DEPARTMENT OF
NETWORKED SYSTEMS
AND SERVICES

Hálózatbiztonság

VIHIBB01 – Kódolás és IT biztonság, 2020

Dr. Holczer Tamás

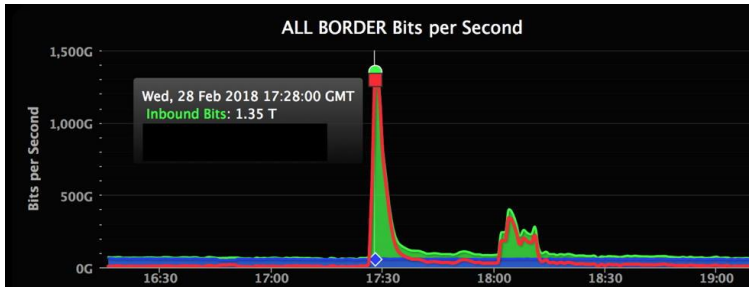
CrySyS Lab, BME
holczer@crysys.hu

www.crysys.hu

MŰEGYETEM 1782

Ismert támadások az elmúlt évekből

- WannaCry ransomware (2017)
  - ~200,000 áldozat ~150 országban
  - Több milliárd dollár veszteség
  - EternalBlue sérülékenység (Microsoft Windows 7,8,10… SMBv1 ellen)
- GitHub DDoS (2018)
  - 1,35 Tbps
  - Több tízezer forrásból indítva



- Mirai (2016)
  - IoT eszközök bevetésével (kamerák, routerek stb…)
  - DDoS Dyn DNS szolgáltató ellen

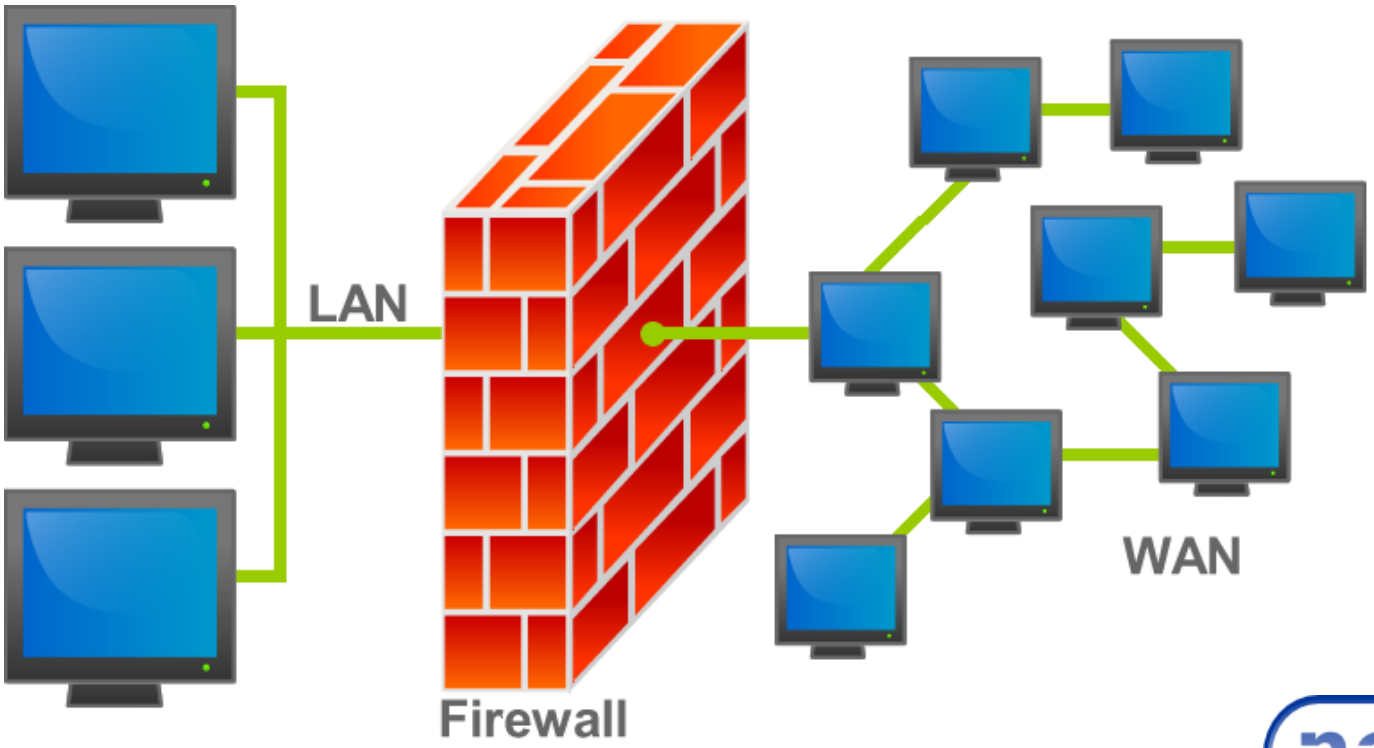
    (pár érintett: Netflix, PayPal, Sony PlayStation…)

Miről szól ez az előadás

- **Tűzfalak**
- **Behatolás érzékelő/megakadályozó rendszerek**
- **Virtuális magánhálózatok (említés szinten)**

- Miről nem szól az előadás (pedig lehetne):
  - DNS biztonság
  - 2. réteg biztonsága
  - Útvonalválasztás biztonsága
  - Csali megoldások
  - Spam
  - DDoS (elosztott szolgáltatásmegtagadásos támadások)

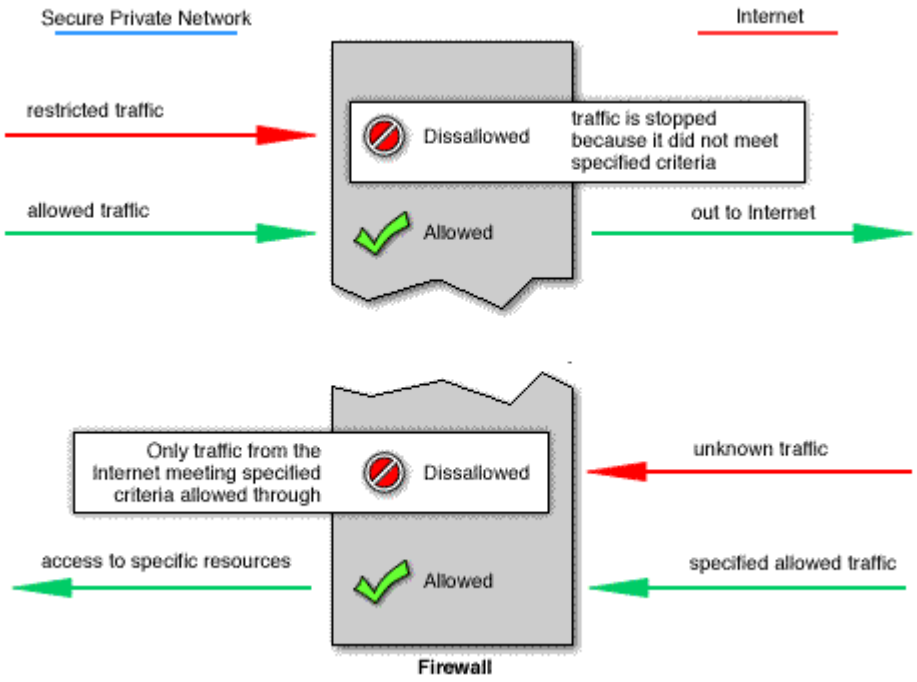
Tűzfalak alapgondolata



netfilter
firewalling, NAT, and packet mangling for linux

Mi az a tűzfal?

- Szabályok alapján hálózati hozzáférést szabályozó rendszer vagy rendszerek csoportja
  - A szabályok rendszerint szűrési szabályokkal vannak megadva
  - A bejövő és kimenő csomagok a szűrési szabályok alapján átengedésre vagy eldobásra kerülnek



Tűzfalak magas szintű tervezési céljai

- Minden forgalom kintről befelé vagy fordítva menjen keresztül a tűzfalon
  - A hálózat megfelelő topológiája és a tűzfal megfelelő elhelyezése teszi lehetővé
  - Sokféle jó topológia és elhelyezés lehetséges
- Csak az engedélyezett forgalmak mehetnek át (engedélyező lista biztonságosabb mint a tiltólista)
  - Megfelelő szabályokkal érhető el
  - A szűrés különböző rétegekben is megvalósítható
- A tűzfal maga is védett legyen a támadások ellen
  - Biztonságos OS, frissítés, karbantartás, minimális funkció halmaz

Csoportosítás 1.

- Csomagszűrő tűzfal: Tipikusan útvonalválasztóval együtt kerül kialakításra, a csomagokat a 3-as illetve 4-es rétegbeli fejlécek alapján szűri
- Állapot alapú tűzfal: Kapcsolatok állapotát (kezdeményezés, élő kapcsolat, lezárás) is figyelembe veszi a döntéseknél
- Alkalmazás szintű tűzfal (proxi tűzfal): Magasabb rétegbeli információkat is figyelembe tud venni a döntéseknél (bonyolult szoftveres megoldás)

Csoportosítás 2.

Típus	Előny	Hátrány
Csomagszűrő	• Egyszerű • Hatékony • Hardveres támogatás lehetséges	• Állapotmentes • Töredezett csomagok • Dinamikus portok
Állapot alapú szűrő	• Egyszerű • Hatékony • Könnyebb benne leírni a szabályokat • DoS ellen is védhet részben	• Több erőforrásra van szüksége • Sok támadás csak alkalmazásrétegben értelmezhető
Alkalmazás szintű szűrő	• Szofisztikált szabályok	• Nagy erőforrásigényű • Lassítja a forgalmat

- Nincs tökéletes tűzfal
- Döntés a szabályokon és a büdzsén is múlik
- Legtöbbször a típusok keveréke a legjobb megoldás

Csoportosítás 3.

- Hoszt alapú tűzfal: routerként működik, van IP címe
- Transzparens tűzfal: két összekapcsolt (bridged) interfész között szűri a forgalmat, IP szinten nem jelenik meg

- PC alapú tűzfal (például Linux egy desktop gépen)
  - Alacsony áteresztőképesség (relatív)
  - Jól konfigurálható
- Router alapú tűzfal (például Cisco vagy Juniper termékek)
  - Közepes áteresztőképesség
- Dedikált tűzfal (például Cisco ASA, FortiGate, Sophos UTM)
  - Magas áteresztőképesség
  - Közepes vagy nagy konfigurálhatóság
  - Gyakran Linux vagy BSD alapú megoldás

Csomagszűrés/ Hozzáférési listák 1.

- Alap (standard) formátum:

```
protocol source-addr [source-wildcard] destination-addr [destination-wildcard] {permit | deny}
```

  - protocol: ip/tcp/udp(/icmp)
  - source-addr: küldő címe, source-wildcard: küldők tartománya
  - destination-addr: fogadó címe, destination-wildcard: fogadók tartománya
  - permit | deny: döntés (engedélyez | tilt)

- Fejlettebb forma:

```
protocol source-addr [source-wildcard] sport destination-addr [destination-wildcard] dport {permit | deny} [log]
```

  - portok: küldő/fogadó
  - log: csomagok naplózása

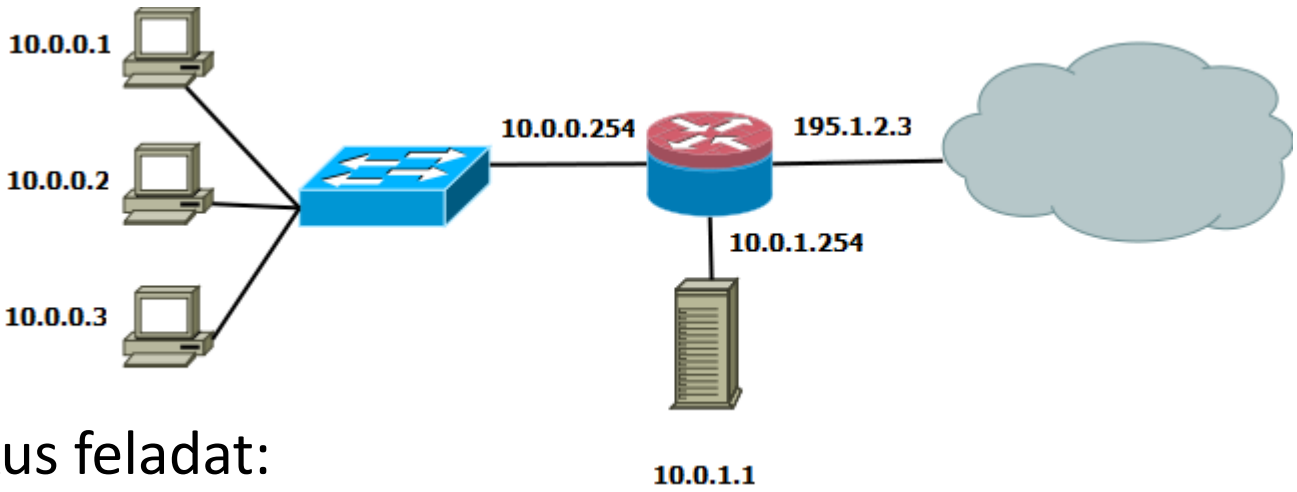
Csomagszűrés/ Hozzáférési listák 2.

- Fogadó IP cím (alhálózat)
- Cél port (tartomány)
- Küldő IP cím (alhálózat)
- Küldő port (tartomány)
- protokol (TCP, UDP, …)
- TCP flagek
  - SYN – kapcsolat kezdeményezés
  - ACK – visszajelzés korábbi csomagokról
  - FIN – kapcsolat lezárása
  - RST – kapcsolat megszakítása hiba esetén
  - …
- ICMP “type” és “code” mező
- Interfész neve, csomag iránya az interfészen

Csomagszűrés/ Hozzáférési listák 3.

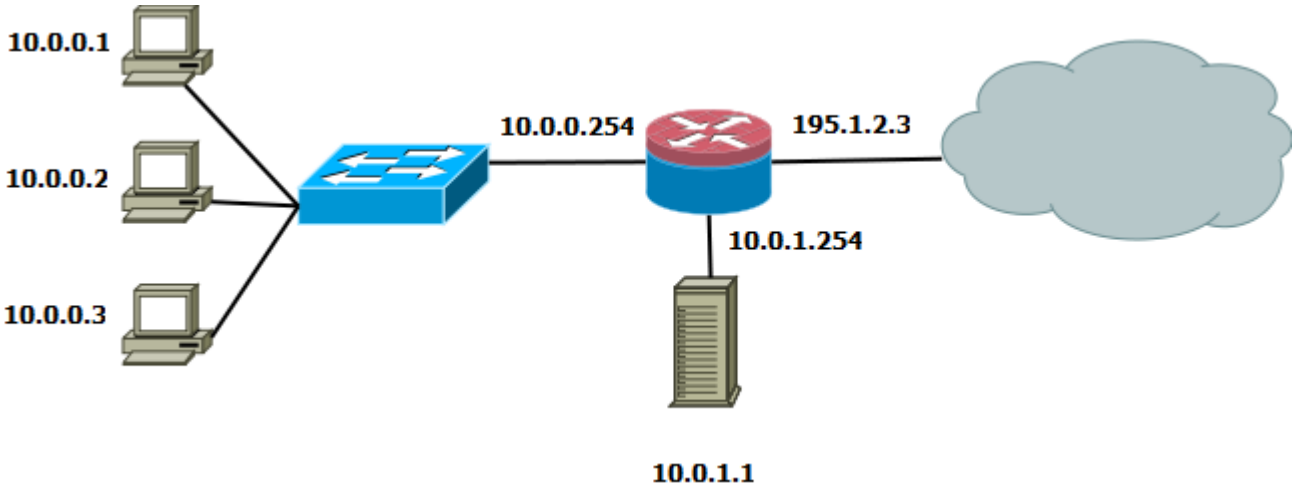
- Tervezési alapok
  - Listák „fentről-lefelé” vannak feldolgozva
    - » Első találat szabályozza a döntést (vagy utolsó, esetleg legjobb)
    - » Új szabály beszúrása történhet a lista elejére, végére, esetleg adott helyre középen
  - Alapértelmezett tiltás a lista végén (lehet alapértelmezett engedélyezés is)
  - Nem mindegy, hova kerül egy lista
    - » Közel a feladóhoz→ korábban ki lehet szűrni, kisebb kapacitás elég
    - » Közel a célhoz→ pontosabban meg lehet határozni a szűrést
    - » Döntés: melyik tűzfalra (melyik interfész melyik irányába)
  - Inkonzisztens szabályok
    - » Árnyékolás (általános specifikus előtt)
    - » Általánosítás (specifikus általános előtt)
    - » Korreláció (részben átfedő szabályok)
    - » Duplikátumok (ugyanaz a szabály többször)

Csomagszűrés/ Hozzáférési listák 4.



- Tipikus feladat:
  - A WEB szerver mindenhonnan elérhető legyen HTTP(S)-en
  - HTTP(S) forgalom mehet bentről kifelé
  - FW csak belülről menedzselhető
  - DNS, NTP, EMAIL… engedélyezve
  - Visszatérő forgalom engedélyezve
  - Minden más eldobásra kerül

Csomagszűrési feladat



- Feladat:
  - A WEB szerver mindenhonnan elérhető legyen HTTP-en
  - HTTP forgalom mehet bentről kifelé
  - Minden más eldobásra kerül

Probléma:
Kifelé nem mehet válasz
Támadó 80-as portról támadhat
TCP flagekkel részben orvosolható

- Megoldás:
  - src=any, sport=any, dst=10.0.1.1, dport=80 → ALLOW
  - src=10.0.0.0/24, sport=any, dst=any, dport=80 → ALLOW
  - src=any, sport=80, dst=10.0.0.0/24, dport=any → ALLOW
  - src=any, sport=any, dst=any, dport=any → DROP

Állapot alapú/dinamikus szűrés

- motiváció
  - Ha a belső gépek el akarnak érni külső erőforrásokat, akkor a választ vissza kell engedni
  - Állapotmentes esetben ez csak a TCP flagek segítségével oldható meg

src = <internal IP range>, sport = any, dst = any, dport = any, prot = tcp → ALLOW

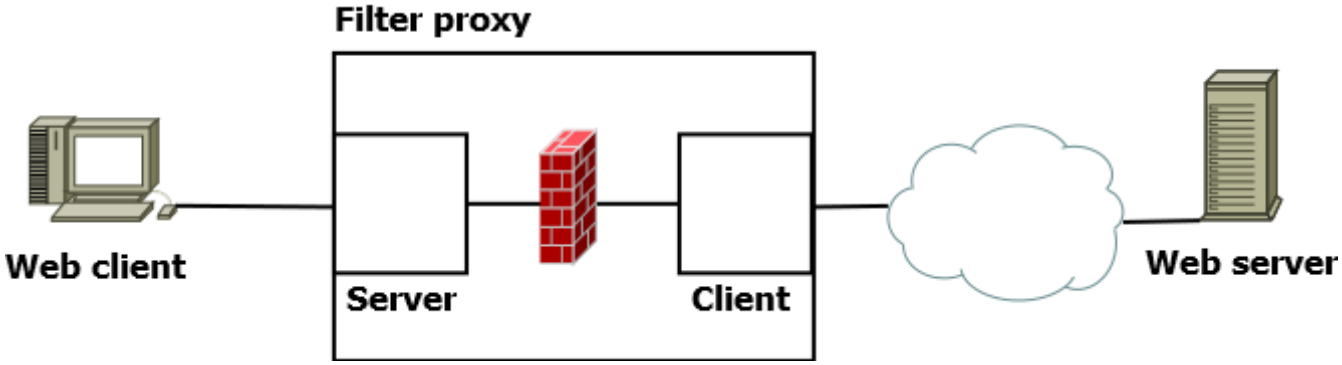
src = any, sport = any, dst = <internal IP range>, dport = any, prot = tcp, ACK = 1 → ALLOW

- Dinamikus szabályok követik a kapcsolatok állapotát, és a visszajövő csomagokat engedélyezik, ha azok egy létező engedélyezett kapcsolathoz tartoznak
  - Ha benne van a kapcsolat táblában, akkor mehet
  - Egyébként a sima csomagszűrő szabályok alapján kell dönteni
  - Sikeres 3-utas kézfogás után a kapcsolatok bekerülnek a kapcsolat táblába
  - Lezárt kapcsolatok kikerülnek a kapcsolat táblából

Miért jók a dinamikus szabályok?

- Csomagszűrű szabályokkal hasonló eredményt lehet elérni, de…
- Így biztosabb, hogy a megfelelő visszairányú csomagok jöhetnek csak be
- Néhány támadást ki lehet így zárni
- Sokkal könnyebb konfigurálni (kisebb az esély a hibázásra)
- Speciális protokollokat könnyebb kezelni (pl FTP, ami új adatkapcsolatot nyit, a kontrol csatorna mellé)

Alkalmazás szintű szűrés / Proxik



- Elrejti a belső struktúrát, kifelé csak a proxi látszik
- Alacsony szintű protokollok biztos nem jutnak át
- Csak azok a protokollok mehetnek át, amikre van proxi
- A proxi egy szerver és egy kliens kombinációja; a belső felhasználók a proxi szerver részével kommunikálnak, a kéréseket a proxi kliens része továbbítja, a kettő között van a szűrés megvalósítva
- Bármilyen alkalmazás rétegbeli adatra lehet szűrni

Alkalmazás szintű szűrés / Proxik

Problémák:

- SSL MitM (root CA-t szokás telepíteni)
- HPKP (fejléc törlése és VPN; Chrome or Firefox nem riaszt HPKP sértésre, ha kézzel telepített CA-t rakott be a proxi)

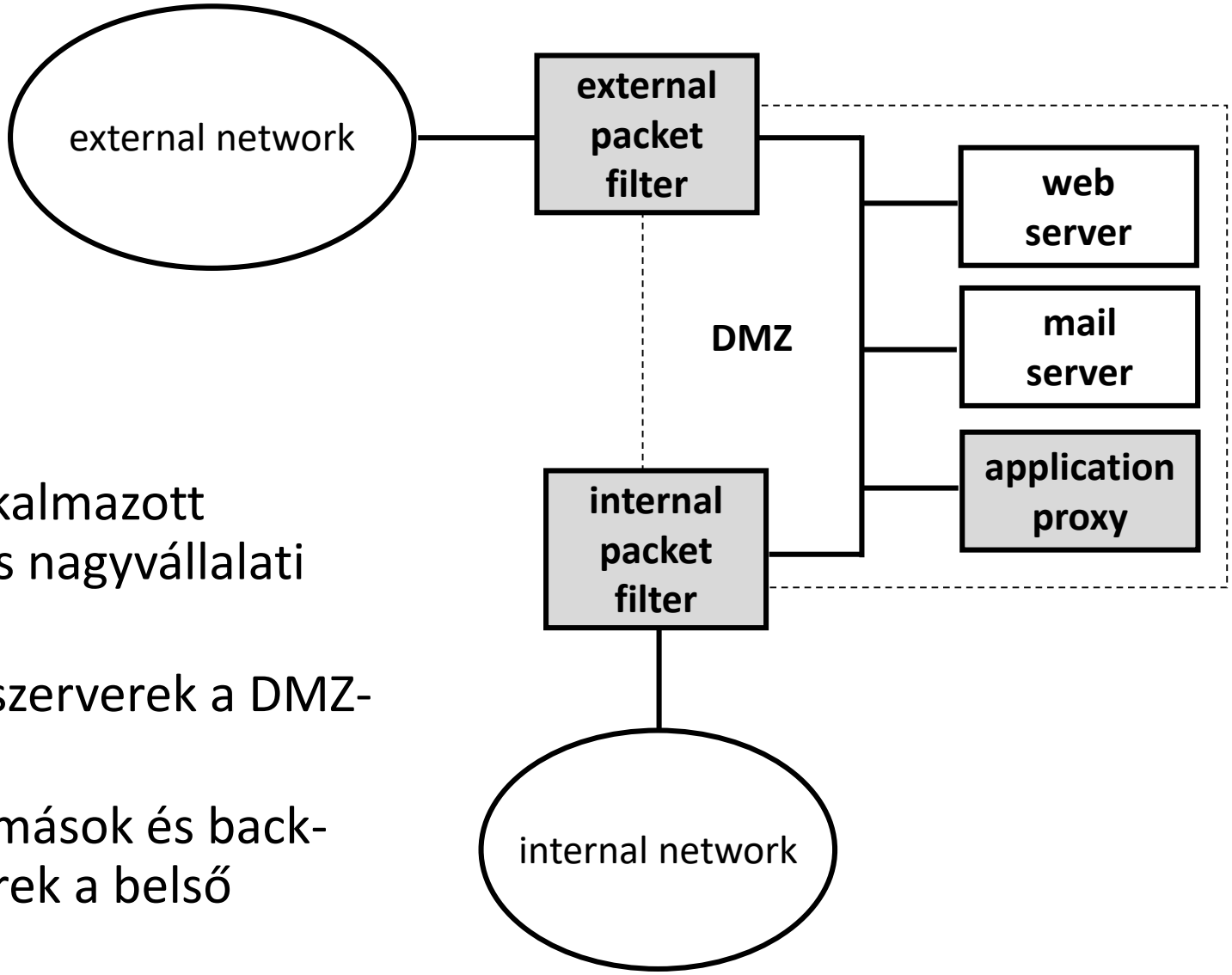
Cégeknek hasznos megoldás:

- Jól lehet szűrni a felhasználók aktivitását
- Gyakran össze van kötve valamilyen weboldal kategorizáló megoldással
- Nem megfelelő kategóriába eső URL-eket le lehet tiltani
- Naplók alapján lehet ellenőrizni a munkatársak aktivitását
- Oldalakat munkakör szerint is lehet szűrni (például a HR-nek szüksége van Facebook-ra, de a többieknek nincs)

Tűzfal helye

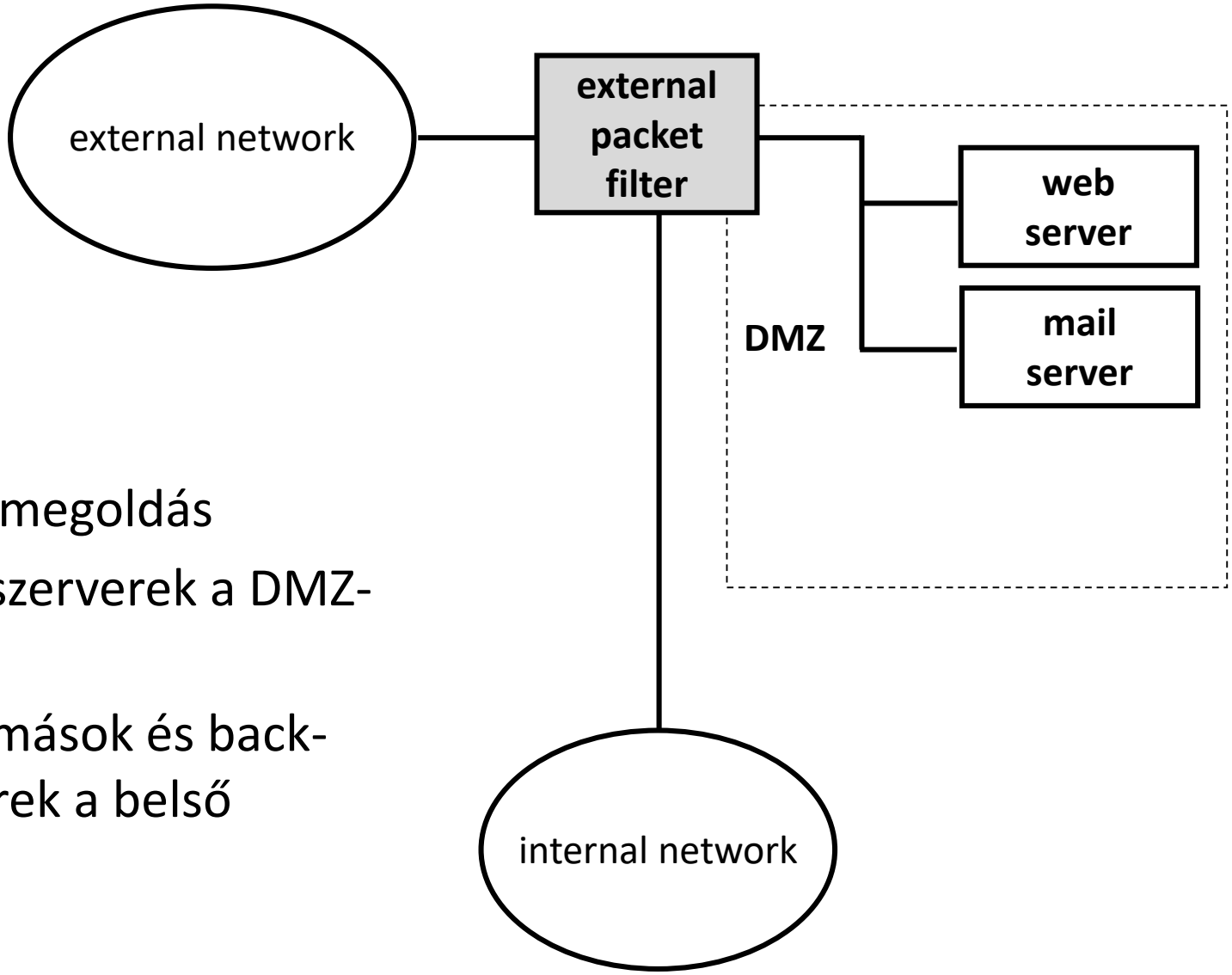
- Tipikusan több tűzfal együttesen alakítja ki a védelmet
- Nagyobb hálózatokban
  - Több tűzfal
  - Különböző fajta tűzfalak, amik kiegészítik egymást
  - Több jó megoldás is létezik
- A következő megoldások gyakran előfordulnak, de nem csak így lehet jól csinálni

DMZ architektúra / DeMilitarized Zone



- Gyakran alkalmazott biztonságos nagyvállalati megoldás
- Front-end szerverek a DMZ-ben
- Munkaállomások és back-end szerverek a belső hálózatban

Egyszerűsített DMZ architektúra



- Kisvállalati megoldás
- Front-end szerverek a DMZ-ben
- Munkaállomások és back-end szerverek a belső hálózatban

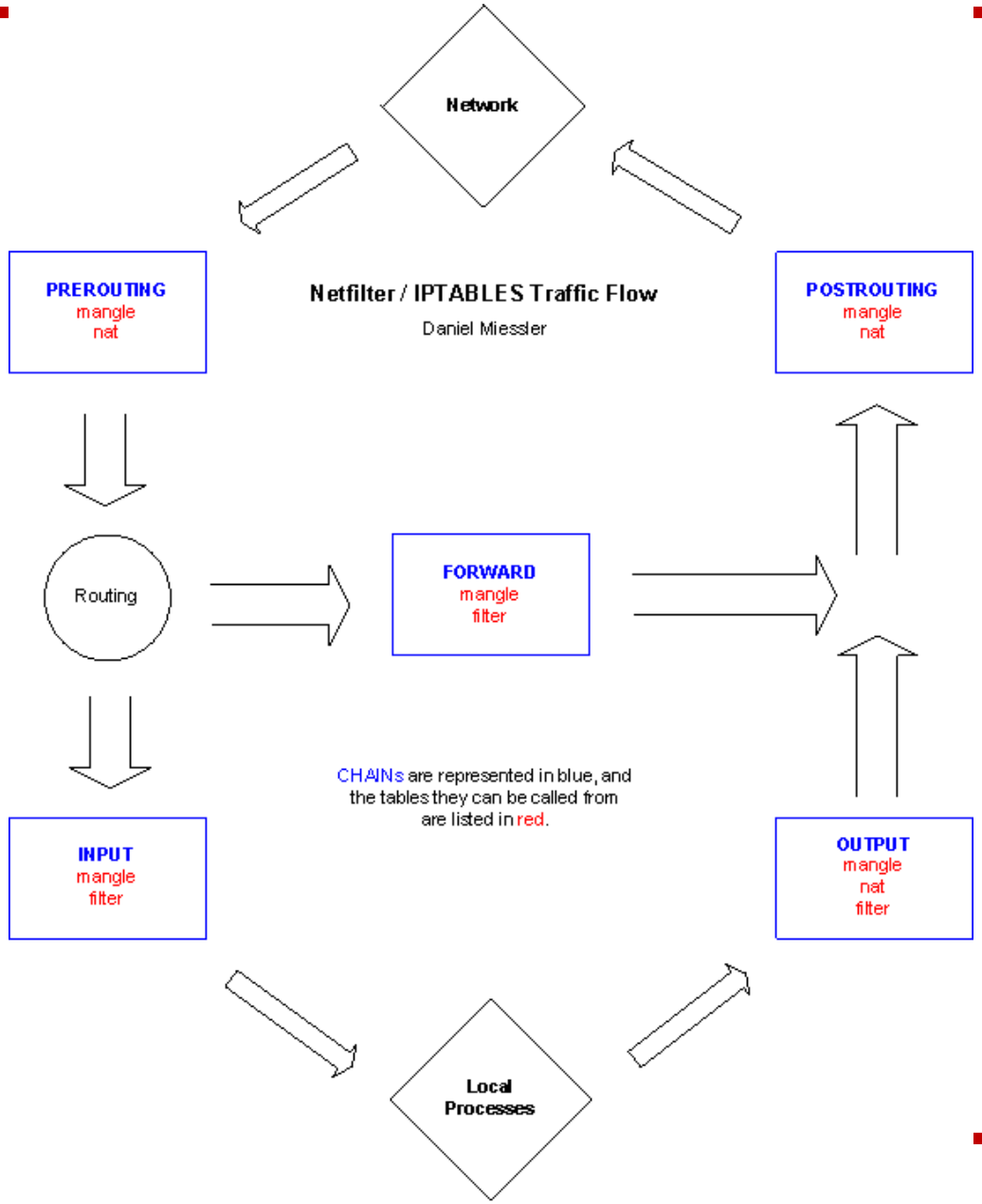
Tűzfalak limitációi

- Tűzfalak maguk is tartalmazhatnak sérülékenységet, be lehet rájuk esetleg jutni
- Tökéletes tűzfal sem véd ki mindent, de hamis biztonságérzetet kelthet
- Tűzfalak nem tudják kiszűrni a rajtuk át nem haladó támadásokat
  - CD, USB drive [-> Stuxnet]
- Új malware-ek ellen nem igazán hatékony megoldás
- Belső támadók és social engineering ellen nem hatékony
- Emaileket csak alkalmazásrétegben lehet szűrni (linkek, csatolmányok)

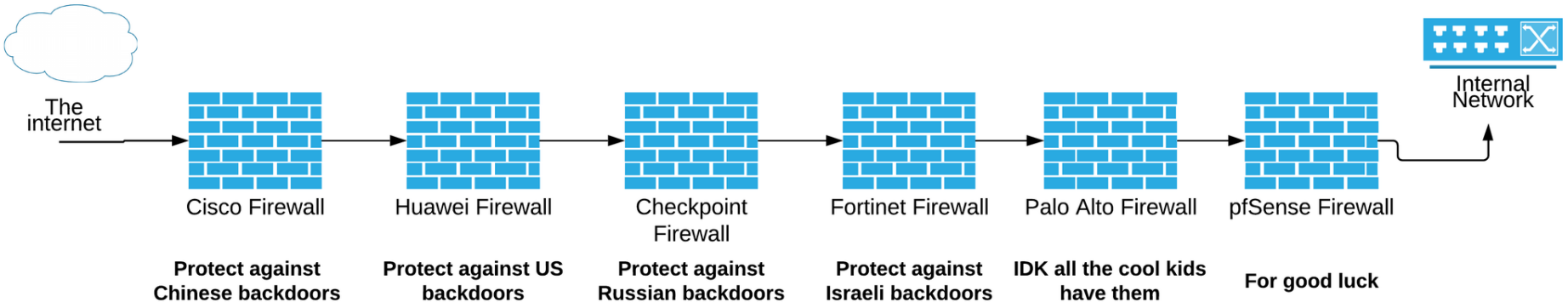
Példa: Netfilter/IPTables

- Linux kernel része
- Iptables paranccsal konfigurálható (más megoldások: ufw, grafikus szerkesztők)
- Útvonalválasztással együttműködik
- Szolgáltatások (táblák):
  - Szűrés/Filter (accept, reject, drop, log), alapértelmezett
  - NAT
  - Mangle (csomag fejléc módosításra)
  - Raw
- Tábla = láncok halmaza (PREROUTING, INPUT, FORWARD, OUTPUT és POSTROUTING.)
- Lánc = szabályok rendezett listája

Példa: Netfilter/IPTables



Melyik tűzfalat használjam?

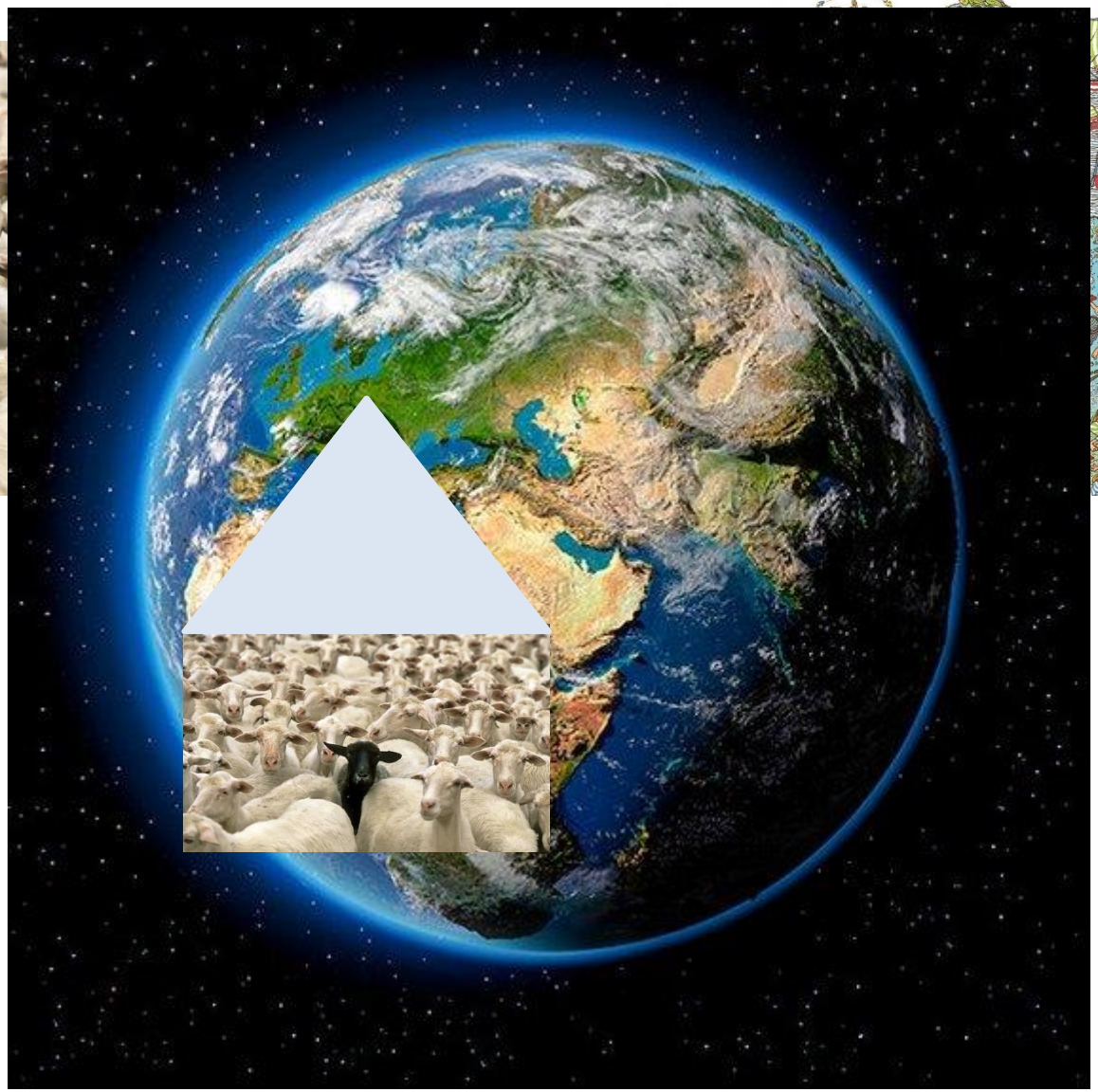


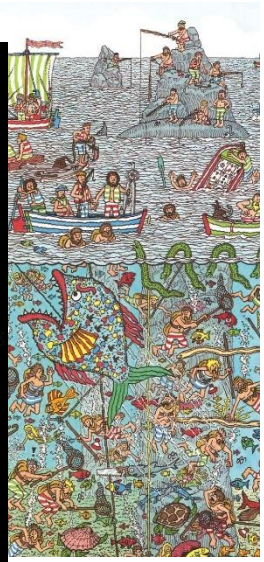
IDS: Meghatározás

- **behatolás**
  - Biztonsági esemény, amelyben egy behatoló megpróbál olyan erőforrásokhoz hozzáférni, amihez nincs jogosultsága
  - Gyakaran valamilyen sérülékenyslget kihasználva
  - Példák: remote root compromise, web server defacement, jelszó törés, backdoor telepítés …

- **Behatolás detektálás**
  - Biztonsági szolgáltatás amely figyeli a rendszer eseményeit, és megpróbálja a behatolásokat (közel) valós időben érzékelni, és riasztásokat generálni

Milyen nehéz feladat?





Miért van szükség behatolás detekcióra?

- Kell egy második vonal a tűzfal mögé, mivel az nem tökéletes
- IDS/IPS olyan mint egy riasztó egy házon
  - Jelez mielőtt nagy baj történne miután a betörő bejutott
    - » Gyors detekció nagyban csökkentheti az elszenvedett kárt
  - Támadók esetleg nem is próbálkoznak, ha nagy az esély a lebukásnak
    - » Vannak könnyebb célpontok
- Tűzfalak nem jók belső támadó ellen, de az IDS-ek igen
- IDS segítségével ki lehet deríteni, hogy történt a támadás, ezáltal jobbá lehet tenni a védelmet

“Második vonal”

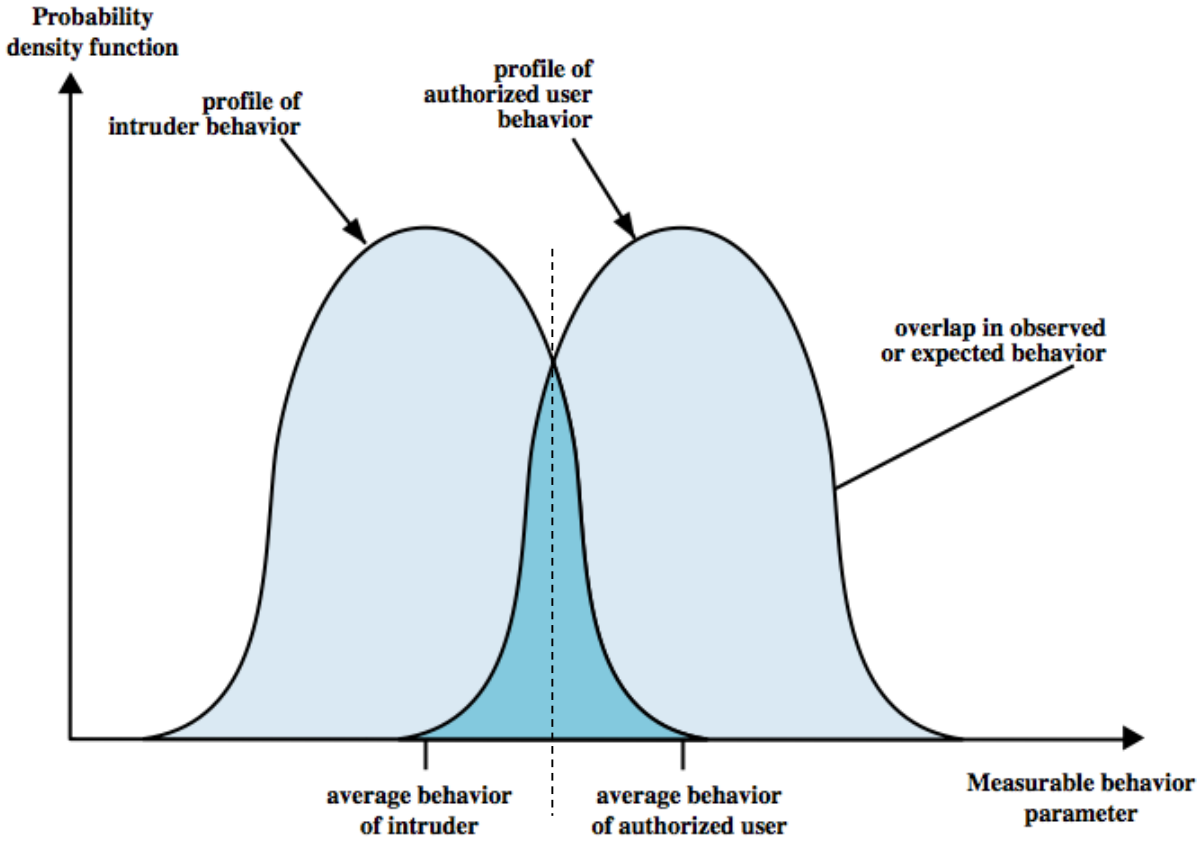
- Szoftverekben van hozzáférésvédelem
- Operációs rendszer ellenőrzi a jogosultságokat
- Tűzfalak szűrik a kapcsolatokat és akár a tartalmakat is
- Mégis szükség van egy sokadik védelmi vonalra is, hogy tudjunk róla, ha az előzők nem működtek.

- A védelem mindig több lépcsőből áll, és mindig érdemes felkészülni, hogy valamelyik nem működik megfelelően

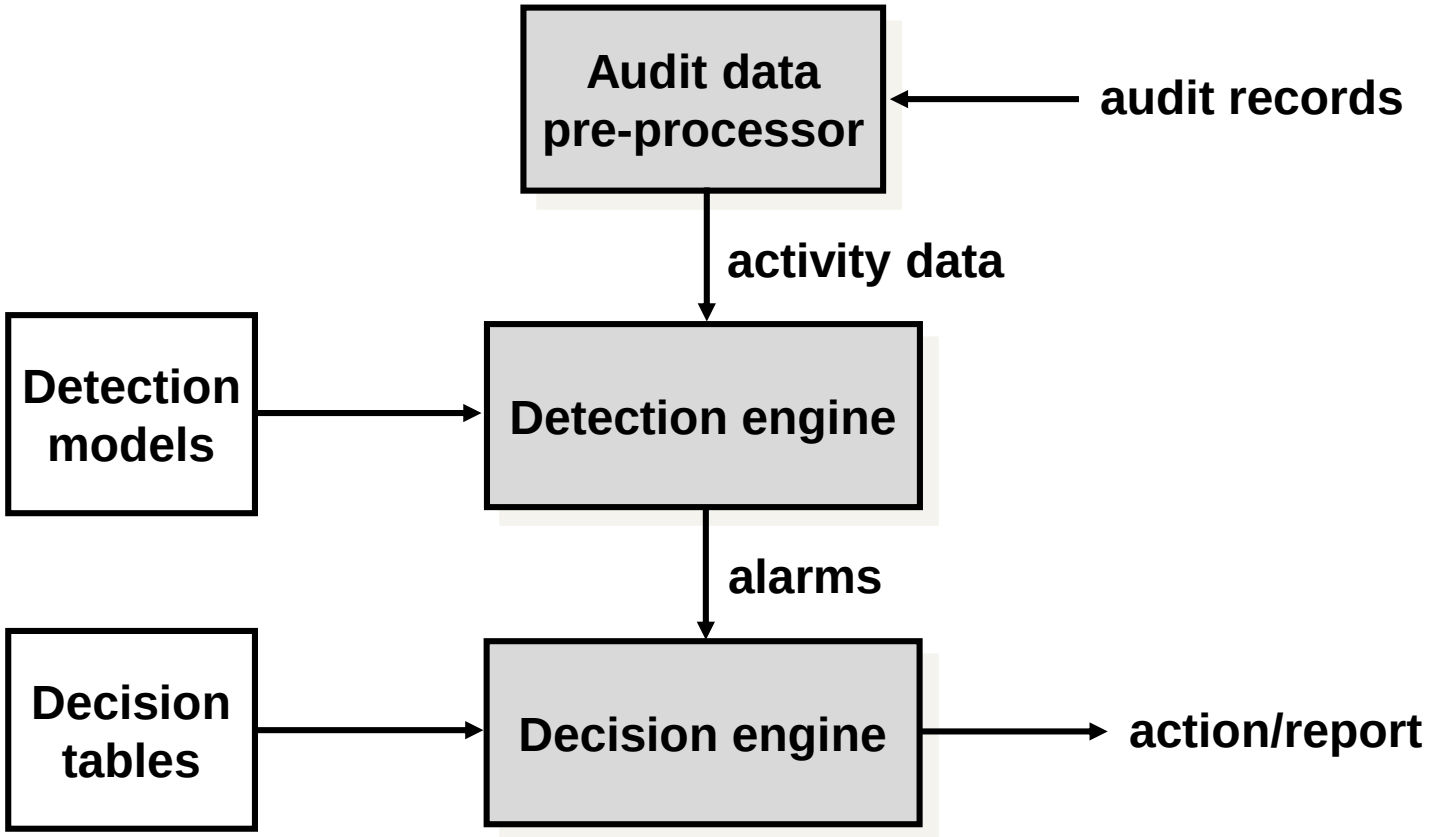
IDS-ek alapfeltevései

1. A rendszer megfigyelhető
2. A normális és támadó aktivitások elkülöníthetőek

- Mindig van átfedés
- False pozitív és negatív mindig előfordul
- Kompromisszumot kell találni



IDS részei


Audit data pre-processor
audit records
activity data
Detection models
Detection engine
alarms
Decision tables
Decision engine
action/report

IDS típusok detekció alapján

- Mintázat alapú detekció
  - Adatbázis a támadások mintázatával
  - Gyűjtött adatokat összehasonlítja az adatbázis tartalmával
  - Csak ismert támadásokat ismer fel
  - Hatékony
  - Ismeretlen támadásokat nem tud felismerni
- Anomália alapú detekció
  - Normális viselkedés profilját építi fel
  - Normálistól való eltérést ellenőrzi
  - Ismeretlen támdásokat is fel tud ismerni, de sok false riasztást generálhat (új normális aktivitásokat is támadásnak néz)
- Állapot alapú protokoll ellenőrzés
  - Protokoll állapotát figyeli
  - Megengedett következő lépéssel és állapotok
  - Lehet benne hihetőség vizsgálat (név < 50 char)
  - Számításigényes
  - Sok mindent nem tud detektálni(pl DoS)

IDS típusok elhelyezkedés alapján

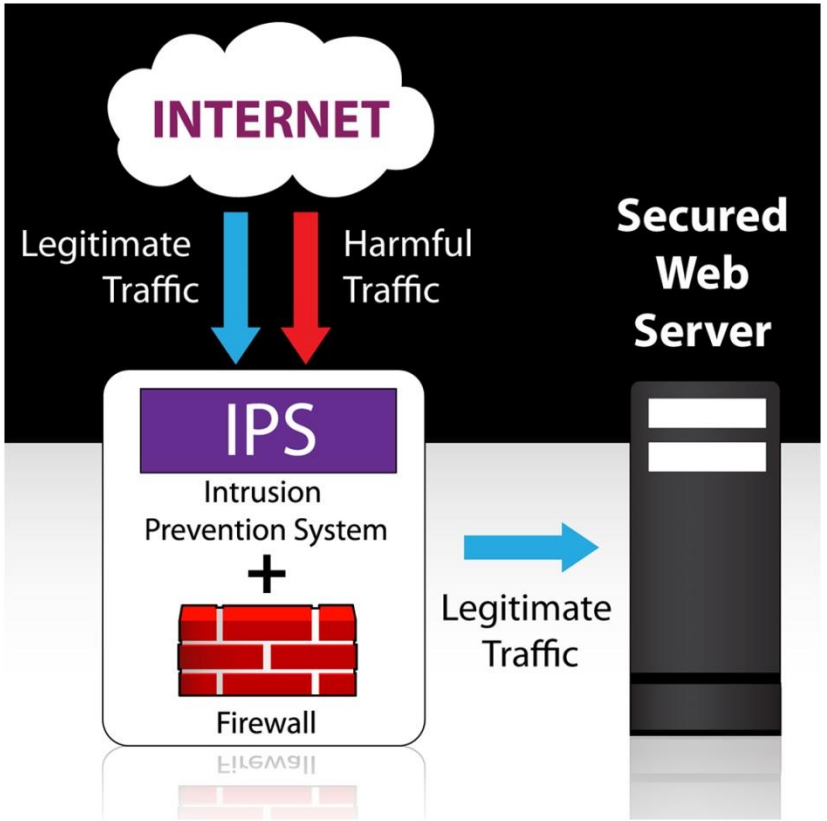
- hoszt-alapú IDS
  - Hoszton történő eseményeket figyeli
  - Naplókat és egyéb OS specifikus adatokat figyel
- Hálózat-alapú IDS
  - Teljes hálózati szegmenst tud ellenőrizni
  - Routereken és switcheken átmenő forgalmat tud vizsgálni
  - Ismert támadási mintákat keres, vagy eltéréseket a szokásos viselkedéstől
  - Csomagok adatrészét vizsgálja
  - Titkosított csomagokat nem tud vizsgálni

IDS vs IPS

- IPS = Intrusion Prevention System=

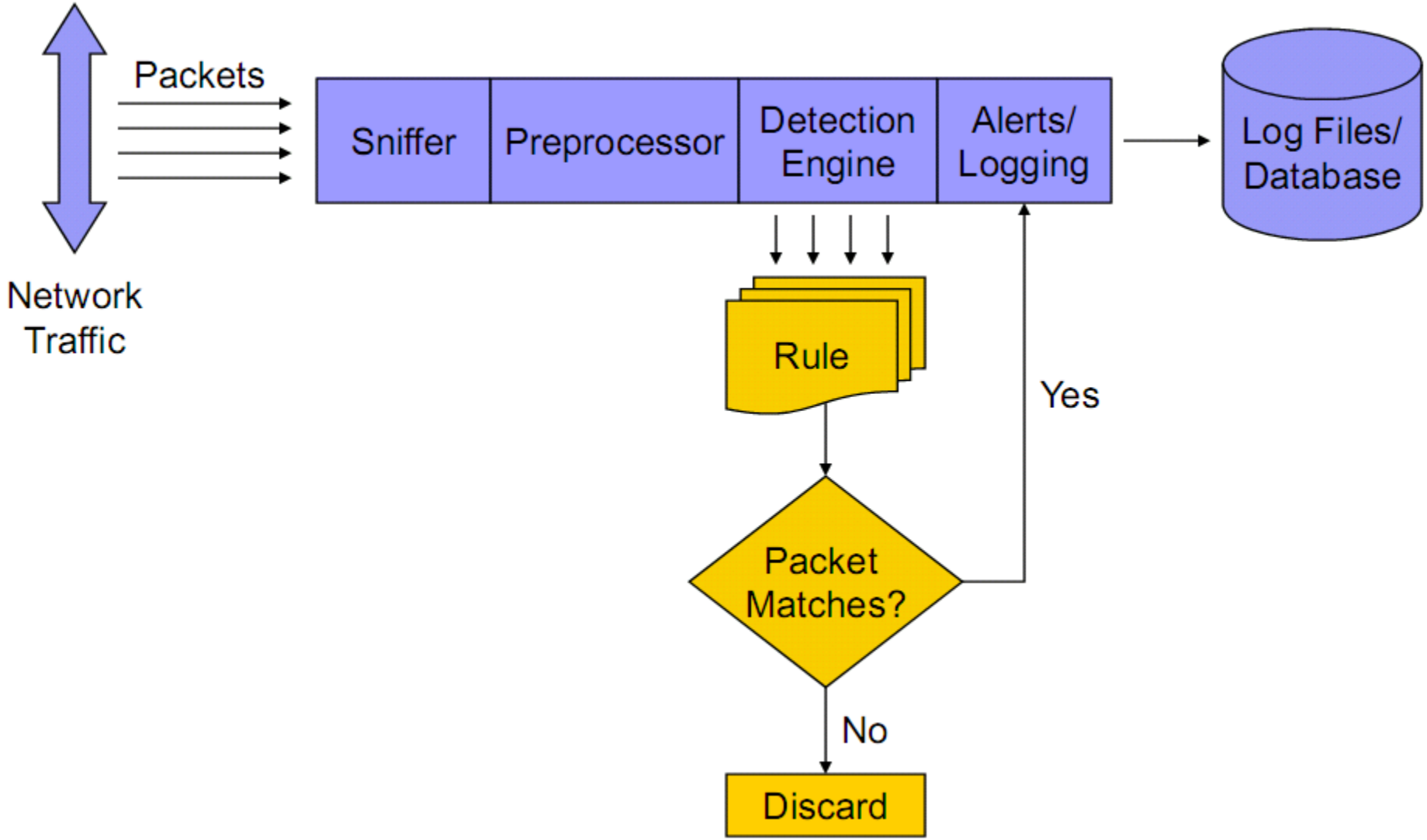
Behatolás Megelőző Rendszer

- Forgalom átmegy rajta
- Blokkolhat kapcsolatokat
- Szűk keresztmetszet lehet
- IPS~IDS+FW



- Hardver típusok
  - PC alapú (IDS vagy IPS)
  - Céleszköz
    - » Router/switch-be integrálva
    - » Bizt. eszköz (e.g.: ASA)
    - » IPS eszköz (e.g.: IPS 4270)

Snort detekció



Snort szabályok

- Egyszerű szabály

alert tcp any any -> 192.168.1.0/24 111 (content:"|00 01 86 a5|"; msg:"mountd access";)

- Conficker B botnet:

```
alert tcp any any -> $HOME_NET 445 (msg: "conficker.b shellcode"; content:
   "|e8 ff ff ff ff c2|_|8d|O|10 80|1|c4|Af|81|9MSu|f5|8|ae c6 9d a0|O|85
   ea|O|84 c8|O|84 d8|O|c4|O|9c cc|Ise|c4 c4 c4|,|ed c4 c4 c4
   94|&<O8|92|\;|d3|WG|02 c3|,|dc c4 c4 c4 f7 16 96 96|O|08 a2 03 c5 bc ea
   95|\;|b3 c0 96 96 95 92 96|\;|f3|\;|24 |i|95 92|QO|8f f8|O|88 cf bc c7 0f
   f7|2I|d0|w|c7 95 e4|O|d6 c7 17 cb c4 04 cb|{|04 05 04 c3 f6 c6 86|D|fe c4
   b1|1|ff 01 b0 c2 82 ff b5 dc b6 1f|O|95 e0 c7 17 cb|s|d0 b6|O|85 d8 c7
   07|O|c0|T|c7 07 9a 9d 07 a4|fN|b2 e2|Dh|0c b1 b6 a8 a9 ab aa c4|]|e7 99 1d
   ac b0 b0 b4 fe eb eb|"; sid: 2000002; rev: 1;)
```

(bináris és ASCII mintázatok vegyesen)

- Reguláris kifejezések is lehetnek szabályokban

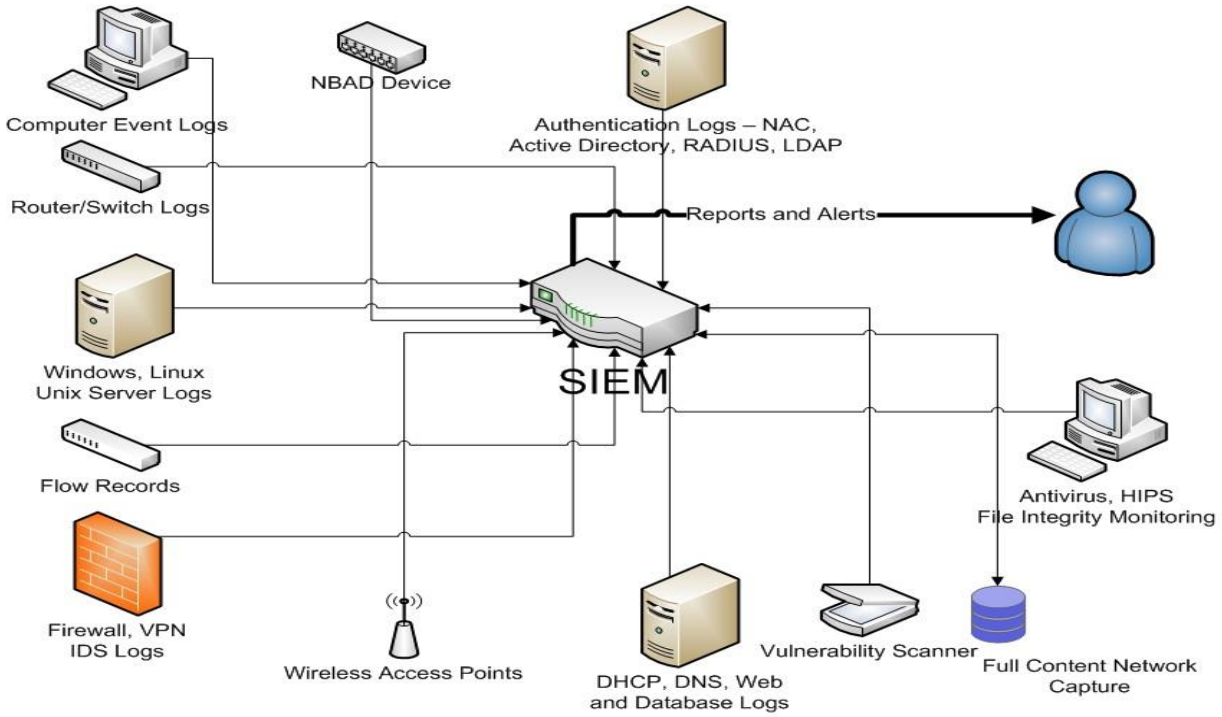
- Szabályokat írni/olvasni/megérteni nehéz feladat
- Szabályokat importálni szokták nem egyedileg megírni (fizetős és ingyenes források is vannak)

SIEM rendszerek

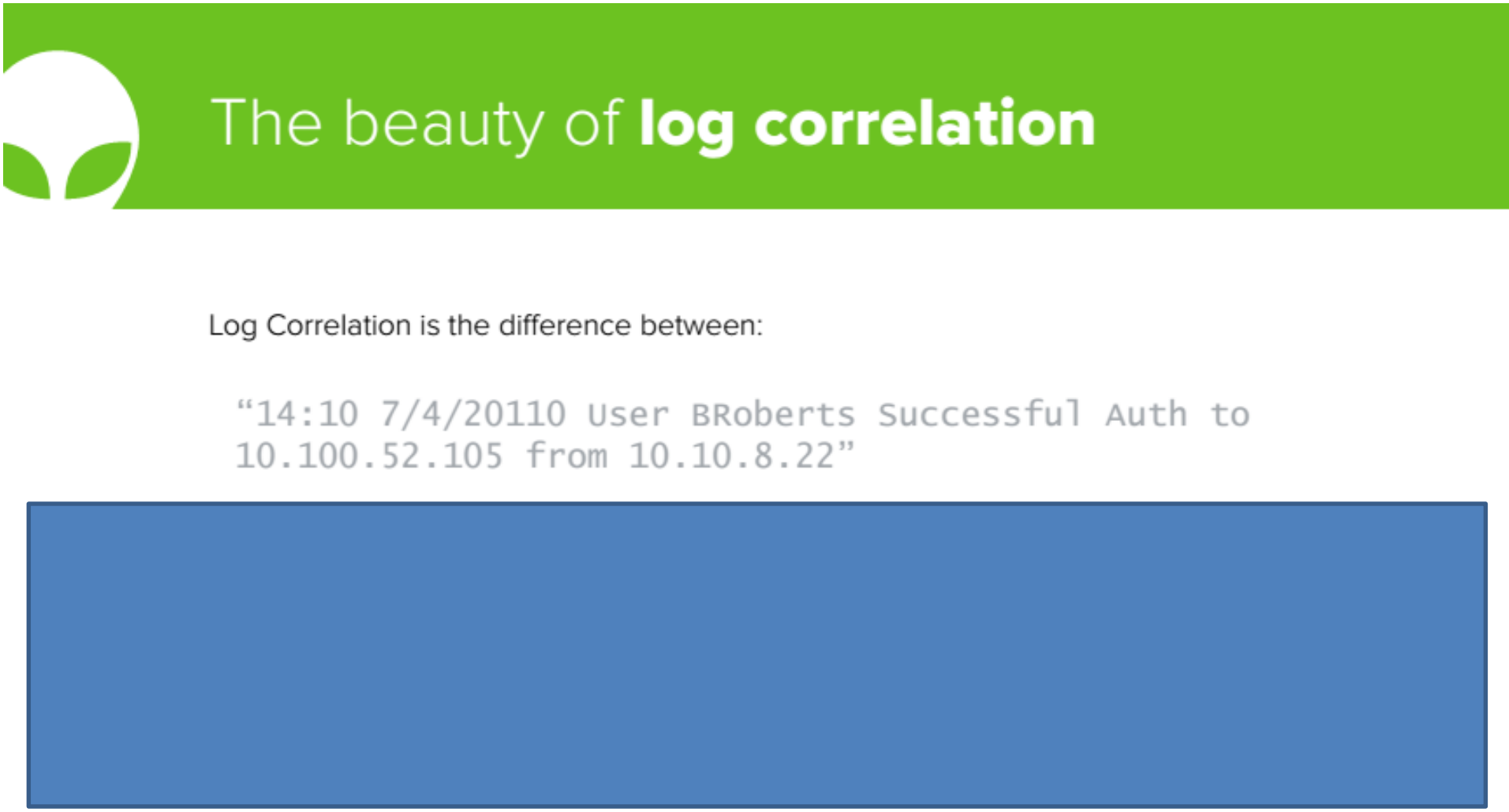
- Probléma: rengeteg log van rengeteg forrásból
- Sok log hasznos, de nehéz kezelni
- Mikor küldjünk riasztást?
- Big data problémák és megoldások
- Események közötti összefüggéseket figyelni kell
- False riasztások nagy részét el lehet kerülni

SIEM adatforrások

- Gyűjtsünk minél több helyről:
  - Munkaállomások
  - Adatbázisok
  - Webszerverek
  - Email szerverek
  - IPS
  - IDS
  - Antivirus
  - Tűzfal
  - Fájlszerver
  - Vezetéknélküli kapcs
  - NAS log
  - VPN log
  - SAP logs
  - …



Miért kell összefüggéseket vizsgálni?

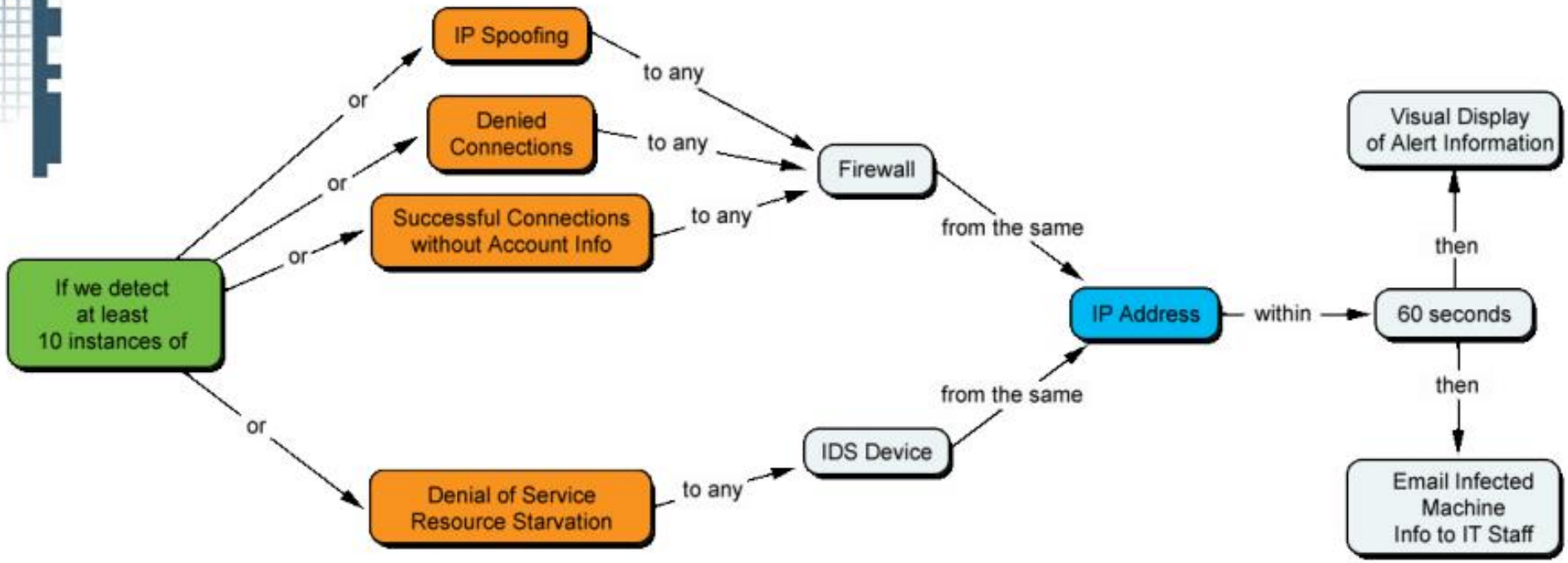


From alienvault.com

Miért kell összefüggéseket vizsgálni?

Correlation Rule Name: W32.Blaster Worm

The goal of this rule is to detect Blaster worm variants as well as other malicious code by analyzing network traffic patterns.



Network Intelligence CORPORATION

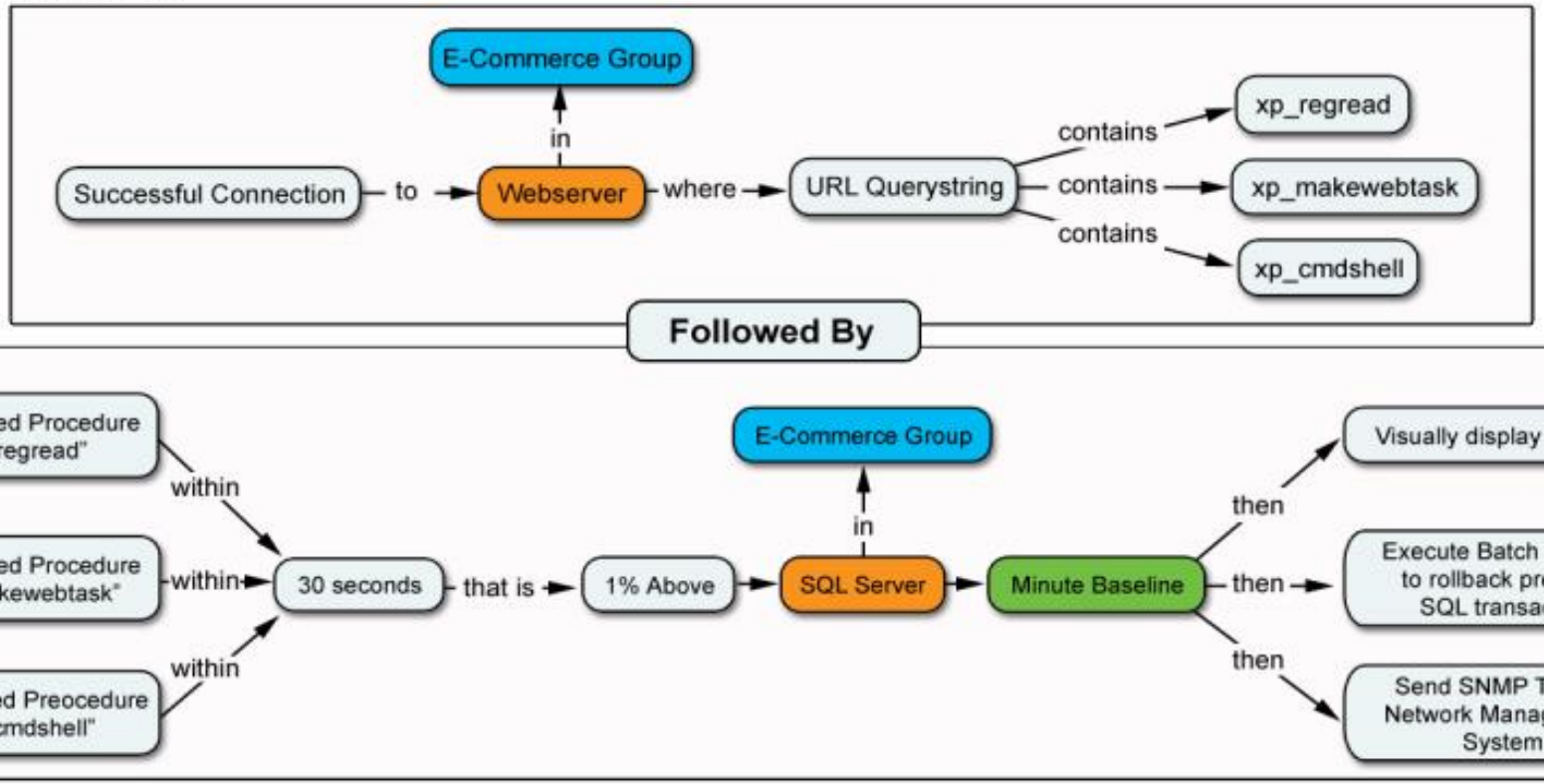
13

From: Network Intelligence Corporation

Miért kell összefüggéseket vizsgálni?

Correlation Rule Name: SQL Injection Attack

The goal of this rule is to detect information theft from E-Commerce websites through the exploitation of the trusted connection between the web server and the database.



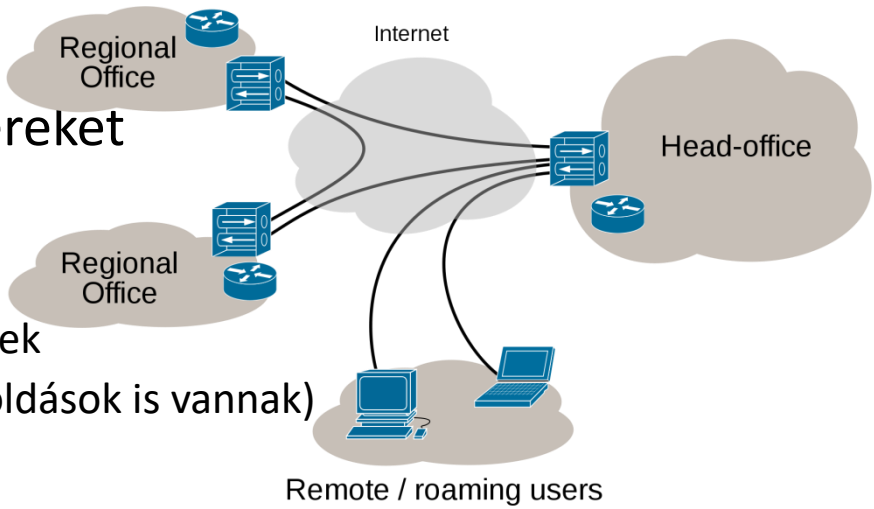
Network Intelligence CORPORATION

14

From: Network Intelligence Corporation

Virtuális magánhálózatok

Internet VPN



- VPN=Virtual Private Network
- Egy hálózatba hoz össze távoli rendszereket
- Alaptípusok
  - Site-2-site
    - » Két távoli alhálót köt össze, fix végződések
    - » IPSec gyakori megoldás (SSL alapú megoldások is vannak)
  - Road-warrior (távoli hozzáférés)
    - » Mozgó felhasználót bárhonnan beköt a központba
    - » SSL alapú megoldások a tipikusak (pl. OpenVPN)
- Routing alapján
  - Default gateway: VPN gateway
  - Proxy-ARP
    - » Felhasználók nem beszik észre, hogy VPN-el vannak összekötve
- Limitációk
  - Broadcast tipikusan nem megy át
  - 2. réteg üzenetei tipikusan nem mennek át

Hogy is működik egy VPN?

How VPN works

Kontrol kérdések

- Miért van szükség tűzfalakra?
- Mi a különbség egy csomagszűrő és egy állapot alapú tűzfal között?
- Hogy működik egy proxi tűzfal?
- Mire valók a láncok és táblák iptables esetén?
- Mire jó egy IDS?
- Hogy tud detektálni egy IDS?
- Milyen forrásokat használhat egy IDS/IPS/SIEM?
- Mi a különbség egy IDS és egy IPS között?
- Milyen problémát old meg egy SIEM?
- Mire való egy VPN?