE-SW

Dispozitivul `CORE-SW` reprezintă punctul de interconectare principal al rețelei simulate. Acesta face legătura între routerul central și cele două switch-uri de distribuție (`SW-1-15` și `SW-16-31`), asigurând propagarea VLAN-urilor către toate zonele rețelei.

Toate porturile active sunt configurate în mod `trunk`, permițând transportul pachetelor etichetate cu VLAN-uri diferite pe aceleași legături fizice. În continuare sunt prezentate cele trei interfețe esențiale care leagă `CORE-SW` de restul infrastructurii:

Interfața `FastEthernet0/1` Această interfață conectează `CORE-SW` la `CORE-ROUTER` și este configurată pentru a permite traficul tuturor VLAN-urilor definite în rețea. Această decizie permite routerului să aibă vizibilitate completă asupra tuturor segmentelor definite:

```
interface FastEthernet0/1
  switchport trunk allowed vlan 10,20,30,40,50,120,270,280,290,
  300,310,320,330,370,380,480,513,540,550,560,570,580,590,610,
  620,900-903
  switchport mode trunk
```

Interfața FastEthernet0/2 Portul Fa0/2 leagă CORE-SW de switch-ul SW-1-15, care gestionează 15 săli/zone/subrețele. În acest caz, sunt permise doar VLAN-urile corespunzătoare acestor săli, pentru a limita traficul la segmentele relevante:

```
interface FastEthernet0/2
  switchport trunk allowed vlan 10,20,30,40,50,120,280,290,300,
  310,320,330,480,590
  switchport mode trunk
```

Interfața FastEthernet0/3 Această interfață asigură conexiunea cu switch-ul SW-16-31, dedicat altor săli și zone administrative. Setul de VLAN-uri permis este diferit și adaptat în funcție de segmentarea logică din această parte a rețelei:

```
interface FastEthernet0/3
  switchport trunk allowed vlan 370,380,540,550,560,900-903
  switchport mode trunk
```

Switch-ul folosește modul PVST (Per VLAN Spanning Tree), care oferă o instanță separată a protocolului Spanning Tree pentru fiecare VLAN în parte. Aceasta asigură protecție împotriva buclelor de rețea și permite o convergență mai rapidă în cazul modificării topologiei:

```
spanning-tree mode pvst
spanning-tree extend system-id
```

CORE-SW este poziționat între routerul principal și cele două switch-uri (SW-1-15, SW-16-32), gestionând propagarea etichetelor VLAN între acestea. Prin această structură, rețeaua este logic segmentată, fiecare grup de săli având acces la gateway-ul său prin intermediul CORE-ROUTER, dar fără a permite trafic nedorit între zone nespecifice. Switch-ul acționează astfel ca un distribuitor VLAN centralizat, păstrând organizarea logică și claritatea fluxului de date în toată infrastructura.

5.1.3 Switch-ul de distribuție – SW-1-15

Switch-ul SW-1-15 are rolul de agregator pentru primele 15 săli din infrastructura simulată. Acesta este conectat direct la CORE-SW prin interfața FastEthernet0/1, configurată ca trunk pentru a permite traficul etichetat dintr-o selecție de VLAN-uri. VLAN-urile acceptate corespund exact sălilor de laborator pe care acest switch le deservește, evitând propagarea inutilă a altor segmente:

```

interface FastEthernet0/1
  switchport trunk allowed vlan 10-11,20,30,40,50,81-83,120,
  270,280,290,300,310,320,330,370,380,480,513-514,540,550,560,
  570,580,590,610,620
  switchport mode trunk
  no cdp enable

```

Pe porturile de acces FastEthernet0/2 până la FastEthernet0/21, fiecare port este configurat pentru un VLAN specific, corespunzător unei săli de laborator. De exemplu, portul Fa0/2 este asociat VLAN-ului 10, iar Fa0/3 cu VLAN 20, și așa mai departe. Această configurare permite o asociere statică între o sală și un VLAN dedicat, păstrând segmentarea rețelei și izolarea traficului:

```

interface FastEthernet0/2
  switchport access vlan 10
  no cdp enable

```

```

interface FastEthernet0/3
  switchport access vlan 20
  no cdp enable

```

...

```

interface FastEthernet0/21
  switchport access vlan 560

```

Comanda `no cdp enable` este aplicată pe majoritatea porturilor pentru a dezactiva protocolul CDP (Cisco Discovery Protocol), reducând expunerea rețelei la posibile atacuri de tip reconnaissance și păstrând o configurație mai curată din punct de vedere al securității.

SW-1-15 contribuie la segmentarea logică a rețelei prin implementarea VLAN-urilor pentru fiecare sală, direcționând traficul către CORE-SW, care mai departe îl transmite către CORE-ROUTER pentru rutare inter-VLAN. Această organizare menține controlul și scalabilitatea infrastructurii, oferind în același timp flexibilitate pentru adăugarea de noi segmente.

5.1.4 Switch-ul de distribuție – SW-16-31

SW-16-31 este al doilea switch de agregare din cadrul zonei CORE și deservește un alt grup de săli, împreună cu câteva departamente funcționale. Acesta este conectat la CORE-SW prin interfața FastEthernet0/1, configurată ca trunk pentru a permite propagarea unui subset specific de VLAN-uri necesare în această zonă:

```

interface FastEthernet0/1
  switchport trunk allowed vlan 300,370,380,540,550,560,900-903
  switchport mode trunk

```

VLAN-urile acceptate pe acest port corespund segmentelor pentru sălile 16 – 31.

Fiecare dintre porturile de acces este asociat unui VLAN individual, într-o manieră statică. Această abordare permite o separare logică clară și o gestionare predictibilă a traficului:

```
interface FastEthernet0/2
  switchport access vlan 300
  no cdp enable
```

```
interface FastEthernet0/3
  switchport access vlan 370
  no cdp enable
```

...

```
interface FastEthernet0/11
  switchport access vlan 900
  no cdp enable
```

Porturile `FastEthernet0/2` până la `Fa0/11` sunt utilizate pentru conectarea stațiilor sau dispozitivelor specifice fiecărei săli sau departament.

`SW-16-31` completează structura zonei `CORE`, asigurând distribuția VLAN-urilor în partea secundară a rețelei simulate. Traficul este direcționat centralizat către `CORE-SW`, iar rutarea și serviciile de rețea sunt tratate ulterior de `CORE-ROUTER`. Această arhitectură modulară permite extinderea facilă și controlul clar al segmentării traficului, adaptat pe săli sau funcționalități.

5.2 Noduri de tip Cloud și configurarea tunelurilor UDP

În cadrul infrastructurii simulate în GNS3, interconectarea proiectelor individuale (corespunzătoare fiecărei săli) cu zona centrală a rețelei (`CORE`) este realizată prin intermediul unor noduri de tip `Cloud`. Aceste noduri nu sunt dispozitive active cu interfață CLI, ci elemente speciale oferite de GNS3 care permit conectarea logică între proiecte distincte prin intermediul tunelurilor UDP.

Fiecare sală este construită ca un proiect separat în GNS3 și conține un nod `Cloud` care asigură legătura cu infrastructura centrală. La rândul său, proiectul `CORE` conține câte un nod `Cloud` dedicat pentru fiecare sală, cu care stabilește conexiunea UDP.

Tunelul UDP este bidirecțional și este format dintr-o pereche de porturi unice pentru fiecare sală. Configurația implică definirea unui port local și a unui port remote în ambele sensuri. Pe nodul `Cloud` dintr-o sală, portul local definit va fi setat ca port remote pe nodul corespunzător din `CORE`, iar portul remote configurat în sală va fi portul local din `CORE`. Comunicarea se face exclusiv local, toate nodurile având ca adresă IP a gazdei remote `127.0.0.1` (localhost).

Un exemplu concret de interconectare între o sală și zona centrală a rețelei este cel dintre SALA-001 și proiectul CORE. Ambele proiecte conțin câte un nod de tip Cloud, care gestionează legătura UDP.

- În proiectul CORE, nodul Cloud asociat sălii 001 este configurat astfel:
 - Port local: 11001
 - Port remote: 10001
- În proiectul SALA-001, nodul Cloud are configurat:
 - Port local: 10001
 - Port remote: 11001

Această configurație permite traficului de date să circule în ambele direcții, fără a ieși în afara mașinii fizice pe care rulează GNS3, întrucât toate conexiunile se fac la adresa 127.0.0.1 (localhost).

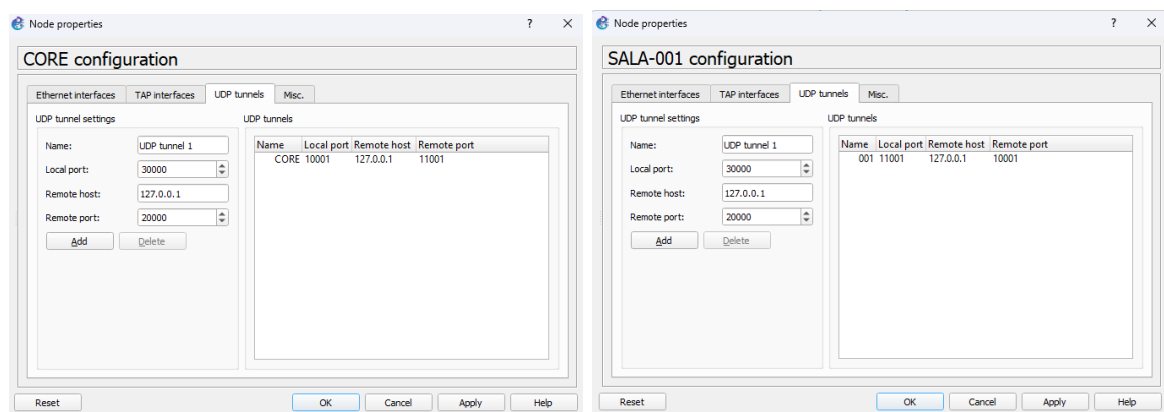


Figura 5.3: Configurarea tunelului UDP dintre CORE și nodul SALA-001

Aceeași structură este urmată pentru toate celelalte săli, fiecare având alocate porturi unice, numerotate coerent. Porturile locale din proiectele cu fiecare sală sunt în intervalul 10001 – 10900, iar cele corespunzătoare din CORE sunt în intervalul 11001 – 11900. Acest sistem simplifică mentenanța și elimină ambiguitățile.

Exemple suplimentare de configurare a tunelurilor UDP pe noduri Cloud sunt prezentate în Anexa 1.

Tunelurile UDP astfel configurate permit transmiterea integrală a traficului de rețea, inclusiv a pachetelor Layer 2 și Layer 3, fără a afecta structura logică a infrastructurii. Această soluție a fost esențială pentru a putea împărți rețeaua în proiecte distincte.

Această metodă de conectare oferă flexibilitate ridicată în dezvoltarea și testarea modulară a rețelei, menținând în același timp o infrastructură funcțională complet interconectată. Tunelurile UDP din GNS3 au fost alese ca soluție tehnică viabilă pentru a depăși limitările impuse de scalabilitatea proiectelor complexe în GNS3.

5.3 Zona ISP și conectarea la rețeaua externă

Pentru a permite testarea conectivității la Internet, infrastructura simulată în GNS3 include un proiect separat pentru zona ISP. Acest proiect conține un router Cisco, numit ISP-ROUTER, care are rolul de a emula un furnizor de Internet (ISP) pentru rețeaua internă.

5.3.1 Conectivitate cu Internetul

Routerul ISP-ROUTER este conectat prin interfața `GigabitEthernet0/0` la un nod de tip `Cloud` (NAT), furnizat de GNS3. Acest nod preia conexiunea de Internet a mașini fizice pe care rulează GNS3 și o expune proiectului ca o conexiune de tip NAT. Pe interfața respectivă este activat DHCP, astfel încât routerul poate obține o adresă IP validă de la sistemul gazdă:

```
interface GigabitEthernet0/0
description Link To NAT (GNS3 Machine)
ip address dhcp
ip nat outside
```

Astfel, rețeaua simulată poate accesa Internetul prin intermediul acestui port.

5.3.2 Conectarea cu infrastructura internă (CORE)

Legătura dintre ISP și infrastructura internă (proiectul CORE) este realizată printr-un tunel UDP, folosind noduri `Cloud` GNS3, la fel ca în cazul sălilor. Interfața `GigabitEthernet1/0` de pe ISP-ROUTER este configurată cu o adresă IP statică și cu masca 255.255.255.252 pentru că este o conexiunea doar cu CORE-ROUTER:

```
interface GigabitEthernet1/0
description Link To CoreRouter
ip address 203.10.10.2 255.255.255.252
ip nat inside
```

Această interfață transmite mai departe traficul intern către routerul central. NAT-ul este activat pentru ca toate adresele IP din rețeaua internă să fie traduse prin IP-ul public obținut prin DHCP:

```
ip nat inside source list 1 interface GigabitEthernet0/0 overload
access-list 1 permit any
```

5.3.3 Configurația completă a routerului ISP

```
hostname ISP-ROUTER
```

```
interface GigabitEthernet0/0
description Link To NAT (GNS3 Machine)
ip address dhcp
```

```

ip nat outside

interface GigabitEthernet1/0
description Link To CoreRouter
ip address 203.10.10.2 255.255.255.252
ip nat inside

interface GigabitEthernet2/0
no ip address

ip name-server 8.8.8.8
ip nat inside source list 1 interface GigabitEthernet0/0 overload
access-list 1 permit any

```

5.3.4 Limitări și observații

Conectivitatea la Internet a fost posibilă doar în mediul GNS3. Cisco Packet Tracer nu permite conectarea la Internetul real, motiv pentru care zona ISP nu a putut fi replicată în acel mediu.

Performanța conexiunii la Internet este influențată de mai mulți factori:

- performanța mașini fizice care rulează GNS3;
- resursele consumate de dispozitivele IOSv (CPU, RAM);
- capacitatea GNS3 de a gestiona simultan mai multe proiecte active.

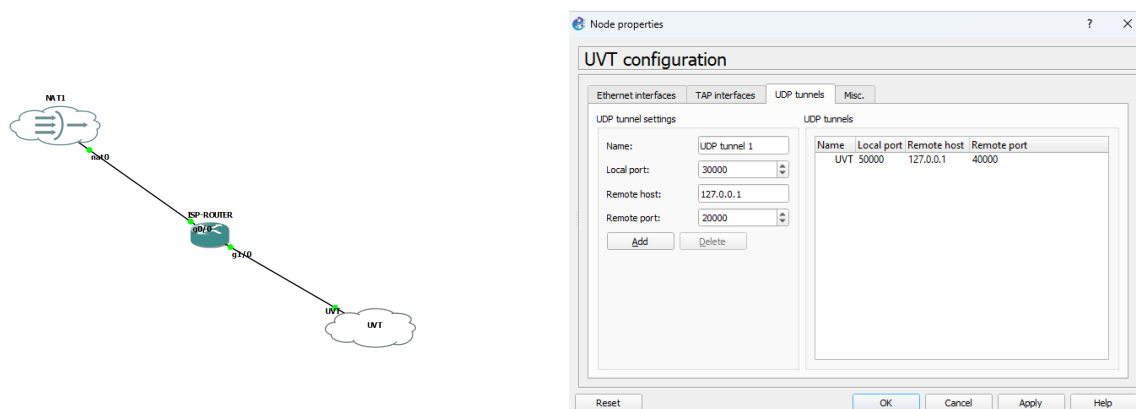


Figura 5.4: Zona ISP în GNS3 și interconectarea prin tunel UDP

5.4 Structura sălilor individuale

Fiecare sală din infrastructură este simulată ca proiect separat în GNS3. În funcție de scopul fiecărei săli, topologia diferă — atât ca număr de stații de lucru, cât și ca tip de echipamente.

În figura de mai jos este prezentată o sală de laborator și una destinată unui departament administrativ. În sala departamentului administrativ sunt doar câteva stații de lucru, un switch și câteva echipamente suplimentare precum o imprimantă și două mașini virtuale (Windows și Linux), utilizate pentru testare. În schimb, sălile de laborator conțin un număr mai mare de stații de lucru, fiecare conectată la un switch dedicat.

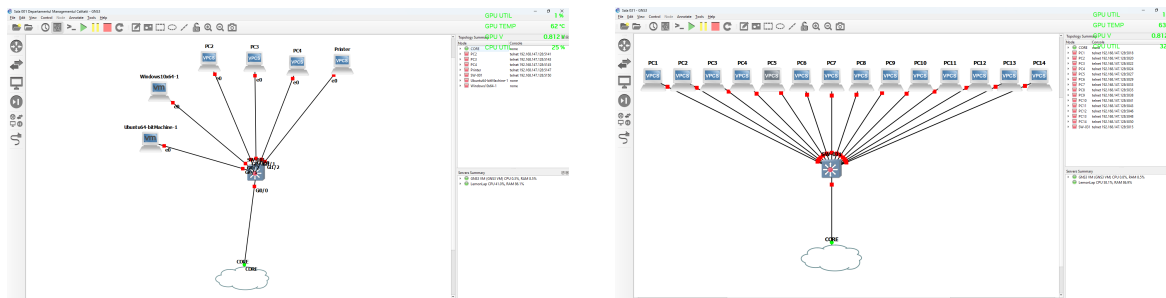


Figura 5.5: Structură săli în GNS3

Toate echipamentele din fiecare sală sunt conectate la un switch local, iar acest switch este legat printr-o singură conexiune la zona CORE printr-un nod **Cloud**. Nu a fost necesară nicio configurare pe switch-urile locale, deoarece toate funcționalitățile de rețea, inclusiv VLAN-urile, DHCP-ul și rutarea, sunt gestionate centralizat din infrastructura CORE. Această abordare permite o administrare simplificată și scalabilă a întregii rețele.

5.5 Simularea atacurilor cibernetice

În această secțiune sunt prezentați pașii concreți pentru simularea atacurilor cibernetice, testate direct în rețeaua realizată, folosind o mașină Kali Linux integrată în sala 003.

5.5.1 Scanarea rețelei folosind nmap

Înainte de inițializarea atacurilor propriu-zise, a fost necesară o etapă preliminară de recunoaștere a dispozitivelor active în rețea și a serviciilor expuse de acestea. Pentru acest scop a fost utilizată unealta **nmap**, un scanner de rețea, capabil să identifice hosturi active, porturi deschise, versiuni ale serviciilor și sistemele de operare de pe dispozitive. Scanarea s-a realizat din interiorul rețelei, folosind mașina virtuală Kali Linux.

Pentru a identifica dispozitivele active din rețea, a fost utilizată o scanare de tip *ping sweep*, folosind comanda:

```
nmap -sn 192.168.30.0/24
```

Această comandă face ca utilitarul **nmap** să trimită pachete ICMP Echo Request, împreună cu alte metode pasive, pentru a determina ce adrese IP din subrețea sunt active. Semnificația opțiunilor este următoarea:

- `-sn` (*Ping Scan*): mod în care `nmap` nu efectuează scanări de porturi, ci doar verifică ce hosturi sunt active;
- `192.168.30.0/24`: reprezintă spațiul de adrese IP din rețea pentru a fi scanate, cu mască de subrețea `255.255.255.0`

Output-ul obținut a identificat următoarele dispozitive active:

```
Nmap scan report for 192.168.30.1
Host is up (0.040s latency).
Not shown: 999 closed tcp ports (reset)
PORT      STATE SERVICE
23/tcp    open  telnet
MAC Address: CA:01:10:81:00:1C (Unknown)
```

```
Nmap scan report for 192.168.30.4
Host is up (0.093s latency).
All 1000 scanned ports on 192.168.30.4 are in ignored states.
Not shown: 1000 filtered tcp ports (no-response)
MAC Address: 00:0C:29:69:A4:15 (VMware)
```

```
Nmap scan report for 192.168.30.135
Host is up (0.0000030s latency).
All 1000 scanned ports on 192.168.30.135 are in ignored states.
Not shown: 1000 closed tcp ports (reset)
```

Nmap done: 256 IP addresses (3 hosts up) scanned in 25.40 seconds

Adresa `192.168.30.135` corespunde mașinii Kali Linux, de pe care a fost efectuată scanarea. Adresa `192.168.30.1` este gateway-ul rețelei (routerul principal/server-ul de DHCP). Stația `192.168.30.4` a fost identificată ca un posibil sistem de operare Windows, pe care l-am selectat ulterior pentru a fi folosit ca victimă în testele următoare.

Scanarea detaliată a mașinii țintă

După identificarea stațiilor active din rețea, am realizat o scanare detaliată a dispozitivului cu adresa IP `192.168.30.4`, folosind comanda:

```
nmap -sS -sV -O -Pn 192.168.30.4
```

Această scanare are rolul de a descoperi porturile deschise, serviciile care rulează și sistemul de operare utilizat de gazdă. Semnificația fiecărei opțiuni este următoarea:

- `-sS` (*TCP SYN scan*): realizează o scanare de porturi folosind pachete TCP cu flag-ul SYN activat. Este o metodă rapidă și relativ discretă, deoarece nu finalizează conexiunea TCP (nu trimite pachetul ACK).
- `-sV` (*Service Version Detection*): detectează versiunea exactă a serviciilor care rulează pe porturile deschise.

- **-O** (*OS Detection*): analizează pachetele răspuns pentru a determina sistemul de operare al gazdei.
- **-Pn** (*No Ping*): mod în care **nmap** nu verifică în prealabil dacă hostul este activ (skip ping); util în cazul firewall-urilor care blochează ICMP sau porturile implicite de ping.

Rezultatul scanării a relevat următoarele informații:

PORT	STATE	SERVICE	VERSION
135/tcp	open	msrpc	Microsoft Windows RPC
139/tcp	open	netbios-ssn	Microsoft Windows netbios-ssn
445/tcp	open	microsoft-ds?	
5357/tcp	open	http	Microsoft HTTPAPI httpd 2.0 (SSDP/UPnP)

Pe baza semnăturilor de protocol și a modului în care hostul a răspuns la scanare, **nmap** a identificat sistemul de operare ca fiind o versiune a Microsoft Windows 10:

```
Running: Microsoft Windows 10
OS CPE: cpe:/o:microsoft:windows_10
OS details: Microsoft Windows 10 1709 - 21H2
```

Această detecție este posibilă doar dacă sistemul țintă permite traficul necesar. Din acest motiv, pentru a evalua comportamentul rețelei în scenarii realiste, în etapa următoare am simulat situații în care firewall-ul este activat, iar conexiunea este configurată ca publică sau privată, pe sistemul de operare Windows.

Comportamentul scanării nmap în funcție de configurarea sistemului Windows

Pentru a evidenția impactul setărilor de securitate asupra detecției rețelei, am testat comportamentul dispozitivului Windows (192.168.30.4) în următoarele trei scenarii:

1. Conexiune **privată**, **fără firewall**;
2. Conexiune **privată**, **cu firewall activ**;
3. Conexiune **publică**, **cu firewall activ**.

1. Conexiune privată, firewall dezactivat

Scanarea a fost realizată cu comanda:

```
nmap -sS -sV -O -Pn 192.168.30.4 -oN scan_victim.txt
```

Această comandă combină mai multe opțiuni:

- **-sS**: scanare TCP SYN, discretă și rapidă;
- **-sV**: detectare versiune servicii;

- -0: identificare sistem de operare;
- -Pn: dezactivează ping-ul inițial;
- -oN scan_victim.txt: salvează rezultatul în fișier text.

Rezultatul complet al scanării este prezentat în Anexa 4.

După cum se observă în figura corespunzătoare din Anexa 4, scanarea are succes deplin: sunt identificate patru porturi deschise (135, 139, 445, 5357), serviciile corespunzătoare, și sistemul de operare – Windows 10, versiuni 1709–21H2. Răspunsurile detaliate sunt posibile datorită lipsei filtrării la nivel de firewall.

2. Conexiune privată, firewall activ

Scanarea a fost efectuată folosind aceeași comandă:

```
nmap -sS -sV -0 -Pn 192.168.30.4 -oN scan_firewall_private.txt
```

Rezultatul complet al scanării este inclus în Anexa 4.

După cum se observă în figura corespunzătoare din Anexa 4, firewall-ul activ filtrează majoritatea porturilor, ceea ce determină **nmap** să marcheze 999 de porturi ca fiind **filtered**. Singurul port detectat ca deschis este 5357/tcp, asociat cu serviciul HTTPAPI. Detecția sistemului de operare devine imprecisă, afișând doar o estimare aproximativă, fără identificare exactă. Acest comportament reflectă eficiența firewall-ului în blocarea scanărilor de rețea.

3. Conexiune publică, firewall activ

Comanda utilizată a fost identică:

```
nmap -sS -sV -0 -Pn 192.168.30.4 -oN scan_firewall_public.txt
```

Rezultatul complet al scanării este inclus în Anexa 4.

În figura corespunzătoare din Anexa 4 observăm un scenariu în care toate cele 1000 de porturi TCP scanate se află în stare **filtered** (fără răspuns). Acest comportament este tipic pentru o configurație strictă de firewall combinată cu o conexiune setată ca fiind „Public”. Detecția sistemului de operare este complet imprecisă, și **nmap** nu reușește să distingă detalii relevante despre gazdă. Este cel mai restrictiv scenariu testat.

5.5.2 Simularea atacului ARP Spoofing

ARP Spoofing este un atac de nivel 2, care are ca scop deturnarea traficului dintre două gazde din rețea, prin trimiterea de pachete ARP false. Pentru acest scenariu, s-a utilizat mașina Kali Linux. Victima este stația Windows 10 cu adresa IP 192.168.30.4 identificată anterior.

Pentru redirecționarea traficului, s-au folosit două comenzi **arpspoof**, una pentru a impersona gateway-ul în fața victimei, și una invers:

Exemplele vizuale de transmitere a pachetelor ARP spoofing către victimă și gateway, precum și monitorizarea pachetelor ARP falsificate capturate cu `tcpdump`, sunt prezentate în Anexa 2.

Atacul a avut ca efect modificarea tabelului ARP ale victimei și gateway-ului, redirecționând tot traficul prin dispozitivul atacator. Această tehnică este frecvent utilizată ca prim pas în atacuri de tip *Man-in-the-Middle* (MitM), permițând interceptarea, modificarea sau blocarea comunicațiilor în rețele locale.

5.5.3 Simularea atacului DHCP Starvation

Atacul *DHCP Starvation* are ca scop epuizarea tuturor adreselor IP disponibile într-un pool DHCP sau pe un server DHCP, prin trimiterea unui număr mare de cereri (DHCP Discover) cu adrese MAC false. În acest mod, serverul DHCP alocă adrese IP fictive atacatorului, blocând distribuirea acestora către stațiile legitime din rețea.

Testul a fost realizat din mașina Kali Linux aflată în SALA-003, conectată în VLAN-ul corespunzător. Pentru generarea atacului, s-a utilizat aplicația Yersinia în interfața grafică.

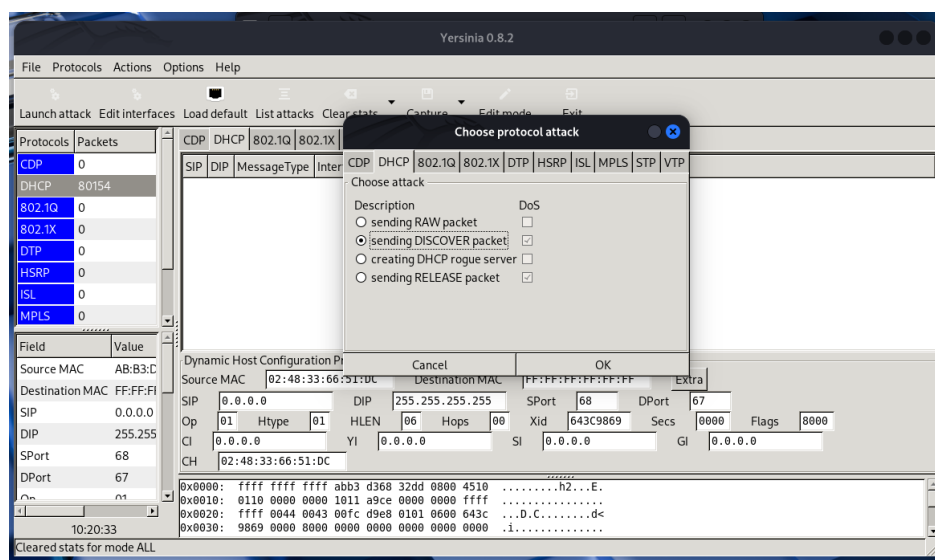


Figura 5.6: Selectarea atacului DHCP Discover din interfața grafică Yersinia

După lansarea atacului, rețeaua a fost monitorizată cu **Wireshark** pentru a observa fluxul de pachete DHCP. Se pot observa multiple mesaje **DHCP Discover** și **DHCP Offer** generate la intervale foarte scurte de timp, toate având sursa 0.0.0.0 și destinația broadcast.

În urma atacului, serverul DHCP de pe **CORE-ROUTER** a alocat toate adresele IP care mai erau disponibile. Acest aspect a fost verificat pe router-ul **CORE-ROUTER** care este și server-ul de DHCP pentru întreaga rețea folosind comanda:

```
show ip dhcp binding
```

Imaginile corespunzătoare acestor verificări sunt prezentate în Anexa 3.

În urma epuizării pool-ului DHCP, orice nou dispozitiv conectat în rețea nu va mai primi o adresă IP, rezultând în imposibilitatea conectării la infrastructura locală sau

la Internet. Acest atac evidențiază vulnerabilitatea serverelor DHCP nesecurizate și impactul semnificativ asupra funcționării unei rețele.

5.5.4 Simularea atacului ICMP Flood

Atacul de tip *ICMP Flood* constă în trimiterea unui număr mare de cereri ICMP Echo Request (“ping”) către o gazdă țintă, cu scopul de a-i satura resursele de rețea și procesare. Acest tip de atac poate afecta performanța sistemului vizat, mai ales dacă firewall-ul sau mecanismele de limitare a traficului nu sunt configurate corespunzător.

În cadrul acestui scenariu, atacul a fost realizat folosind tot mașină Kali Linux din VLAN-ul aferent SALA-003, iar gazda țintă a fost stația Windows 10 cu adresa IP 192.168.30.4 prezentă tot în același VLAN.

Comanda folosită pentru declanșarea atacului a fost:

```
sudo hping3 -1 --flood 192.168.30.4
```

Parametrii comenzii sunt:

- `-1` – specifică utilizarea protocolului ICMP;
- `--flood` – trimite pachetele continuu, fără a aștepta răspunsuri;
- `192.168.30.4` – adresa IP a victimei.

Pentru a verifica efectele atacului, s-a utilizat **Wireshark** pe conexiunea către stația Windows, observându-se o cantitate mare de pachete ICMP primite. Captura Wireshark realizată în timpul atacului este prezentată în Anexa 5.

Atacul ICMP Flood a fost simulat cu succes folosind **hping3**, demonstrând potențialul de perturbare a serviciilor unei rețele atunci când lipsesc măsurile de protecție esențiale.

5.6 Măsuri de securitate

În această secțiune sunt prezentate măsurile de securitate implementate pe switch-ul SW-003 din sala 003 pentru a proteja rețeaua împotriva atacurilor simulate anterior: ARP Spoofing, DHCP Starvation și ICMP Flood. Aceste măsuri au fost configurate utilizând funcționalități specifice Cisco IOS versiunea 15.2 și testate pentru a asigura protecția fără a compromite funcționalitățile de bază ale rețelei.

5.6.1 Măsuri de protecție împotriva atacului ARP Spoofing

Pentru prevenirea deturnării traficului prin trimiterea de pachete ARP false, s-a implementat o combinație de mecanisme de securitate pe switch-ul SW-003, responsabil de conexiunile stațiilor din SALA-003. Configurația aplicată este:

- **DHCP Snooping** – pentru validarea sursei adreselor IP;

- **Dynamic ARP Inspection (DAI)** – pentru filtrarea pachetelor ARP falsificate;
- **IP Source Guard** – pentru blocarea pachetelor care nu corespund unei legături IP/MAC/port valide.

Configurare realizată pe switch-ul SW-003:

```
ip dhcp snooping
ip dhcp snooping vlan 30
ip arp inspection vlan 30
interface Gi0/0
    ip dhcp snooping trust
interface range Gi0/1 - 0/15
    ip verify source
    ip arp inspection limit rate 15
```

Portul Gi0/0 este configurat ca trusted, deoarece face legătura cu switch-ul de distribuție SW-1-15. Celelalte porturi sunt porturi access pentru stațiile din sală. După activare, atacurile de tip **arpspoof** nu mai au efect, iar pachetele ARP false sunt blocate de switch. Testarea a confirmat faptul că traficul nu mai este redirecționat către Kali Linux.

5.6.2 Măsuri de protecție împotriva atacului DHCP Starvation

Protecția împotriva atacului *DHCP Starvation* presupune blocarea cererilor masive de adrese IP provenite de la un atacator care folosește adrese MAC aleatorii. Aceste cereri au ca scop epuizarea completă a pool-ului de adrese IP al serverului DHCP, ceea ce duce la imposibilitatea conectării dispozitivelor legitime.

Mecanismul de bază utilizat pentru protecție este **DHCP Snooping**, care, pe lângă rolul său de validare a sursei adreselor IP, permite adăugarea unor restricții suplimentare specifice acestui tip de atac, cum ar fi **limitarea ratei de cereri DHCP** pe porturile de acces.

Configurația aplicată pe switch-ul SW-003, aflat în SALA-003, este următoarea:

```
ip dhcp snooping
ip dhcp snooping vlan 30

interface GigabitEthernet0/0
    description Uplink to SW-1-15
    ip dhcp snooping trust

interface range Gi0/1 - 0/15
    description Porturi pentru sta\c tii din sal\u a
    ip dhcp snooping limit rate 5
```

Prin aplicarea comenzii `ip dhcp snooping limit rate 5`, switch-ul permite maximum 5 cereri DHCP pe secundă pentru fiecare port, ceea ce este suficient pentru o funcționare normală, dar blochează eficient flood-ul generat de un atacator.

Monitorizarea rețelei în timpul atacului *DHCP Starvation* cu DHCP Snooping activ este prezentată în Anexa 6.

După aplicarea măsurilor, atacul generat din Kali Linux folosind aplicația Yersinia nu a mai produs epuizarea adreselor IP, fapt confirmat prin comanda `show ip dhcp binding` executată pe CORE-ROUTER, care a indicat menținerea adreselor disponibile pentru clienți legitimi.

5.6.3 Măsuri de protecție împotriva atacului ICMP Flood

Atacul de tip *ICMP Flood* constă în trimiterea unui număr mare de pachete ICMP Echo Request (ping) pentru a satura resursele de procesare ale unei gazde sau a echipamentelor de rețea intermediare. Pentru prevenirea acestui tip de atac, s-a implementat o politică de **limitare a traficului ICMP** pe switch-ul de acces SW-003, folosind un ACL aplicat pe porturile fizice către clienți, cu următoarea configurație:

```
access-list 101 permit icmp any any echo
access-list 101 permit icmp any any echo-reply
access-list 101 deny icmp any any
access-list 101 permit ip any any

interface range Gi0/1 - 0/15
rate-limit input access-group 101 2000000 8000 8000 \
conform-action transmit exceed-action drop
```

Această configurație permite comunicarea ICMP de bază (ping), dar blochează trimiterea masivă de pachete peste 2 Mbps, comportament tipic pentru atacuri *flood*. După aplicarea măsurii de securitate, traficul ICMP excesiv a fost blocat la nivel de port, iar stația victimă Windows 10 nu a mai primit pachetele injectate de Kali Linux.

6 Rezultate și discuții

Lucrarea de față a avut ca obiectiv proiectarea, implementarea și testarea unei infrastructuri de rețea funcționale, scalabile și securizate. Concret, rețeaua simulată reprezintă o variantă realistă a infrastructurii de comunicație de la parterul UVT, fiind organizată în mai multe săli (ex. SALA-003, SALA-031 etc.) conectate la o rețea centralizată CORE prin intermediul unor conexiuni GNS3 de tip `cloud node` și tuneluri UDP.

Proiectul a inclus atât partea de proiectare logică (VLAN-uri, segmentare, adrese IP, DHCP, rutare statică și dinamică), cât și implementarea fizică simulată în GNS3, utilizând echipamente Cisco IOSv și IOSvL2. Fiecare sală funcționează ca un domeniu de broadcast separat, iar rețeaua centrală (CORE) joacă rolul de centru de control, găzduind routerul principal (CORE-ROUTER), switch-ul de distribuție (CORE-SW) și un număr variabil de switch-uri de acces intermediare. Funcționalitatea generală a rețelei a fost confirmată prin testarea conectivității end-to-end între stații din VLAN-uri diferite, verificarea alocării adreselor IP prin DHCP, propagarea rutelor OSPF între subrețele și asigurarea accesului la Internet pentru toate stațiile legitimate.

Testarea securității rețelei a constituit o componentă esențială a proiectului. Au fost simulate trei atacuri cibernetice frecvente în medii reale, toate lansate dintr-o mașină virtuală Kali Linux conectată în SALA-003. Astfel, au fost realizate scanări de rețea cu `nmap` pentru analiza vizibilității dispozitivelor, un atac de tip ARP Spoofing destinat interceptării traficului dintre stațiile legitime, un atac DHCP Starvation pentru epuizarea adreselor IP disponibile în pool-ul de pe serverul DHCP și un atac ICMP Flood care a vizat blocarea stației victimă prin trimiterea excesivă de pachete ICMP.

Fiecare atac a fost inițial executat cu succes, demonstrând vulnerabilitățile rețelei într-un context nesecurizat. Ulterior, au fost implementate măsuri specifice de protecție pe echipamentele implicate. DHCP Snooping a fost activat pe toate switch-urile de pe traseul client-server DHCP, iar mecanisme precum Dynamic ARP Inspection și IP Source Guard au fost configurate pe switch-ul SW-003, pentru a preveni falsificarea pachetelor ARP. De asemenea, rate-limiting pentru ICMP a fost aplicat pe porturile destinate stațiilor finale, iar etichetarea corectă a interfețelor `trusted` pe legăturile trunk a permis funcționarea controlată a traficului DHCP. După implementarea acestor configurări, pachetele ARP false nu mai erau propagate în rețea, serverul DHCP nu mai răspundea cererilor neautorizate, iar stațiile victime nu mai recepționau pachetele ICMP injectate masiv. Configurația ACL și valorile de rate-limit alese au reușit să asigure un echilibru optim între securitate și funcționalitate.

Pe parcursul testării, una dintre dificultățile majore a fost reprezentată de configurarea corectă a DHCP Snooping. Într-un stadiu intermediar, stațiile nu mai primeau adrese IP, din cauza lipsei comenzii `ip dhcp snooping trust` pe anumite interfețe

trunk. Această problemă a fost depășită printr-o analiză detaliată a fluxului de pachete în infrastructură, folosind instrumente precum **Wireshark** și comenzi de tip `debug ip dhcp server packet` pe router.

Un alt obstacol semnificativ a fost consumul ridicat de resurse al simulatorului GNS3, în special în etapa de integrare a tuturor componentelor într-un ansamblu funcțional unic. Utilizarea mai multor instanțe de IOSv și IOSvL2 pentru fiecare sală de calcul, împreună cu proiectul central **CORE** și mașina virtuală **Kali Linux**, a dus la o solicitare intensă a resurselor hardware ale sistemului gazdă, ceea ce a provocat uneori blocaje sau instabilitate în rularea simulatorului. Aceste dificultăți au fost gestionate prin reporniri parțiale ale proiectelor sau realocări de memorie, fără a afecta validitatea testelor efectuate.

În final, rețeaua obținută este complet funcțională, adaptabilă, segmentată corespunzător și protejată împotriva unor atacuri esențiale. Configurația propusă poate servi drept model de referință pentru proiectarea unei infrastructuri reale de tip campus universitar, la scară mică spre medie.

7 Concluzii și direcții viitoare

Lucrarea de față a urmărit proiectarea, implementarea și testarea unei rețele informatice adaptate pentru un mediu educațional. Folosind platformele GNS3 și Cisco Packet Tracer, a fost realizată o infrastructură virtuală complet funcțională, care a permis simularea unor scenarii reale de funcționare a rețelei, precum și testarea unor aspecte esențiale legate de securitate. În cadrul proiectului au fost simulate o serie de atacuri cibernetice uzuale, iar în urma acestora au fost implementate măsuri de protecție pentru consolidarea securității rețelei. Acest demers a permis o mai bună înțelegere a modului în care pot fi identificate vulnerabilitățile și aplicate soluții pentru prevenirea sau limitarea efectelor unor eventuale atacuri.

Rezultatele obținute confirmă realizarea unei infrastructuri stabile și sigure, care poate constitui un model de bază pentru implementări similare în mediul educațional. Îmbinarea aspectelor teoretice cu aplicarea practică a contribuit la dezvoltarea unor competențe esențiale în domeniul rețelisticii și securității informatice.

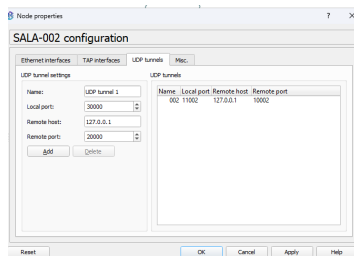
Posibile direcții viitoare ar putea include extinderea lucrării în cadrul unei disertații de master, prin dezvoltarea întregii infrastructuri a campusului universitar, cu integrarea mai multor clădiri și servere dedicate. De asemenea, se pot explora soluții avansate de detecție și prevenire a intruziunilor, prin integrarea unui sistem IDS/IPS și a unui centru de operațiuni de securitate (SOC) cu analiză centralizată a logurilor și alertelor. Alte dezvoltări relevante includ configurarea unui mediu Active Directory pentru gestionarea centralizată a utilizatorilor, testarea compatibilității infrastructurii cu IPv6, precum și adăugarea unui mediu wireless complet funcțional cu access point-uri simulate. Nu în ultimul rând, integrarea de metode de automatizare a configurării și administrării rețelei ar permite trecerea către o arhitectură modernă, eficientă și ușor de scalat. Totodată, simularea unor atacuri cibernetice mai complexe, precum cele de tip MITM avansat, DNS Tunneling, DoS distribuit sau atacuri asupra protocoalelor de autentificare, ar putea contribui la o înțelegere și mai aprofundată a vulnerabilităților reale din rețelele moderne. Astfel, această lucrare constituie o bază solidă atât din punct de vedere tehnic, cât și educațional, oferind numeroase posibilități de dezvoltare ulterioară în domeniul rețelisticii și securității informatice.

Bibliografie

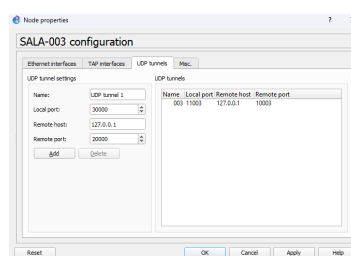
- [1] Mohammed Rabiâ Djouama. Full design and configuration of enterprise campus network using cisco devices and gns3. <http://archives.univ-biskra.dz/handle/123456789/11074>, June 2017. Master's thesis, Faculté des Sciences et de la Technologie, Université Mohamed Khider Biskra. Accessed April 2025.
- [2] Osman Goni. Implementation of local area network (lan) and build a secure lan system for atomic energy research establishment (aere). *International Journal of Modern Communication Technologies and Research*, 8(2):1–8, April 2022. Accessed April 2025.
- [3] Jhansi Bharathi Madavarapu and V.Šujatha Ravi. Secure virtual local area network design and implementation for electronic data interchange. *International Journal of Advanced Computer Science and Applications*, 14(7):1–7, July 2023.
- [4] J. Moy. Ospf version 2, rfc 2328. <https://datatracker.ietf.org/doc/html/rfc2328>, 1998.
- [5] Chigozie Chidimma Ngene, Ifeanyi C. Ezeugbor, and I.Ķ. Okolo. Design and deployment of local area network (lan). *International Journal of Research Publication and Reviews*, 3(3):1609–1619, March 2022. Accessed April 2025.
- [6] Wendell Odom. *CCNA 200-301 Official Cert Guide, Volume 1*. Cisco Press, 2020.
- [7] Andrew S. Tanenbaum and David J. Wetherall. *Computer Networks*. Pearson, 5 edition, 2011.

Anexe

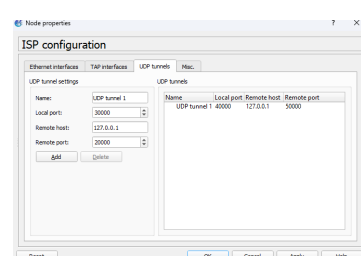
Anexa 1 - Exemple de configurare a tunelurilor UDP pe alte noduri Cloud



SALA-002

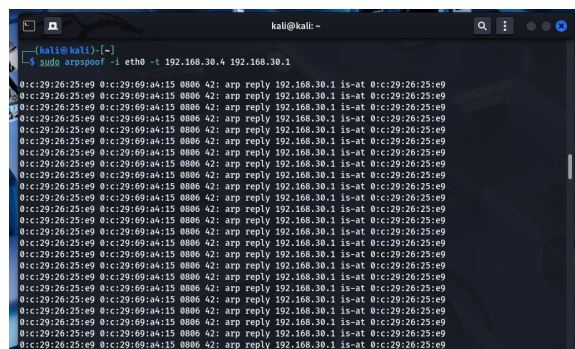


SALA-003

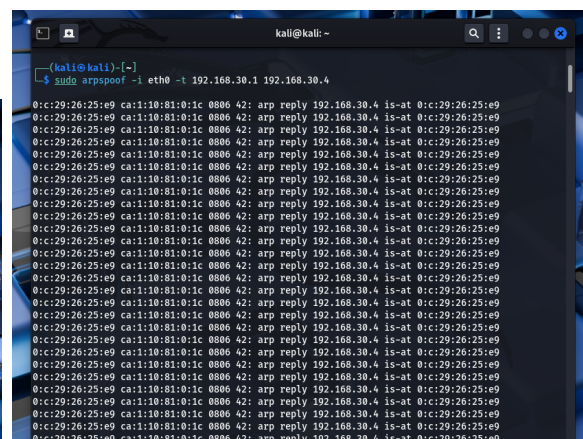


ISP

Anexa 2 - Lansare atac ARP Spoofing și monitorizare pachete ARP falsificate cu tcpdump



Trimitere pachete ARP spoofing către victimă

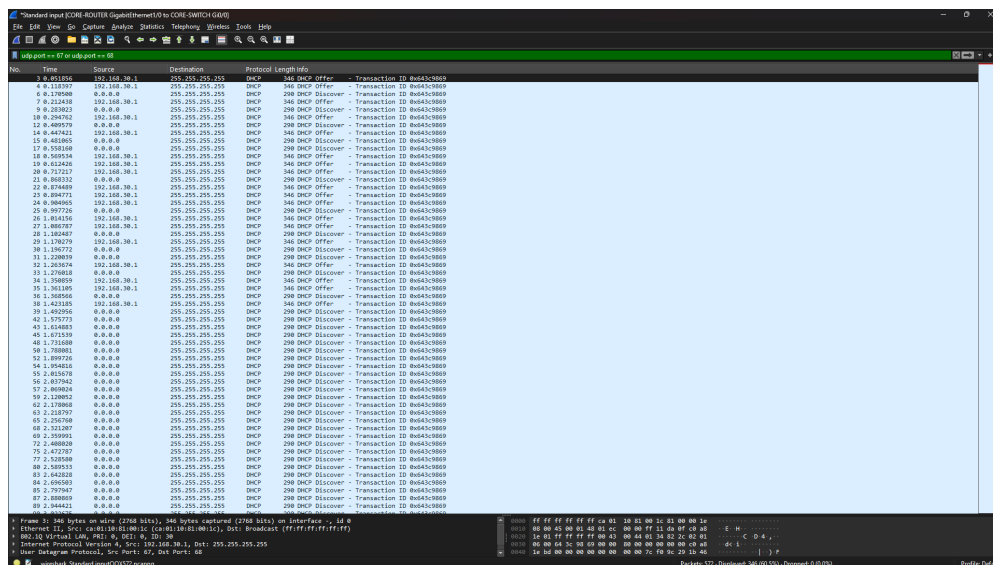


Trimitere pachete ARP spoofing către gateway

```
kali@kali: ~  
(kali@kali)~  
$ sudo tcpdump -i eth0 -n arp  
[sudo] password for kali:  
tcpdump: verbose output suppressed, use -v[v]... for full protocol decode  
listening on eth0, link-type EN10MB (Ethernet), snapshot length 262144 bytes  
06:12:09.944728 ARP, Reply 192.168.30.1 is-at 00:0c:29:26:25:e9, length 28  
06:12:10.008386 ARP, Reply 192.168.30.4 is-at 00:0c:29:26:25:e9, length 28  
06:12:11.945310 ARP, Reply 192.168.30.1 is-at 00:0c:29:26:25:e9, length 28  
06:12:12.010442 ARP, Reply 192.168.30.4 is-at 00:0c:29:26:25:e9, length 28  
06:12:13.946029 ARP, Reply 192.168.30.1 is-at 00:0c:29:26:25:e9, length 28  
06:12:14.011404 ARP, Reply 192.168.30.4 is-at 00:0c:29:26:25:e9, length 28  
06:12:15.946851 ARP, Reply 192.168.30.1 is-at 00:0c:29:26:25:e9, length 28  
06:12:16.012852 ARP, Reply 192.168.30.4 is-at 00:0c:29:26:25:e9, length 28  
06:12:17.678784 ARP, Request who-has 192.168.30.1 tell 192.168.30.135, length 28  
06:12:17.699544 ARP, Reply 192.168.30.1 is-at ca:01:10:81:00:1c, length 46  
06:12:17.947951 ARP, Reply 192.168.30.1 is-at 00:0c:29:26:25:e9, length 28  
06:12:18.013891 ARP, Reply 192.168.30.4 is-at 00:0c:29:26:25:e9, length 28  
06:12:19.948603 ARP, Reply 192.168.30.1 is-at 00:0c:29:26:25:e9, length 28  
06:12:20.014746 ARP, Reply 192.168.30.4 is-at 00:0c:29:26:25:e9, length 28  
06:12:21.949191 ARP, Reply 192.168.30.1 is-at 00:0c:29:26:25:e9, length 28  
06:12:22.015030 ARP, Reply 192.168.30.4 is-at 00:0c:29:26:25:e9, length 28  
06:12:23.949589 ARP, Reply 192.168.30.1 is-at 00:0c:29:26:25:e9, length 28  
06:12:24.016461 ARP, Reply 192.168.30.4 is-at 00:0c:29:26:25:e9, length 28
```

Interceptare pachete utilizând tcpdump

Anexa 3 - Simularea atacului DHCP Starvation



Captură Wireshark – flux de pachete DHCP Discover și DHCP Offer în timpul atacului

CORE-ROUTER(config)#do show ip dhcp binding				
Bindings from all pools not associated with VRF:				
IP address	Client-ID/ Hardware address/ User name	Lease expiration	Type	
192.168.30.2	54e3.411a.e696	Jun 09 2025 02:46 PM	Automatic	
192.168.30.3	4264.7e4a.efef	Jun 09 2025 02:46 PM	Automatic	
192.168.30.4	f7ee.0512.50b5	Jun 09 2025 02:46 PM	Automatic	
192.168.30.5	2666.7e4d.2488	Jun 09 2025 02:46 PM	Automatic	
192.168.30.6	78bf.f907.ec7b	Jun 09 2025 02:46 PM	Automatic	
192.168.30.7	3fe0.5e57.452b	Jun 09 2025 02:46 PM	Automatic	
192.168.30.8	b584.9259.421f	Jun 09 2025 02:46 PM	Automatic	
192.168.30.9	eba3.f47c.c3b8	Jun 09 2025 02:46 PM	Automatic	
192.168.30.10	9f2a.ed50.76e6	Jun 09 2025 02:46 PM	Automatic	
192.168.30.11	3d72.3c14.a15d	Jun 09 2025 02:46 PM	Automatic	
192.168.30.12	c939.827a.e9c3	Jun 09 2025 02:46 PM	Automatic	
192.168.30.13	c706.2407.faf7	Jun 09 2025 02:46 PM	Automatic	
192.168.30.14	102f.f25f.989d	Jun 09 2025 02:46 PM	Automatic	
192.168.30.15	a180.c579.8fab	Jun 09 2025 02:46 PM	Automatic	
192.168.30.16	eb54.5b5e.47a0	Jun 09 2025 02:46 PM	Automatic	
192.168.30.17	1912.7b29.303d	Jun 09 2025 02:46 PM	Automatic	
192.168.30.18	933a.576f.02e0	Jun 09 2025 02:46 PM	Automatic	
192.168.30.19	146d.790b.2ad6	Jun 09 2025 02:46 PM	Automatic	
192.168.30.20	ba81.e53b.81aa	Jun 09 2025 02:46 PM	Automatic	
192.168.30.21	ad92.894b.600c	Jun 09 2025 02:46 PM	Automatic	
192.168.30.22	9b49.f52a.0c47	Jun 09 2025 02:46 PM	Automatic	
192.168.30.23	0e27.df40.3f21	Jun 09 2025 02:46 PM	Automatic	
192.168.30.24	803a.2675.6099	Jun 09 2025 02:47 PM	Automatic	
192.168.30.25	8b6f.c96c.bd00	Jun 09 2025 02:47 PM	Automatic	
192.168.30.26	3eab.5028.c53b	Jun 09 2025 02:47 PM	Automatic	
192.168.30.27	0a61.631f.2683	Jun 09 2025 02:47 PM	Automatic	
192.168.30.28	918f.4169.b232	Jun 09 2025 02:47 PM	Automatic	
192.168.30.29	be79.7875.da7c	Jun 09 2025 02:47 PM	Automatic	
192.168.30.30	fc46.8208.0599	Jun 09 2025 02:47 PM	Automatic	
192.168.30.31	4b97.0a37.2e80	Jun 09 2025 02:47 PM	Automatic	
192.168.30.32	7a5f.aa0a.a325	Jun 09 2025 02:47 PM	Automatic	
192.168.30.33	f34e.e433.aa7d	Jun 09 2025 02:47 PM	Automatic	
192.168.30.34	1c66.c951.4720	Jun 09 2025 02:47 PM	Automatic	
192.168.30.220	2e98.fd7d.1650	Jun 09 2025 02:46 PM	Automatic	
192.168.30.221	0835.8d7a.7246	Jun 09 2025 02:46 PM	Automatic	
192.168.30.222	2fc2.271c.f70c	Jun 09 2025 02:46 PM	Automatic	
192.168.30.223	f439.b400.2942	Jun 09 2025 02:46 PM	Automatic	
192.168.30.224	a95d.e834.b2b7	Jun 09 2025 02:46 PM	Automatic	
192.168.30.225	e433.dd3b.9f62	Jun 09 2025 02:46 PM	Automatic	
192.168.30.226	2901.146c.e107	Jun 09 2025 02:46 PM	Automatic	
192.168.30.227	c2b3.2a03.a8e9	Jun 09 2025 02:46 PM	Automatic	
192.168.30.228	9ff6.0223.55d4	Jun 09 2025 02:46 PM	Automatic	
192.168.30.229	a260.ca6f.fb3e	Jun 09 2025 02:46 PM	Automatic	
192.168.30.230	a074.1956.f09f	Jun 09 2025 02:46 PM	Automatic	
192.168.30.231	4392.5123.5c3d	Jun 09 2025 02:46 PM	Automatic	
192.168.30.232	d9d4.2073.43d5	Jun 09 2025 02:46 PM	Automatic	
192.168.30.233	4932.9e32.5cf9	Jun 09 2025 02:46 PM	Automatic	
192.168.30.234	fecb.1306.0cff	Jun 09 2025 02:46 PM	Automatic	
192.168.30.235	0287.a231.9e40	Jun 09 2025 02:46 PM	Automatic	
192.168.30.236	e39c.a066.27cc	Jun 09 2025 02:46 PM	Automatic	
192.168.30.237	eff5.951f.50a9	Jun 09 2025 02:46 PM	Automatic	
192.168.30.238	db3f.243e.cf20	Jun 09 2025 02:46 PM	Automatic	
192.168.30.239	db1f.5216.129a	Jun 09 2025 02:46 PM	Automatic	
192.168.30.240	5b59.334a.a3eb	Jun 09 2025 02:46 PM	Automatic	
192.168.30.241	edb5.4275.b77a	Jun 09 2025 02:46 PM	Automatic	
192.168.30.242	3c20.6579.843d	Jun 09 2025 02:46 PM	Automatic	
192.168.30.243	ac45.d078.13f3	Jun 09 2025 02:46 PM	Automatic	
192.168.30.244	4712.4a03.291e	Jun 09 2025 02:46 PM	Automatic	
192.168.30.245	5eff.8156.a1d7	Jun 09 2025 02:46 PM	Automatic	
192.168.30.246	6afd.f00a.1005	Jun 09 2025 02:46 PM	Automatic	
192.168.30.247	b2e2.d564.1aa8	Jun 09 2025 02:46 PM	Automatic	
192.168.30.248	a2be.861f.3715	Jun 09 2025 02:46 PM	Automatic	
192.168.30.249	bbe3.cd42.8f3f	Jun 09 2025 02:46 PM	Automatic	
192.168.30.250	0e2b.3549.ef66	Jun 09 2025 02:46 PM	Automatic	
192.168.30.251	7bc8.4566.5a0b	Jun 09 2025 02:46 PM	Automatic	
192.168.30.252	74b3.b118.2f68	Jun 09 2025 02:46 PM	Automatic	
192.168.30.253	7e25.243e.2640	Jun 09 2025 02:46 PM	Automatic	
192.168.30.254	b57f.3509.2302	Jun 09 2025 02:46 PM	Automatic	

Alocarea tuturor adreselor disponibile pe CORE-ROUTER

Anexa 4 - Rezultatele scanărilor NMAP

```

kali@kali: ~
(kali@kali)-[~]
$ sudo nmap -sS -sV -O -Pn 192.168.30.4 -oN scan_victim.txt

[sudo] password for kali:
Starting Nmap 7.95 ( https://nmap.org ) at 2025-06-09 06:03 EDT
Nmap scan report for 192.168.30.4
Host is up (0.17s latency).
Not shown: 996 closed tcp ports (reset)
PORT      STATE SERVICE        VERSION
135/tcp    open  msrpc          Microsoft Windows RPC
139/tcp    open  netbios-ssn   Microsoft Windows netbios-ssn
445/tcp    open  microsoft-ds?
5357/tcp   open  http           Microsoft HTTPAPI httpd 2.0 (SSDP/UPnP)
MAC Address: 00:0C:29:69:A4:15 (VMware)
Device type: general purpose
Running: Microsoft Windows 10
OS CPE: cpe:/o:microsoft:windows_10
OS details: Microsoft Windows 10 1709 - 21H2
Network Distance: 1 hop
Service Info: OS: Windows; CPE: cpe:/o:microsoft:windows

OS and Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 17.82 seconds

```

Scanare nmap cu firewall dezactivat și conexiune privată

```
kali@kali: ~  
(kali@kali)-[~]  
$ sudo nmap -sS -sV -O -Pn 192.168.30.4 -oN scan_firewall_private.txt  
Starting Nmap 7.95 ( https://nmap.org ) at 2025-06-09 06:07 EDT  
Nmap scan report for 192.168.30.4  
Host is up (0.026s latency).  
Not shown: 999 filtered tcp ports (no-response)  
PORT      STATE SERVICE VERSION  
5357/tcp  open  http      Microsoft HTTPAPI httpd 2.0 (SSDP/UPnP)  
MAC Address: 00:0C:29:69:A4:15 (VMware)  
Warning: OSScan results may be unreliable because we could not find at least 1 open and 1 closed port  
Device type: general purpose  
Running (JUST GUESSING): Microsoft Windows 10|11|2019 (92%)  
OS CPE: cpe:/o:microsoft:windows_10 cpe:/o:microsoft:windows_11 cpe:/o:microsoft:windows_server_2019  
Aggressive OS guesses: Microsoft Windows 10 1803 (92%), Microsoft Windows 10 1903 - 21H1 (92%), Microsoft Win  
dows 11 (87%), Microsoft Windows 10 1909 (85%), Microsoft Windows 10 1909 - 2004 (85%), Windows Server 2019 (85%), Microsoft Windows 10 1809 (85%)  
No exact OS matches for host (test conditions non-ideal).  
Network Distance: 1 hop  
Service Info: OS: Windows; CPE: cpe:/o:microsoft:windows  
  
OS and Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .  
Nmap done: 1 IP address (1 host up) scanned in 52.86 seconds
```

Scanare nmap cu firewall activ și conexiune privată

```
kali@kali: ~  
(kali@kali)-[~]  
$ sudo nmap -sS -sV -O -Pn 192.168.30.4 -oN scan_firewall_public.txt  
Starting Nmap 7.95 ( https://nmap.org ) at 2025-06-09 06:09 EDT  
Nmap scan report for 192.168.30.4  
Host is up (0.013s latency).  
All 1000 scanned ports on 192.168.30.4 are in ignored states.  
Not shown: 1000 filtered tcp ports (no-response)  
MAC Address: 00:0C:29:69:A4:15 (VMware)  
Too many fingerprints match this host to give specific OS details  
Network Distance: 1 hop  
  
OS and Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .  
Nmap done: 1 IP address (1 host up) scanned in 24.25 seconds
```

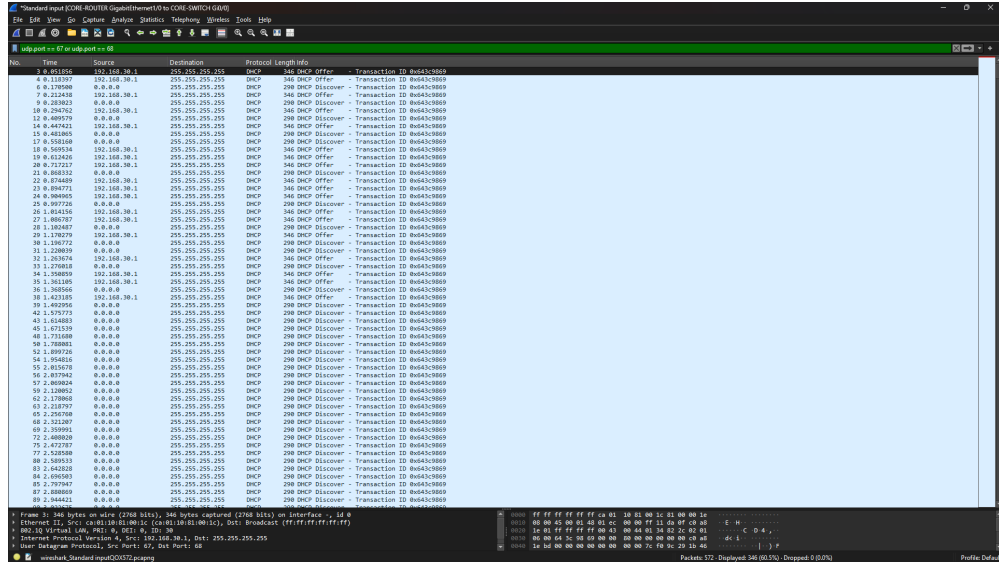
Scanare nmap cu firewall activ și conexiune publică

Anexa 5 - Captură Wireshark ICMP Flood

*Standard input [SW-003 Gi1/6 to Windows10x64-1 Ethernet0]						
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help						
icmp						
No.	Time	Source	Destination	Protocol	Length	Info
2 0.090931	192.168.30.135	192.168.30.4	ICMP	60	Echo (ping) request	id=0xf30d, seq=28435/4975, ttl=64 (no response found!)
3 0.094423	192.168.30.135	192.168.30.4	ICMP	60	Echo (ping) request	id=0xf30d, seq=28691/4976, ttl=64 (no response found!)
4 0.119182	192.168.30.135	192.168.30.4	ICMP	60	Echo (ping) request	id=0xf30d, seq=28947/4977, ttl=64 (no response found!)
5 0.121045	192.168.30.135	192.168.30.4	ICMP	60	Echo (ping) request	id=0xf30d, seq=29203/4978, ttl=64 (no response found!)
6 0.155251	192.168.30.135	192.168.30.4	ICMP	60	Echo (ping) request	id=0xf30d, seq=29459/4979, ttl=64 (no response found!)
7 0.158430	192.168.30.135	192.168.30.4	ICMP	60	Echo (ping) request	id=0xf30d, seq=29715/4980, ttl=64 (no response found!)
10 0.181924	192.168.30.135	192.168.30.4	ICMP	60	Echo (ping) request	id=0xf30d, seq=29971/4981, ttl=64 (no response found!)
11 0.184205	192.168.30.135	192.168.30.4	ICMP	60	Echo (ping) request	id=0xf30d, seq=30227/4982, ttl=64 (no response found!)
12 0.210655	192.168.30.135	192.168.30.4	ICMP	60	Echo (ping) request	id=0xf30d, seq=30483/4983, ttl=64 (no response found!)
13 0.238875	192.168.30.135	192.168.30.4	ICMP	60	Echo (ping) request	id=0xf30d, seq=30739/4984, ttl=64 (no response found!)
16 0.253405	192.168.30.135	192.168.30.4	ICMP	60	Echo (ping) request	id=0xf30d, seq=30995/4985, ttl=64 (no response found!)
17 0.255429	192.168.30.135	192.168.30.4	ICMP	60	Echo (ping) request	id=0xf30d, seq=31251/4986, ttl=64 (no response found!)
20 0.278981	192.168.30.135	192.168.30.4	ICMP	60	Echo (ping) request	id=0xf30d, seq=31507/4987, ttl=64 (no response found!)
21 0.282314	192.168.30.135	192.168.30.4	ICMP	60	Echo (ping) request	id=0xf30d, seq=31763/4988, ttl=64 (no response found!)
26 0.315600	192.168.30.135	192.168.30.4	ICMP	60	Echo (ping) request	id=0xf30d, seq=32019/4989, ttl=64 (no response found!)
27 0.320027	192.168.30.135	192.168.30.4	ICMP	60	Echo (ping) request	id=0xf30d, seq=32275/4990, ttl=64 (no response found!)
28 0.357567	192.168.30.135	192.168.30.4	ICMP	60	Echo (ping) request	id=0xf30d, seq=32531/4991, ttl=64 (no response found!)
29 0.361437	192.168.30.135	192.168.30.4	ICMP	60	Echo (ping) request	id=0xf30d, seq=32787/4992, ttl=64 (no response found!)
34 0.403668	192.168.30.135	192.168.30.4	ICMP	60	Echo (ping) request	id=0xf30d, seq=33043/4993, ttl=64 (no response found!)
35 0.425255	192.168.30.135	192.168.30.4	ICMP	60	Echo (ping) request	id=0xf30d, seq=33299/4994, ttl=64 (no response found!)
37 0.475122	192.168.30.135	192.168.30.4	ICMP	60	Echo (ping) request	id=0xf30d, seq=33555/4995, ttl=64 (no response found!)
38 0.475168	192.168.30.135	192.168.30.4	ICMP	60	Echo (ping) request	id=0xf30d, seq=33811/4996, ttl=64 (no response found!)
40 0.491789	192.168.30.135	192.168.30.4	ICMP	60	Echo (ping) request	id=0xf30d, seq=34067/4997, ttl=64 (no response found!)
41 0.518428	192.168.30.135	192.168.30.4	ICMP	60	Echo (ping) request	id=0xf30d, seq=34323/4998, ttl=64 (no response found!)
49 0.606973	192.168.30.135	192.168.30.4	ICMP	60	Echo (ping) request	id=0xf30d, seq=34579/4999, ttl=64 (no response found!)
50 0.607952	192.168.30.135	192.168.30.4	ICMP	60	Echo (ping) request	id=0xf30d, seq=34835/5000, ttl=64 (no response found!)
51 0.662072	192.168.30.135	192.168.30.4	ICMP	60	Echo (ping) request	id=0xf30d, seq=35091/5001, ttl=64 (no response found!)
52 0.664749	192.168.30.135	192.168.30.4	ICMP	60	Echo (ping) request	id=0xf30d, seq=35347/5002, ttl=64 (no response found!)
55 0.689563	192.168.30.135	192.168.30.4	ICMP	60	Echo (ping) request	id=0xf30d, seq=35603/5003, ttl=64 (no response found!)
56 0.689602	192.168.30.135	192.168.30.4	ICMP	60	Echo (ping) request	id=0xf30d, seq=35859/5004, ttl=64 (no response found!)
63 0.702748	192.168.30.135	192.168.30.4	ICMP	60	Echo (ping) request	id=0xf30d, seq=36115/5005, ttl=64 (no response found!)
64 0.707542	192.168.30.135	192.168.30.4	ICMP	60	Echo (ping) request	id=0xf30d, seq=36371/5006, ttl=64 (no response found!)
69 0.727051	192.168.30.135	192.168.30.4	ICMP	60	Echo (ping) request	id=0xf30d, seq=36627/5007, ttl=64 (no response found!)
70 0.733789	192.168.30.135	192.168.30.4	ICMP	60	Echo (ping) request	id=0xf30d, seq=36883/5008, ttl=64 (no response found!)
73 0.792026	192.168.30.135	192.168.30.4	ICMP	60	Echo (ping) request	id=0xf30d, seq=37139/5009, ttl=64 (no response found!)
74 0.792098	192.168.30.135	192.168.30.4	ICMP	60	Echo (ping) request	id=0xf30d, seq=37395/5010, ttl=64 (no response found!)
75 0.798020	192.168.30.135	192.168.30.4	ICMP	60	Echo (ping) request	id=0xf30d, seq=37651/5011, ttl=64 (no response found!)
76 0.799663	192.168.30.135	192.168.30.4	ICMP	60	Echo (ping) request	id=0xf30d, seq=37907/5012, ttl=64 (no response found!)
77 0.808331	192.168.30.135	192.168.30.4	ICMP	60	Echo (ping) request	id=0xf30d, seq=38163/5013, ttl=64 (no response found!)
78 0.815997	192.168.30.135	192.168.30.4	ICMP	60	Echo (ping) request	id=0xf30d, seq=8980/5155, ttl=64 (no response found!)
81 0.878034	192.168.30.135	192.168.30.4	ICMP	60	Echo (ping) request	id=0xf30d, seq=9236/5156, ttl=64 (no response found!)
82 0.879688	192.168.30.135	192.168.30.4	ICMP	60	Echo (ping) request	id=0xf30d, seq=9492/5157, ttl=64 (no response found!)
85 0.973103	192.168.30.135	192.168.30.4	ICMP	60	Echo (ping) request	id=0xf30d, seq=9748/5158, ttl=64 (no response found!)
86 0.973190	192.168.30.135	192.168.30.4	ICMP	60	Echo (ping) request	id=0xf30d, seq=10004/5159, ttl=64 (no response found!)
87 1.024220	192.168.30.135	192.168.30.4	ICMP	60	Echo (ping) request	id=0xf30d, seq=10260/5160, ttl=64 (no response found!)
88 1.025738	192.168.30.135	192.168.30.4	ICMP	60	Echo (ping) request	id=0xf30d, seq=10516/5161, ttl=64 (no response found!)
89 1.034179	192.168.30.135	192.168.30.4	ICMP	60	Echo (ping) request	id=0xf30d, seq=10772/5162, ttl=64 (no response found!)
90 1.035704	192.168.30.135	192.168.30.4	ICMP	60	Echo (ping) request	id=0xf30d, seq=11028/5163, ttl=64 (no response found!)
91 1.077603	192.168.30.135	192.168.30.4	ICMP	60	Echo (ping) request	id=0xf30d, seq=11284/5164, ttl=64 (no response found!)
92 1.081356	192.168.30.135	192.168.30.4	ICMP	60	Echo (ping) request	id=0xf30d, seq=11540/5165, ttl=64 (no response found!)
99 1.292587	192.168.30.135	192.168.30.4	ICMP	60	Echo (ping) request	id=0xf30d, seq=11796/5166, ttl=64 (no response found!)
100 1.329101	192.168.30.135	192.168.30.4	ICMP	60	Echo (ping) request	id=0xf30d, seq=12052/5167, ttl=64 (no response found!)
108 1.453350	192.168.30.135	192.168.30.4	ICMP	60	Echo (ping) request	id=0xf30d, seq=12308/5168, ttl=64 (no response found!)
110 1.458658	192.168.30.135	192.168.30.4	ICMP	60	Echo (ping) request	id=0xf30d, seq=12564/5169, ttl=64 (no response found!)
113 1.473625	192.168.30.135	192.168.30.4	ICMP	60	Echo (ping) request	id=0xf30d, seq=12820/5170, ttl=64 (no response found!)

Captură Wireshark în timpul atacului ICMP Flood

Anexa 6 - Monitorizare DHCP Starvation protejat cu DHCP Snooping



No.	Time	Source	Destination	Protocol	Length	Info
1	0.001564	192.168.1.1	255.255.255.255	DHCP	346	DHCP Offer - Transaction ID: 8643c089
4	0.111797	192.168.1.1	255.255.255.255	DHCP	346	DHCP Offer - Transaction ID: 8643c089
6	0.170940	0.0.0.0	255.255.255.255	DHCP	290	DHCP Discover - Transaction ID: 8643c089
7	0.221410	192.168.1.1	255.255.255.255	DHCP	346	DHCP Offer - Transaction ID: 8643c089
8	0.282621	0.0.0.0	255.255.255.255	DHCP	290	DHCP Discover - Transaction ID: 8643c089
10	0.295762	192.168.1.1	255.255.255.255	DHCP	346	DHCP Offer - Transaction ID: 8643c089
12	0.400779	0.0.0.0	255.255.255.255	DHCP	290	DHCP Discover - Transaction ID: 8643c089
14	0.447421	192.168.1.1	255.255.255.255	DHCP	346	DHCP Offer - Transaction ID: 8643c089
15	0.453601	0.0.0.0	255.255.255.255	DHCP	290	DHCP Discover - Transaction ID: 8643c089
17	0.558160	0.0.0.0	255.255.255.255	DHCP	290	DHCP Discover - Transaction ID: 8643c089
18	0.597514	192.168.1.1	255.255.255.255	DHCP	346	DHCP Offer - Transaction ID: 8643c089
19	0.612426	192.168.1.1	255.255.255.255	DHCP	346	DHCP Offer - Transaction ID: 8643c089
20	0.721717	192.168.1.1	255.255.255.255	DHCP	346	DHCP Offer - Transaction ID: 8643c089
21	0.804332	0.0.0.0	255.255.255.255	DHCP	290	DHCP Discover - Transaction ID: 8643c089
22	0.874480	192.168.1.1	255.255.255.255	DHCP	346	DHCP Offer - Transaction ID: 8643c089
23	0.894771	192.168.1.1	255.255.255.255	DHCP	346	DHCP Offer - Transaction ID: 8643c089
24	0.908065	192.168.1.1	255.255.255.255	DHCP	346	DHCP Offer - Transaction ID: 8643c089
25	0.907726	0.0.0.0	255.255.255.255	DHCP	290	DHCP Discover - Transaction ID: 8643c089
26	1.014151	192.168.1.1	255.255.255.255	DHCP	346	DHCP Offer - Transaction ID: 8643c089
27	1.000787	192.168.1.1	255.255.255.255	DHCP	346	DHCP Offer - Transaction ID: 8643c089
28	1.101687	0.0.0.0	255.255.255.255	DHCP	290	DHCP Discover - Transaction ID: 8643c089
29	1.170279	192.168.1.1	255.255.255.255	DHCP	346	DHCP Offer - Transaction ID: 8643c089
30	1.100772	0.0.0.0	255.255.255.255	DHCP	290	DHCP Discover - Transaction ID: 8643c089
31	1.208039	0.0.0.0	255.255.255.255	DHCP	290	DHCP Discover - Transaction ID: 8643c089
32	1.201074	192.168.1.1	255.255.255.255	DHCP	346	DHCP Offer - Transaction ID: 8643c089
33	1.270818	0.0.0.0	255.255.255.255	DHCP	290	DHCP Discover - Transaction ID: 8643c089
34	1.350859	192.168.1.1	255.255.255.255	DHCP	346	DHCP Offer - Transaction ID: 8643c089
35	1.361185	192.168.1.1	255.255.255.255	DHCP	346	DHCP Offer - Transaction ID: 8643c089
36	1.368566	0.0.0.0	255.255.255.255	DHCP	290	DHCP Discover - Transaction ID: 8643c089
39	1.402965	0.0.0.0	255.255.255.255	DHCP	290	DHCP Discover - Transaction ID: 8643c089
42	1.575773	0.0.0.0	255.255.255.255	DHCP	290	DHCP Discover - Transaction ID: 8643c089
43	1.610883	0.0.0.0	255.255.255.255	DHCP	290	DHCP Discover - Transaction ID: 8643c089
45	1.671339	0.0.0.0	255.255.255.255	DHCP	290	DHCP Discover - Transaction ID: 8643c089
48	1.771188	0.0.0.0	255.255.255.255	DHCP	290	DHCP Discover - Transaction ID: 8643c089
50	1.78881	0.0.0.0	255.255.255.255	DHCP	290	DHCP Discover - Transaction ID: 8643c089
52	1.897726	0.0.0.0	255.255.255.255	DHCP	290	DHCP Discover - Transaction ID: 8643c089
54	1.954626	0.0.0.0	255.255.255.255	DHCP	290	DHCP Discover - Transaction ID: 8643c089
55	1.915728	0.0.0.0	255.255.255.255	DHCP	290	DHCP Discover - Transaction ID: 8643c089
56	2.077962	0.0.0.0	255.255.255.255	DHCP	290	DHCP Discover - Transaction ID: 8643c089
57	2.005024	0.0.0.0	255.255.255.255	DHCP	290	DHCP Discover - Transaction ID: 8643c089
59	2.139912	0.0.0.0	255.255.255.255	DHCP	290	DHCP Discover - Transaction ID: 8643c089
62	2.178868	0.0.0.0	255.255.255.255	DHCP	290	DHCP Discover - Transaction ID: 8643c089
63	2.212797	0.0.0.0	255.255.255.255	DHCP	290	DHCP Discover - Transaction ID: 8643c089
65	2.294760	0.0.0.0	255.255.255.255	DHCP	290	DHCP Discover - Transaction ID: 8643c089
68	2.321307	0.0.0.0	255.255.255.255	DHCP	290	DHCP Discover - Transaction ID: 8643c089
69	2.359991	0.0.0.0	255.255.255.255	DHCP	290	DHCP Discover - Transaction ID: 8643c089
72	2.488920	0.0.0.0	255.255.255.255	DHCP	290	DHCP Discover - Transaction ID: 8643c089
75	2.472787	0.0.0.0	255.255.255.255	DHCP	290	DHCP Discover - Transaction ID: 8643c089
77	2.535580	0.0.0.0	255.255.255.255	DHCP	290	DHCP Discover - Transaction ID: 8643c089
80	2.589533	0.0.0.0	255.255.255.255	DHCP	290	DHCP Discover - Transaction ID: 8643c089
83	2.641818	0.0.0.0	255.255.255.255	DHCP	290	DHCP Discover - Transaction ID: 8643c089
84	2.690583	0.0.0.0	255.255.255.255	DHCP	290	DHCP Discover - Transaction ID: 8643c089
85	2.791987	0.0.0.0	255.255.255.255	DHCP	290	DHCP Discover - Transaction ID: 8643c089
87	2.868869	0.0.0.0	255.255.255.255	DHCP	290	DHCP Discover - Transaction ID: 8643c089
89	2.944421	0.0.0.0	255.255.255.255	DHCP	290	DHCP Discover - Transaction ID: 8643c089

Frame 1: 346 bytes on wire (2768 bits), 346 bytes captured (2768 bits) on interface ..., id 0
Ethernet II, Src: C8B11B01001C (ca-b1-1b-01-00-1c), Dst: Broadcast (ff:ff:ff:ff:ff:ff)
Internet Protocol Version 4, Src: 192.168.1.1, Dst: 255.255.255.255
User Datagram Protocol, Src Port: 67, Dst Port: 68

Packet 172: Displayed 346 (80.1%) Dropped 0 (0.0%) Profile: Default

Monitorizarea rețelei în timpul atacului DHCP Starvation cu DHCP Snooping activ