LUCRAREA 3

Urmarirea conexiunilor (Partea 2)

1 Mecanismul de stare a conexiunii (continuare)

1.1 Conexiuni UDP

Conexiunile UDP sunt în sine conexiuni fara stare. Sunt cateva motive pentru care ele sunt considerate astfel, principalul motiv fiind acela că ele nu includ etapele de stabilire a conexi-unii sau închiderea acesteia. Primirea a doua datagrame UDP într-o ordine specifică, nu ne spune nimic despre ordinea in care acestea au fost trimise. Cu toate acestea este posi-bilă setarea anumitor stări ale conexiunii în interiorul kernelului. În continuare vom vedea cum se pot urmari astfel de conexiuni, şi cum arată ele în conntrack.

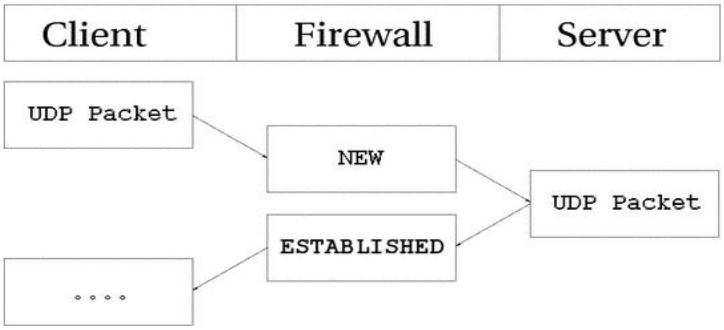

Figura 3.2

După cum se vede, conexiunea este realizată aproape la fel ca şi în cazul conexi-unii TCP. Acesta este ceea ce se poate observa din punctul de vedere al utilizatorului. Intern însă, informaţia conntrack arată puţin diferit. Să privim înainte de toate la o intrare în baza de date conntrack după ce a fost trimis pachetul UDP iniţial:

```
udp      17 20 src=192.168.1.2 dst=192.168.1.5 sport=137 dport=1025 \
     [UNREPLIED] src=192.168.1.5 dst=192.168.1.2 sport=1025 \
     dport=137 use=1
```

După cum se poate observa, prima şi a doua valoare indică faptul că este vorba despre un pachet UDP. Astfel prima valoare este numele protocolului, şi a doua este numarul protocolului. A treia valoare valoare spune timpul de viaţă al acstei stari. Dupa aceasta, urmează sursa şi destinaţia pachetului precum şi portul sursa respective cel destinaţie. În acest moment, flagul UNREPLIED spune că inca nu a sosit un răspuns la acest pachet. Următoarea stare se atinge atunci când soseste un pachet ca raspuns la cel trimis. Valoare de timeout în acest caz este de 30 de secunde.

```
udp      17 170 src=192.168.1.2 dst=192.168.1.5 sport=137 \
     dport=1025 src=192.168.1.5 dst=192.168.1.2 sport=1025 \
     dport=137 use=1
```

În acest moment, serverul a văzut răspunsul la pachetul ce a fost trimis, şi acum conexiunea va fi considerată ESTABLISHED (stabilită). Valoarea de timeout în acest caz este de 160 de secunde. Pentru a de atinge urmatoarea stare, şi anume aceea de ASSURED, trebuie să se mai vehiculeze cateva pachete.

```
udp      17 175 src=192.168.1.5 dst=195.22.79.2 sport=1025 \
     dport=53 src=195.22.79.2 dst=192.168.1.5 sport=53 \
     dport=1025 [ASSURED] use=1
```

În acest moment, conexiunea este asigurată. Dacă conexiunea nu este utilizată timp de 180 de secunde, aceasta expiră.

1.2 Conexiuni ICMP

Pachetele ICMP sunt considerate a fi pachete fara stare, deoarece ele sunt folosite pentru controlul conexiunilor, şi nu pentru a stabili conexiuni. Cu toate acestea există patru tipuri de ICMP care generează pachete de răspuns, şi acestea au două stari dferite. Aeste mesaje ICMP pot lua starile **NEW** şi **ESTABLISHED**. Tipurile de ICMP despre care discutăm sunt Echo request (cerere ecou), Echo reply (raspuns de tip ecou), Timestamp request (cererea amprentei de timp), Timestamp reply (raspunsul la cererea amprentei de timp), Information request (cererea de informaţie), Information reply (raspuns la cererea de informaţie) si în cele din urmă Address mask request (cererea maştii adresei) şi Address mask reply (raspuns la cererea maştii adresei). Cererea de amprentă şi cea de informaţie sunt mai vechi şi nu se mai utilizează, deci acestea pot să fie direct blocate prin firewall. Mesajele de tip ecou, sunt utilizate pentru verificarea conexiunii, asa cum este cazul instructiunii ping, iar cererile de masca, sunt foarte rarutilizate. O astfel de conexiune este reprezentată în figura următoare:

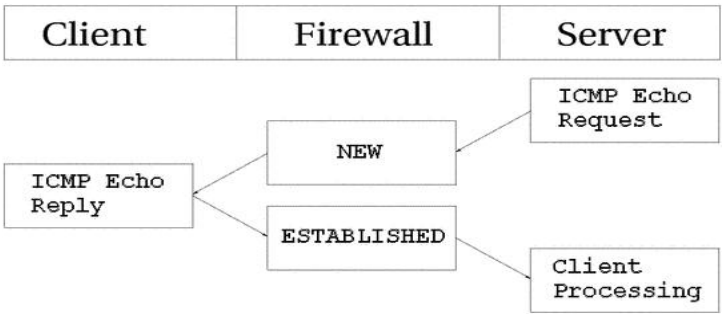

Figura 3.3

După cum se vede din figura de mai sus, hostul timite un pachet echo request calculatorului destinaţie, pachet care este considerat de către firewall ca fiind **NEW.** Calculatorul ţintă raspunde printr-un pachet echo reply pe care firewallul îl consideră ca

fiind in starea **ESTABLISHED.** Când primul pachet echo request este vazut, în ip_conntrack vom avea urmatoarea intrare:

```
icmp     1 25 src=192.168.1.6 dst=192.168.1.10 type=8 code=0 \
    id=33029 [UNREPLIED] src=192.168.1.10 dst=192.168.1.6 \
    type=0 code=0 id=33029 use=1
```

Această intrare arata puțin diferit față de intrările standard TCP si UDP. Avem în această intrare protocolul, timeout-ul, sursa şi destinația. Pe langă acestea mai avem alte trei câmpuri şi anume type, code, şi id. Campul type specifica tipul ICMP-ului, iar campul code specifică codul ICMP. Acestea se pot vedea în tabelul din anexa 2. Câmpul id conține ID-ul pachetului ICMP. Fiecare pachet ICMP primeşte un ID cand acesta este trimis, iar când receptorul primeşte mesajul ICMP, el setează acelaşi ID în pachetul pe care îl trimite, astfel încât emițătorul îl va recunoaşte.
În urmatorul camp recunoaştem flagul UNREPLIED, care ca şi mai înainte indica faptul că urmarim o nexiune care nu a sesizat trafic în una din directii. Pachetul de răspuns este considerat ca fiind în starea ESTABLISHED, dupa cum deja am văzut in secțiunile anterioare. Cu toate acestea, putem ştii că după ICMP reply, nu va mai fi nici un fel de traffic în aceeaşi conexiune. Pentru acest motiv, intrarea în baza de date de urmarire a conexiunii este distrusă din moment ce pachetul de raspuns a traversat structura de filtrare a rețelei.

În cazurile ce urmează vom considera cererea (request-ul) ca fiind **NEW,** iar răspunsul este considerat ca fiind **ESTABLISHED**. Când firewallul vede un pachet cerere (request), îl consideră NEW, iar când hostul trimite un raspuns (reply), acesta este considerat ca fiind **ESTABLISHED**.
Cererea ICMP are un timeout de 30 de secunde. Aceasta valoare poate fi modificata din intrarea /proc/sys/net/ipv4/netfilter/ip_ct_icmp_timeout.

O importantă majora a protocolului ICMP este aceea ca el este utilizat pentru a spune hostului ce s-a intamplat cu anumite conexiuni TCP si UDP, sau cu anumite încercări de conexiune. Datorită acestui fapt, raspunsurile ICMP vor fi foarte des recunoscute ca fiind în stare **RELATED** cu încercările de conexiune originale. Un exemplu simplu ar fi mesajele ICMP Host unreachable (host de neatins) sau ICMP Network unreachable (rețea de neatins). Aceste mesaje ar trebui să le primim tot timpul, dacă încercarea de a ne conecta la un alt host eşuează datorită faptului că rețeaua sau hostul interogat ar putea fi inactive. Astfel ultimul ruter care încearcă să comunice cu entitatea solicitată, va răspunde printr-un mesaj ICMP, spunandu-ne că încercarea de conectare a eşuat. În acest caz, raspunsul ICMP este considerat ca fiind un pachet **RELATED** cu protocolul pe care am încercat să îl utilizăm pentru a apela calculatorul destinație. În figura 3.4 se explică acest lucru.

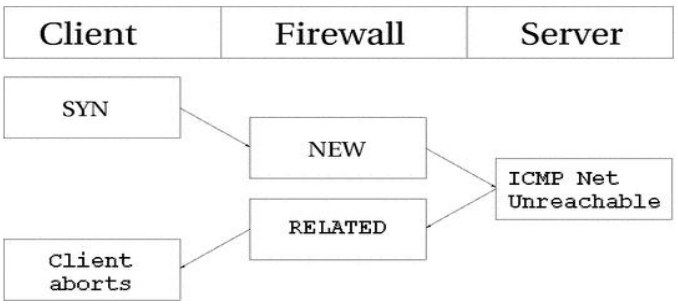

Figura 3.4

S-a trimis un pachet SYN spre o anumită adresă. Aceata este consideră de firewall ca fiind o conexiune **NEW**. Rețeaua la care pachetul dorește să ajungă, este de neatins, asa ca un ruter trimite un mesaj ICMP de eroare (network unreachable). Programul de urmarire a conexiunii recunoaste acest pachet ca fiind în relație de înrudire cu protocolul TCP astfel raspunsul ICMP este corect trimis la client care va abandona încercarea de conectare. În acest timp, firewallul renunța la intrarea de urmărire a acestei conexiuni din moment ce ştie ca acesta a fost un mesaj de eroare.

Aceeaşi comportare o observăm şi in cayul unei conexiuni UDP. Toate mesajele ICMP trimise ca răspuns la conexiunea UDP sunt considerate ca fiind **RELATED** cu protocolul UDP**.** Să considerăm următoarea figură:

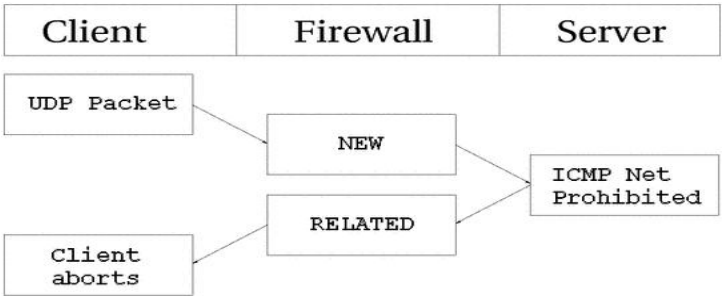

Figura 3.5

De această dată un pachet UDP este trimis către un host. Această conexiune UDP este considerată ca fiind NEW. Se poate ca reteaua să fie inhibată de un anumit firewall, sau de un router pe care pachetul ar trebui să îl străbată. În această situație firewallul nostru primeste ca raspuns la pachetul nostru un mesaj ICMP de eroare. Firewallul ştie că acest mesaj ICMP de eroare este inrudit cu conexiunea UDP deja deschisă, şi îl trimite ca pachet **RELATED** clientului. În acest moment firewalul va renunța la intrarea de urmarire a acestei conexiuni, iar clientul primeste mesajul ICMP şi abandonează conexiunea.

3.1 Protocoale complexe şi urmărirea conexiunii

Unele protocoale sunt mai complexe decât altele deci sunt mai greu de urmarit. Astfele de protocoale sunt: IRC (Internet Relay Chat), FTP (File Transfer Protocol).

Să privim de exemplu protocolul FTP. Inițial acest protocol, deschide o singură conexiune numită sesiune de control FTP. Când emitem comenzi în această sesiune, se deschid alte porturi pentru a transporta restul datelor provenite de la comanda respectivă. Aceste conexiuni pot fi facute în doua moduri: activ şi pasiv. Când o conexiune este facuta activ, clientul FTP trimite serverului un port şi o adresa IP la care acesta să se conecteze. După aceasta, clientul deschide portul, şi serverul se conectează la acel port de pe popriul port 20 (cunoscut ca si port FTP de date) şi trimite date prin acesta. Problema aici este că, firewallul nu va şti despre aceste extraconexiuni, atâta timp cât acestea sunt negociate în pachetul actual al protocolului de date. Din această cauză, firewalul nu va şti daca trebuie să lase serverul să se conecteze la client pe porturile specificate. Soluția la această problemă este să adaugăm din kernel un ajutor special modulului de urmarire a conexiunii, astfel încât în controlul conexiunii să se poată scana datele pentru a găsi sintaxe şi informații speciale. Când se obtine informatia corectă, aceasta se va considera ca fiind **RELATED** cu protocolul FTP, iar serverul va fi capabil să urmărească conexiunea. Aceste lucruri se pot urmari si pe figura urmatoare.

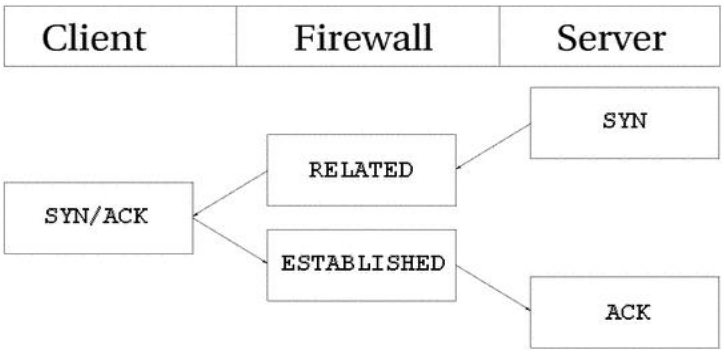

Figura 3.6

Conexiunile FTP pasive funcționează exact invers. Clientul FTP spune serverului că doreste o anume dată, la care serverul raspunde cu adresa IP şi portul la care clientul se poate conecta. Clientul va recepționa această informație, se va conecta de la portul său 20 (FTP+data port) la acel port, si va prelua data pe care a solicitat-o. Dacă avem un server FTP în spatele unui firewall, este necesara utilizarea modulului FTP adițional modulului iptables din kernel, pentru a lasa clienții de pe internet sa se conecteze la serverul nostru. În figura 3.7 se prezintă o conexiune FTP pasivă.

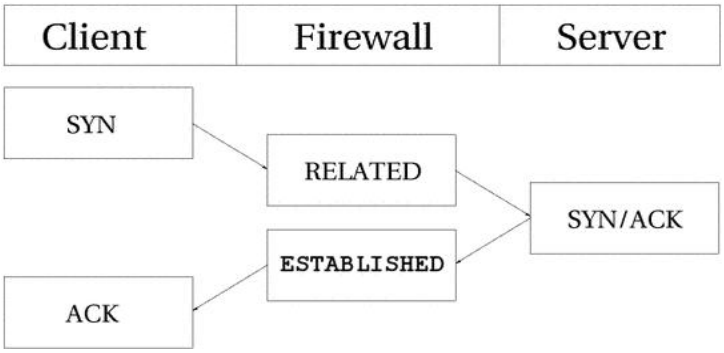

Figura 3.7

Intrebari LUCRAREA 3:

1) Care este diferenta intre starea ESTABLISHED si ASSURED a unei conexiuni?
2) Care este diferenta intre o conexiune FTP active si una pasiva?