Tűzfalak működése és összehasonlításuk

Készítette **Sári Zoltán**
YF5D3E
Óbudai Egyetem
Neumann János Informatikai Kar

1. Bevezetés

A tűzfalak fejlődése a számítógépes hálózatok evolúciójával párhuzamosan, a hálózattechnológia fejlődési irányvonalához igazodik.

A számítógépes hálózatok létrejöttének alapvető célja az erőforrások megosztása, az üzembiztonság fokozása és a költséghatékonyság volt.

Kezdetben katonai, illetve tudományos célokat szolgálva, azonos típusú berendezések között teremtettek kapcsolatot. A működés szabványosítása, az olcsó és üzembiztos technológiák elterjedése eredményeként az 1980-es években széleskörű terjedésnek indultak.

Az eredetileg katonai és szakmai célokra tervezett hálózat gyorsan általános kommunikációs, információtovábbító médiává vált, majd maguktól adódtak az emberi kapcsolatteremtés újabb, sokszor korábban soha nem ismert formái. Napjainkban a mobilkommunikáció elterjedésével, virtuális irodák és távoli hozzáférés alkalmazásával szükségessé vált az, hogy a felhasználók bárhonnan és bármikor hozzáférjenek adataikhoz biztonságos hálózaton keresztül.

Az Internet alapvető szabványait azzal a feltételezéssel tervezték, hogy a hálózatba kizárólag jóindulatú, intézményi felhasználók kapcsolódnak. Az adminisztratív szabályok betartásáról a hálózatba kapcsolódó intézmények maguk gondoskodtak, azok protokoll szintű szabályozása nem alakult ki.

Az Internet szabaddá tételével a helyzet alapvetően megváltozott. Gyakorlatilag bárki kapcsolódhat a hálózathoz, földrajzi, időbeli korlátok nélkül. A felhasználók érdekazonosságában már nem lehet bízni, hiszen konkurens vállalatok és kormányok, valamint ellenőrizetlen magánfelhasználók is a rendszer felhasználói lettek.

A számítógépeken futó alkalmazásokból, az alkalmazások által használt protokollokból és az emberi gondatlanságból kifolyólag a hálózatra kötött számítógépek számos visszaélésre adnak lehetőséget. A kockázat mértéke pedig az ugyanazon hálózatra csatlakozó felhasználók számával arányosan nő. A tűzfalak célja e kockázatok csökkentése.

A tűzfal a hálózat elleni támadásokkal szembeni védelem egyik legfontosabb eleme. Biztonságot nyújt az üzleti adatok számára, megvédi a szellemi tulajdont és a személyes információkat a támadások, káros szoftverekkel szemben. A tűzfalak védelmet kínálnak a belső hálózat számára külső behatolások, online támadások és a helyi hálózatról érkező fertőzések, vírusok, férgek, trójaiak ellen.

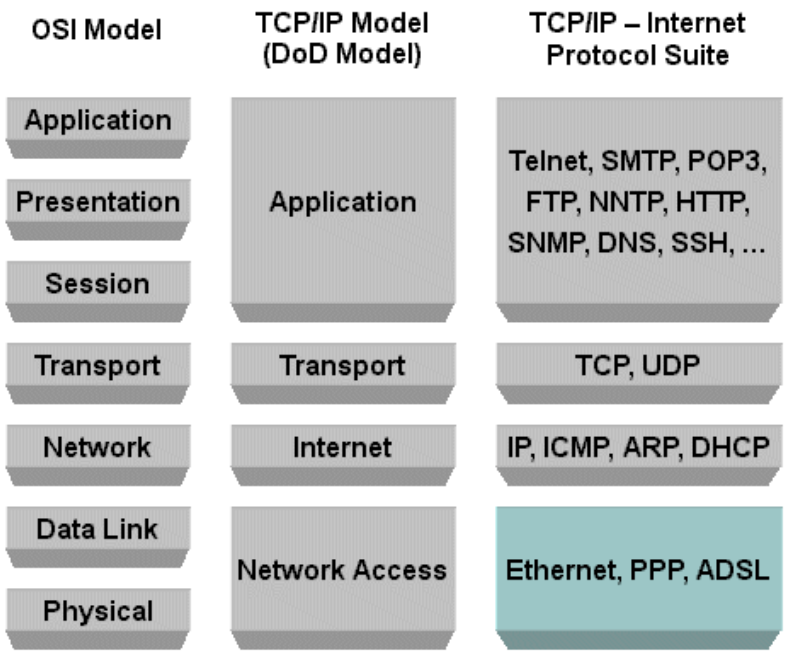
A különböző tűzfal termékek gyakran egymástól teljesen eltérő technológiát használnak, eltérő jellemzőkkel és képességekkel. Dolgozatomban bemutatom, működésük alapján összehasonlítom a napjainkban kínált tűzfal-technológiai megoldások fő irányvonalait.

2. A TCP/IP protokollcsalád

A tűzfalak működése vizsgálatának kiindulópontját a csomagkapcsolt hálózatok működésének áttekintése jelenti. **A számítógép hálózatok elleni támadások végrehajtásához túlnyomórészt a TCP/IP protokollcsalád hálózati rétegében megtalálható ICMP, és az együttműködési rétegben lévő TCP protokollokat használják fel.**

A számítógépes hálózati elemek (hosztok) egymással való kommunikációját, a kapcsolat kialakításához és fenntartásához szükséges feladatokat a négyrétegű TCP/IP protokollcsalád szabályozza.

A TCP/IP protokollcsalád



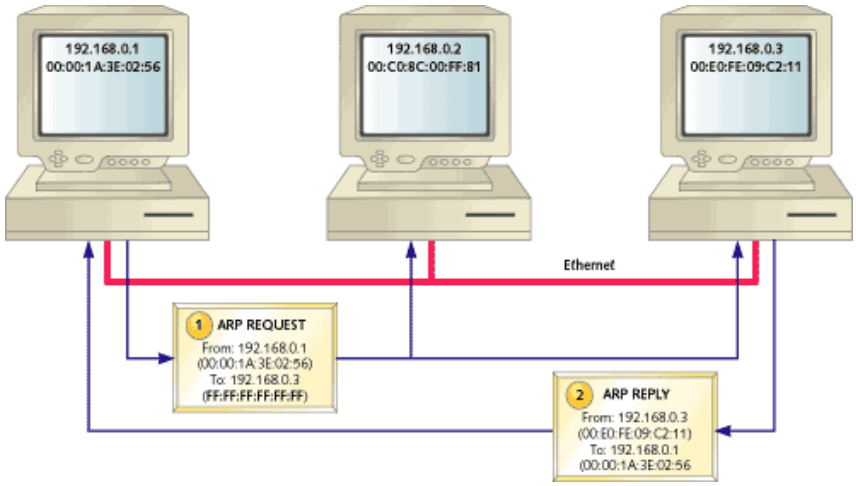
Az Interneten bármelyik hoszt kapcsolatba léphet bármely másik hoszttal. A kapcsolat kialakításának ezért elengedhetetlen feltétele, hogy minden hoszt rendelkezzen legalább egy egységes szerkezetű, hardverfüggetlen egyedi azonosítóval, az **IP címmel**.

A 32 bites IP cím logikailag két részre, az alhálózatot azonosító **netid**-re és az alhálózaton egyedi címmel rendelkező hosztot azonosító **hostid**-ra bomlik.

Az interneten a hosztok egymást IP címeikkel azonosítják, de az adó és a vevő csak a másik **fizika címének** ismeretében tud kapcsolatba lépni. A kommunikációs kapcsolat felépítésére az **ARP protokoll** segítségével végzett címfeloldást követően kerülhet sor. A címfeloldás egy adott fizikai cím hozzárendelése egy IP címhez az ARP protokoll által koordinált kérdés-válasz (ARP REQUEST, ARP REPLY) során. Az APR protokoll működése arra épül, hogy a hálózatban létezik **broadcast cím**, amely címre elküldött kérését minden hálózaton lévő hoszt megkapja.

Mivel minden hoszthoz megérkezik a broadcast címmel kiküldött csomag, a keresett IP címmel rendelkező host, felismerve, hogy a kérésben saját IP címe szerepel, válaszcsomagot küldd a feladónak, amelyben megadja egyedi fizikai azonosítóját (MAC address). A kérést intéző hoszt a további kapcsolatépítés hatékonyságának javítása érdekében egy ideig tárolja az IP cím – fizikai cím párokat (ARP cache).

A címfeloldás



Az **Internet Protokoll** az internet hálózat egyik alapvető szabványa. Ezen protokoll segítségével kommunikálnak egymással az internethez kapcsolódó eszközök. A protokoll meghatározza az egymásnak küldhető csomagok (datagram) felépítése mellett azt, hogy az egyes hosztok milyen szabályok szerint kezeljék azokat.

Az IP csomag fejlécének szerkezete

4-bit	8-bit	16-bit	32-bit	
Ver.	Header Length	Type of Service	Total Length	
Identification			Flags	Offset
Time To Live	Protocol		Checksum	
Source Address				
Destination Address				
Options and Padding				

A tűzfalak működése szempontjából az IP csomag fejlécének **Time To Live** (TTL) mezeje kitüntetett figyelmet érdemel. A TTL mező az Internetre küldött csomag maximális élettartamát korlátozza. A feladó által küldött csomag az útba eső forgalomirányítók (útválasztó, router) közreműködésével jutnak el a címzett hosztig. A TTL mező értékét minden egyes útválasztó 1-gyel csökkenti. Ha a mező értéke 0-ra változik, akkor a router eldobja a csomagot és erről értesítést küldd a feladónak.

A TCP/IP modell része az **ICMP protokoll** (Internet Control Message Protocol), amely a hálózati problémákat hivatott detektálni és jelezni. Az ICMP integráns része az IP specifikációnak, megvalósítása minden IP csomagban kötelező. Ha valamelyik router olyan körülményt észlel, amelyek lehetetlenné teszik a datagram továbbítását, akkor ezt jeleznie kell a feladó részére, lehetővé téve, hogy az intézkedhessen a hiba kezeléséről. **Az ICMP a hibajelzésen túl hálózat diagnosztikai funkciók ellátására is felhasználható.**

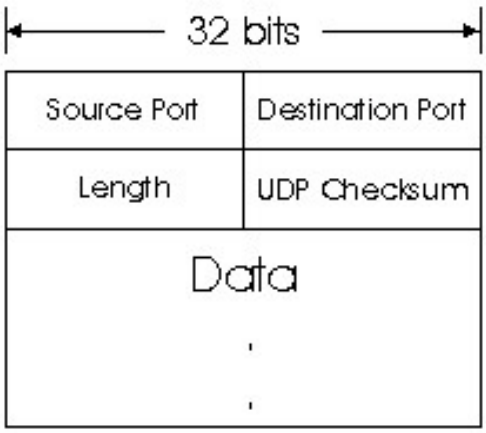
A dolgozat témáját tekintve legfontosabb ICMP üzenetek:

- **echo**: ellenőrizhető, hogy az adott IP című hoszt elérhető-e (ping parancs)
- **host/port unreachable**: adott hoszt/port elérhetetlenségére utal, amennyiben az IP csomag nem tud végső címzettjéhez eljutni
- **time exceeded**: a router küldi, amennyiben a vett csomag TTL mezejének értéke nulla, azaz a csomag élettartama lejárt

A hálózati csomagok végső címzettje a csomagot felhasználó program (process) operációs rendszerbeli absztrakciója. Ezt a képzelt elemet reprezentálják a **kapuk** (portok). Egy hoszton belül a portokat egy 16 bites előjeltelen bintáris szám azonosítja.

Az Internet Protokoll egy adott címre feladott csomagot a hosztig szállítja, de nincs eszköze arra, hogy a hálózatban üzemelő gép portjait is megcímezze. Egy adott hoszton belül a portok címzését az **UDP** (User Datagram Protocol) teszi lehetővé. Az UDP szolgáltatás az IP-ra épül, ahhoz képest csak a port-címzés lehetőségét nyújta.

Az UDP datagram szerkezete



Ahhoz, hogy egy távoli kliens kapcsolatba léphessen a server egy adott programjával, a server IP címén túl ismernie kell annak port címét is. Az IP-cím és a portszám együttese alkotja a **socketet**, amely lehetőséget nyújt bejövő adatcsomag megfelelő alkalmazáshoz való kézbesítésére.

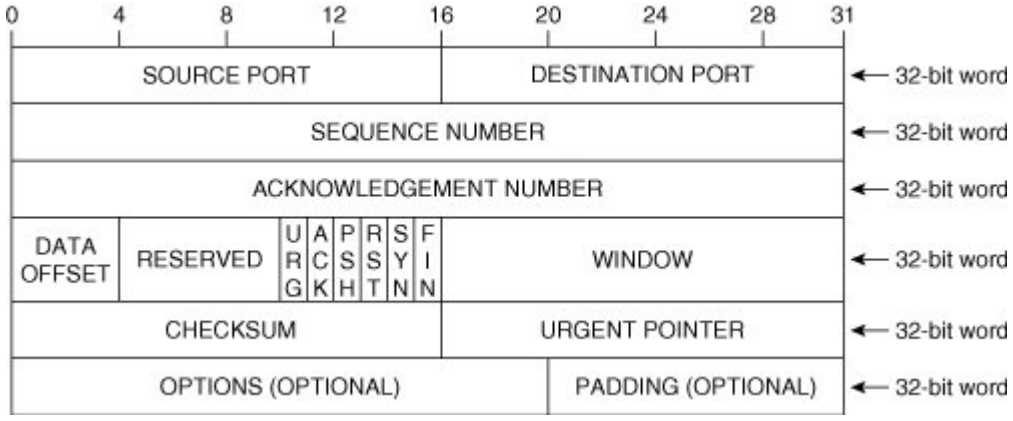
A jól ismert portok

Port Number	Protocol	Application
20	TCP	FTP data
21	TCP	FTP control
22	TCP	SSH
23	TCP	Telnet
25	TCP	SMTP
53	UDP, TCP	DNS
67, 68	UDP	DHCP
69	UDP	TFTP
80	TCP	HTTP (WWW)
110	TCP	POP3
161	UDP	SNMP
443	TCP	SSL
16,384–32,767	UDP	RTP-based Voice (VoIP) and Video

Az UDP feladata rövid, gyors üzenetek küldése. Jellemzően akkor használják, amikor a gyorsaság fontosabb a megbízhatóságnál, mert az UDP nem garantálja a csomag megérkezését.

A hálózat számos gyakorlati felhasználása nem lenne képes feladatát ellátni, ha az átvitel alatt adatvesztés következne be. A **TCP** (Transmission Control Protocoll) mentesíti az alkalmazásokat hibamentes adatátvitel megvalósítása alól. A TCP az IP-re épülve nyugtázási eljárásával megbízható, sorrendhelyes szállítási szolgáltatást nyújt.

A TCP szegmens felépítése



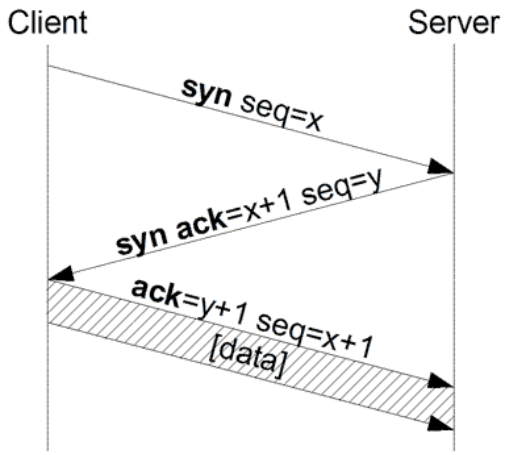
Fő feladata egy megbízható, és biztonságos kapcsolat kiépítése és fenntartása két folyamat között.

A TCP kapcsolat felépítését egy úgynevezett „háromutas” kézfogás előzi meg:

1. a **kliens SYN** (szinkronizáló) csomagot küld
2. a **szerver SYN-ACK** csomaggal nyugtáz
3. a **kliens ACK** csomaggal nyugtáz

A kapcsolat ettől a ponttól működőképes. Ezután megkezdődik az adatok átvitele, és a kapcsolat mindaddig nyitva marad, amíg bármelyik fél nem kéri annak megszakítását egy FIN (egyirányú) vagy RST (azonnali kétirányú) vezérlőbit jelzéssel.

A háromutas kézfogás



A fent bemutatott protokolloknak (ICMP, UDP, TCP) kulcsszerepe van a hálózatra csatlakozott számítógépek elleni támadások előkészítése során.

3. Hálózatok felderítése

A támadások előkészítése során az általános és technikai információgyűjtést követő lépés az elérhető eszközök és elérhető szolgáltatások keresése.

Az **elérhető eszközök keresésének** alapvető eszköze az ICMP protokoll echo üzenetére építő ping program. A ping segítségével ellenőrizhető, hogy az adott távoli számítógép elérhető-e egy IP hálózaton keresztül.

Az **ICMP scannelés** lényege, hogy a támadó ICMP echo kérésekkel árasztja el az adott hálózati címtartományt, és ahonnan ICMP echo reply jön, azok aktív, működő hosztok. Az eljárással felderíthető, hogy milyen eszközök találhatóak meg egy lokális hálózaton belül.

A hálózati architektúra felderítésére alkalmas eszközt a traceroute program nyújt. Az ICMP time exceeded üzenettel operáló traceroute egy számítógép-hálózati diagnosztikai eszköz, amely segítségével meghatározhatóak az IP hálózaton áthaladó csomagok útvonalai. A traceroute parancs használatával egy hacker gyorsan feltérképezheti a közte és a megismert elérhető eszközök között elhelyezkedő útvonalválasztókat.

A támadások előkészítését az elérhető szolgáltatások keresése teszi teljessé. Ahhoz, hogy egy támadó kapcsolatba tudjon lépni egy hoszt adott szolgáltatásával a célpont IP címén túl ismernie kell annak port címét is (socketet).

A **portszkennelés** a célpontként megjelölt eszközön éppen hallgató portok szisztematikus átvizsgálását jelenti, azzal a céllal, hogy felderítésre kerüljön mely **TCP illetve UDP portok** vannak nyitva, továbbá azok mögött milyen alkalmazások figyelnek.

A portszkennelés eredménye egy adott porton:

- **nyitott port (OPEN)**: a hoszt egy választ küldött, jelezve ezzel, hogy egy szolgáltatás hallgatja a portot, ekkor a számítógép **sebezhető**
- **zárt port (CLOSED)** vagy nem engedélyezett vagy nem hallgató: a host küldött egy üzenetet, ezzel jelezve, hogy a kapcsolatokat elutasítja ezen a porton (pl.: egy TCP RST-vel válaszol), különböző hacker eszközökkel a hoszt ekkor is **sebezhető**
- **szűrt vagy blokkolt port (FILTERED, STEALTH):** nem volt válasz a hosttól, az automatizált támadások ellen **védekezési módot jelent**

A jól ismert portokból kiindulva következtetni lehet, hogy milyen szolgáltatások futnak az adott rendszeren. Minden nyitott port egy potenciálisan sérülékeny alkalmazás lehet. A portszkennelés eredménye kellő kiindulási pontot jelent a támadók számára, ugyanis a megszerzett információk alapján elkezdhetik felkutatni azon sérülékenységet kihasználó programokat (exploit), amelyeket kifejezetten ezekhez az alkalmazásokhoz írtak. A legismertebb és széleskörűen használható port scanner program az Nmap.

A portszkennelés az UDP és/vagy a TCP protokollra támaszkodik. A TCP és az UDP protokoll közötti alapvető különbségekből adódóan felmerül a mérés futásidejének kérdése.

Az UDP protokollal történő felderítés esetében nem történik szinkronizáció, egy port megszólítása során minden esetben meg kell várni az időkorlátot. Ez problémát jelenthet viszonylag sok UDP port szkennelése során.

A TCP protokoll használatakor a két végpont között a háromutas kézfogásnak köszönhetően szinkronizációs folyamat történik meg, és annak meghatározott időn belül létre kell jönnie, vagy a kapcsolat létrehozására irányuló törekvés megszakad. Mivel a kapcsolat teljesen kiépül, így többnyire megjelenik a naplófájlokban, emiatt könnyen detektálható és szűrhető.

Speciális TCP protokollt használó szkennelési módszer a **TCP SYN scan** (stealth scan vagy half-open scan), amely egy be nem fejezett TCP kapcsolódást hajt végre. A szkennelést végző egy TCP csomagot küld SYN flaggel a célhostnak.

A hoszt válasza felvilágosítást ad a portról:

- **a port nyitva van**: ha egy SYN/ACK csomagot küldve elfogadja a portcsatlakozást
- **a port zárt**: ha egy RST-csomaggal válaszol
- **a port szűrt**: ha nem érkezik válasz

A célhoszt válaszát követően a támadó RST üzenettel kezdeményezi a kapcsolat azonnali megszakítását. A TCP SYN szkennelés előnye, hogy mivel a kapcsolatépítés nem fejeződik be,

ezért ezt számos rendszer nem veszi kapcsolódási kísérletnek. Ennek köszönhetően kisebb feltűnést kelt az támadott hálózaton.

A hálózatok felderítése és a belső és a külső hálózatok határvonala elleni támadások elleni védekezés bástyáját a tűzfalak jelentik.

4. A tűzfalak szerepe

A határvonal a kívülről érkező támadások egy lehetséges belépési pontja. A teljes bejövő és kimenő forgalom, így a hálózatok felderítése és a sérülékenységet kiaknázó támadások is keresztülhaladnak a határvonalon. A tűzfal két vagy több hálózat között elhelyezkedve, a fizikai és logikai szeparációkon keresztül nyújt preventív kontrollt a kívülről érkező támadásokkal szemben. A tűzfalak célja, hogy a hálózati szegmensek elválasztása és a hálózati forgalom szűrésén keresztül csökkentsék a sikeres behatolás valószínűségét.

A tűzfal a számítógépen található erőforrásokat teszi elérhetetlenné a külvilág felé a portok elrejtésével. A portok elrejtésével biztosítható, hogy számítógépünk beleegyezésünk nélkül ne válaszoljon a nem kívánatos kérésekre, illetve egy program tudtunk nélkül ne cseréljen információt a hálózattal.

A különböző tűzfal megoldások eltérő technikákat használnak annak meghatározására, hogy mely forgalom számára legyen engedélyezve vagy tiltva az erőforrásokhoz való hozzáférés.

5. A tűzfalak működése a fejlődés tükrében

A tűzfalakat úgy kell konfigurálni, hogy védelmet nyújtsanak a támadások ellen, webhely, IP-cím, forgalmi minta, alkalmazás és a protokoll alapú szűrés biztosításával.

Az egyes tűzfal technológiák-e konfigurálási lehetőségek komplexitásában jelentősen eltérnek egymástól.

a. Csomagszűrő

A tűzfalak leggyakrabban alkalmazott típusa a csomagszűrő tűzfalak (packet-filter firewall). Működési elvüket tekintve **az átáramló csomagok mindegyikét megvizsgálva azok tulajdonságait összevetik a felhasználó által meghatározott szabályrendszerrel**. A csomag vizsgálatánál csak a fejlécét elemzik, a magasabb rétegeket (alkalmazási réteg) adatnak tekintik, nem foglalkoznak vele.

A csomagszűrők a fejlécet azonban több szinten vizsgálják. Az **IP szint minden esetben kiértékelésre kerül**, azaz a döntést mindenképpen befolyásolja a csomag forrás és cél címe, esetleges fragmentálás adatai, illetve ritka esetben még az IP opciók értékei is.

A **legtöbb implementáció estében kiértékelésre kerül a TCP/UDP protokollokat tartalmazó adatkapcsolati réteg** is. Ezek plusz információt jelentenek a döntési szabályok megfogalmazásakor, lehetőség nyílik a forrás, illetve a cél portra való szűrésre, illetve a TCP flagek figyelembe vételére is.

Ha a csomag fejlécéből nyert információk illeszkednek valamely szabályban **megfogalmazott mintákra** (forrás cím/port, cél cím/port) akkor a csomag a szabályban meghatározott döntés értelmében **eldobásra vagy átengedésre kerül**.

A csomagszűrők **előnye** a gyorsaság és az egyszerűen definiálható szabályok.

Hátrányuk, hogy a szabályrendszer összetett esetekben nagyon nagyra dagad, ezért kezelhetetlenné válik. Működési elvükből következően pedig nem alkalmasak bonyolult igények (pl.: adattartalom vizsgálata) implementálására, az átmenő forgalom összetett szűrésére.

A csomagok egymástól elkülönülve kerülnek kezelésre, amelynek eredményeképpen nincs lehetőség a csomagok közötti összefüggések figyelembe vételére és a TCP kapcsolatok állapotának megbízható nyomon követésére, ami támadási felületet biztosít.

b. Állapottartó csomagszűrő

Az állapottartó csomagszűrők (stateful packet filter) működésük során a bejövő csomagokat tulajdonságaik alapján fogadják el, továbbítják, vagy dobják el, ami megegyezik a csomagszűrő tűzfalakkal.

A csomagszűrő tűzfalakhoz képest a vizsgálat mélységében nem, de az abból nyert információk feldolgozásában jelentős különbség van. Az állapottartó csomagszűrő magát a **kapcsolatot, a csomagok közötti összefüggéseket is** (nem csak 1-1 csomag fejlécét) **képes vizsgálni** a fejlécekre hagyatkozva.

A tűzfal azonosítani tudja a kapcsolat kezdetét és végét, valamint a kettő között zajló adatfogalmat, ezáltal ki tudja szűrni a kapcsolatba nem illő, potenciális veszélyt jelentő csomagokat.

Az állapottartó csomagszűrők is szabályláncolattal (mintaillesztés) dolgoznak. A döntés során a teljes TCP és IP rétegből kinyerhető információkat (forrás, cél port és IP, seq és ack, csomagok sorrendje illetve helye) figyelembe veszi.

Az állapottartó csomagszűrő legfőbb **előnye** a csomagszűrőkhöz képest az, hogy kevesebb szabály alapján tudnak működni és a kapcsolatorientált protokollok esetében (TCP) plusz lehetőségeket, ennek köszönhetően fokozottabb biztonságot képes nyújtani.

A TCP protokoll esetében képesek a kapcsolat állapotának követésére, azaz képesek megkülönböztetni a kapcsolat kiépülését végző csomagokat, a kiépülés után adatot közvetítő

csomagokat, valamint a kapcsolat lezárását. A tűzfal figyelembe veszi azt is, hogy adott csomag csak adott helyen jelenhet meg a kommunikációban. Például adatot tartalmazó csomag nem előzheti meg a kapcsolat kiépítését.

TCP csomagok esetében nem csak az ACK megléte, vagy hiánya szolgáltat alapot a döntéshez, hanem a teljes TCP kapcsolat nyomon követése adja meg a segítséget a tűzfalnak. Az új kapcsolat kezdeményezése ennek köszönhetően nem csak a SYN flag meglétén múlik, ezáltal a tűzfal **képes ellehetetleníteni a TCP SYN portszkennelési technika alkalmazását**.

Hátránya, hogy a klasszikus csomagszűrő tűzfalakhoz hasonlóan az ismeretlen elemeket szűrés nélkül átengedi.

Az állapottartó csomagszűrő tűzfalak jelentős továbblépést jelentenek a klasszikus csomagszűrőkhöz képest, de működésük során továbbra is csak a fejlécből nyert adatokra támaszkodnak.

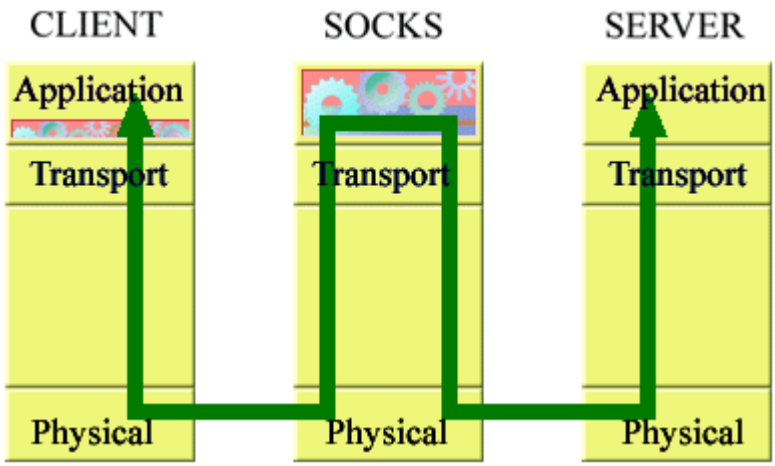
A funkcionalitás bővítésének biztosítását a tűzfalfejlődés másik ágát jelentő alkalmazásszintű tűzfalak teremtették meg.

c. SOCKS tűzfalak

A SOCKS tűzfalak félúton helyezkednek el a csomagszűrő és a proxy megoldások között. A SOCKS proxyk működési elve alapján **beépül az alkalmazás és a TCP réteg közé**.

A működés során egy speciális a kliens gépre telepített **alkalmazás elveszi a kapcsolatot az operációs rendszertől és a tűzfalnak adja**. Amikor egy program kapcsolódni akar egy szerverhez, akkor a kapcsolódási kérését a socks modul kezeli és a program helyett az előre beállított socks proxyhoz kapcsolódik, majd megadja a proxynak, hogy milyen címre szeretne kapcsolódni.

A SOCKS tűzfal működési elve



Ezután a proxy kapcsolódik a kliens program által kijelölt szerverhez. A kapcsolat kiépülése után a kliens program az adatforgalmat a SOCKS proxyn keresztül végzi.

A megoldás hálózati szempontból nem tekinthető transzparensnek (kliensek a tűzfal IP-jét címzik), de a program szempontjából igen, mivel a programon nem kell külön beállítást végezni.

A SOCKS csomagszűrőnek nem tekinthető, mert a csomagok a kliens és a szerver között nem közlekednek. Nem tekinthetők alkalmazásszintű tűzfalnak sem mivel a forgalom nem alkalmazási szinten kerül szűrésre, hanem csak hálózati szinten.

A SOCKS **előnye**, hogy a lehetőség van az operációs rendszerbe beépülő modul alkalmazására. A SOCKS megoldás segítségével, az átengedett protokolltól függetlenül lehetőség nyílik az átmenő forgalom tetszőleges authentikálására.

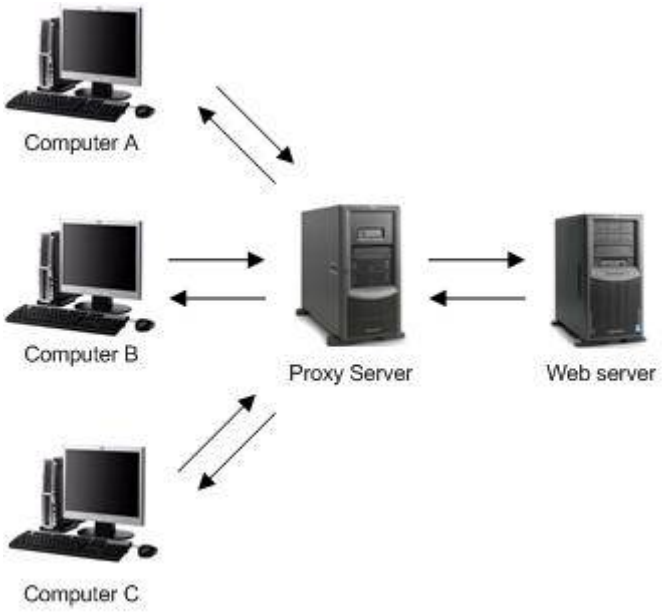
A megoldás **hátrányai** közé tartozik, hogy a szerver nem védhető és az ismeretlen elemek kezelése során alapvetően nincs alkalmazás szintű védelem.

d. Proxy tűzfalak

A proxy tűzfalak döntéseinek alapját már az alkalmazási réteg protokolljai jelentik.

A proxy tűzfalak nem teszik lehetővé a közvetlen kommunikációt a külső és a védett hálózat között. A **kliens és a kiszolgálók között nem épül fel közvetlen kapcsolat, hanem mindketten a tűzfalon futó proxy alkalmazással kommunikálnak**. A proxy az egyik hálózati csatolójával az ismeretlen hálózat kiszolgálóihoz kapcsolódik, a másikkal pedig a belső hálózatban található kliensekhez.

A proxy tűzfal működése



A proxy szerver a kliens nevében eljárva csatlakozik a megadott szerverhez, és igényli az erőforrást számára. A proxy esetlegesen megváltoztathatja a kliens kérését vagy a szerver válaszát, esetenként kiszolgálhatja a kérést a szerverhez való csatlakozás nélkül is (cache).

A proxy architektúra képessé teszi a tűzfalakat arra, hogy **alkalmazásszinten ellenőrizzék a rajtuk áthaladó információáramot**. A proxy alkalmazások már nem csupán a csomagok fejlécét vizsgálták, hanem azok adatrészébe is belenéznek és akár módosításokat is végrehajtottak.

Például az alkalmazásszűrés könnyedén fel tudja ismerni a különbséget a peer-to-peer fájlmegosztás és a hagyományos http forgalom között. Az alkalmazásszűrés segítségével megelőzhetjük, hogy a belső felhasználók fontos és titkos információt küldjenek a hálózaton kívülre.

A proxy tűzfalak **előnye** az alkalmazásszintű védelemnek köszönhető kifinomultabb szűrés és a többcsatornás protokollok elemzésének lehetősége.

A proxy tűzfalak számára **nem jelent gondot a több portot használó protokollok tűzfalazása**, mivel az architektúrának köszönhetően a proxy az alkalmazásszintből minden információval rendelkezik az újabb kapcsolet megnyitásához.

A kapcsolat kettősségéből kifolyólag a proxy tűzfalak **képesek kivédeni a csomagszintű támadásokat**.

A proxy tűzfalak megvalósításukat tekintve összetettek, döntésük alapja egyrészt mintaillesztés a hálózati rétegekben másrészt mintaillesztés és értelmezés az alkalmazási rétegben. **Hátrányuk**, hogy kizárólag olyan kommunikáció koordinálására alkalmasak, amelyek értelmezésére képesek. A használni kívánt protokollnak támogatnia kell a proxy-s működést.

6. Konklúzió

A kívülről érkező támadások elleni védekezés szempontjából, a tűzfalak szükségesek a hálózatba történő behatolás megakadályozásához, azonban még mindig nem elégségesek a hálózat biztonságához. A valódi hálózati biztonsághoz elengedhetetlen ezenkívül a termékek és szolgáltatások kombinációját egyesítő, átfogó biztonságpolitika és az azzal kapcsolatos elkötelezettség.