

Szilágyi Ferenc

Cisco CCNA Security



ciscoworld.hu

www.ciscoworld.hu

Előszó

Köszönetet szeretném mondani mindeneknak, akik támogattak, bíztattak és mellettem álltak a könyv megírásának három hónapjában, akár türelmükkel, akár szavaikkal akár azzal, hogy meghallgatták a gondolataimat.

CCNA Security vizsgáról

CCNA Security vizsga napjaink egyik legjobb alapszintű hálózati biztonságtechnikával foglalkozó vizsgának számít. Megszerzésével nem csupán a CCNA Security minősítés (certificate), hanem az amerikai Nemzetbiztonsági Hivatal (NSA – National Security Agency) és az amerikai Nemzeti Biztonsági Rendszerekért felelős Bizottság (CNSS – Committee on National Security Systems) által meghatározott 4011-es szabványnak való megfelelés igazolása is jár.

Vizsgázás

CCNA Security vizsga bármely Pearson-VUE központban letehető. (www.vue.com)

Vizsgának a száma: 640-553 IINS (Implementing Cisco IOS Network Security)

A vizsga során átlagosan 55-65 kérdést kell megválaszolni. A kérdések között van 1-2 laborgyakorlat, 3-5 szimulációs feladat, a többi pedig feleletválasztós. A helyes válaszok számát is megadják, így a bizonytalan kérdéseknél is van esély a helytelen válaszok kizárással a helyeseket megtalálni.

A vizsgára 90 perc áll rendelkezésre, illetve a nem anyanyelvi szinten angolul beszélőknek plusz 30 perc.

A vizsga az angolban kívül Japán, Kínai, Orosz, Portugál, Koreai, Francia és Spanyol nyelven érhető el.

A vizsga 300 pontról indul és a maximális pontszám 1000, így 700 pont összegyűjtésére van lehetőség. A sikeres vizsgához 820-830 pontot kell elérni.

A vizsga előfeltétle a CCNA minősítés megléte. Ebből kifolyólag feltételezem, a könyv olvasója már rendelkezik a CCNA tananyag ismeretével, így az oda tartozó részeket (pl. ACL-ek) ebben a könyvben nem tárgyalom.

Vizsgázás után

A vizsga eredménye azonnal rendelkezésre áll, azonban a papírt postai úton küldik Amerikából, így egy 3-6 hetes átfutásra számítani kell. Mivel a minősítés érvényességi ideje 3 évenként lejár, ezért azt meg kell újjítani.

A megújjításnak (recertification) a módja, ugynaz, vagy azonos szinten levő vizsga, vagy akár egy magasabb szinten levő vizsga sikeres letétele.

Könyvről

A könyvet ajánlom mindeneknek, akik a CCNA security vizsgára akarnak felkészülni, vagy akár többet szeretnének megtudni az hálózati biztonságról.

Javításokat, frissítéseket a <http://www.ciscoworld.hu> weboldalon lehet megtekinteni, illetve letölteni.

Sok sikert kívánok a vizsgához és a karrierhez!

Szilágyi Ferenc
www.ciscoworld.hu

Tartalomjegyzék

Adatbiztonsági alapok	4
IP spoofing	5
Hijacking	5
IP source routing	5
Man-in-the-middle támadás	6
Trust relationship támadás	6
Támadási előkészületei	7
Adatok elleni támadás	7
Jelszavak lopása	7
Bizalmasság elleni támadás	7
Botnet	7
Kliensek biztonsága	8
Port scan és ping sweep	8
Cisco Security Agent	8
HIPS rendszerek	8
Email védelem	8
Rendelkezésre állás ellen támadások	9
TCP syn-flood	9
ICMP támadások	9
Hálózati biztonság kiértékelése	10
Hoszt és hálózat alapú védelem	10
Cisco eszközök hozzáférési módjai	11
IOS által használt jelszavak	11
Hibás bejelentkezések kezelése	11
Privilégium szintek	11
CLI view	12
Telnet/SSH/HHTP elleni támadások védelem	13
Figyelemfelkeltő banner üzenetek	14
Fájlok védelme	14
Cisco Self-defending network	14
Konfiguráció SDM (Security Device Manager)	15
AAA – helyi adatbázis használatával	16
AAA konfigurációja SDM-ben	17
Lokális authentikáció konfigurálása SDM-ben	19
AAA RADIUS és TACACS+ authentikáció	20
TACACS+ és RADIUS konfigurálása SDM-ben	22
Router Lock-Down	25
Out-of-Band management (OOB)	25
SSH és Telnet	26
Syslog	28
SNMP – Simple Network Management Protocol	28
Layer 2-es támadások	29
CAM overflowing – MAC spoofing és port security	29
VLAN hopping	30
Double tagging	31
Spanning-tree védelme - Root guard, BPDU guard	31
DHCP snooping	32
DHCP starvation	32
Dynamic ARP inspection (DAI)	32
VLAN ACL-ek (VACL)	33
Private VLAN	33
SAN (Storage Area Network)	34
VOIP biztonság	35
Cisco Identity-Based Network Services (IBNS)	36
Tüzfalak	37
Cisco tüzfal eszközei	37
ACL-ek	38
IP spoofing kikuszöbölése ACL-el	39
VTY hozzáférés	39
Classic Firewall	40
SPI és CBAC	40
Zone-Based Firewall	41
A zone-based firewall konfigurálásának lépései	42

Intrusion Prevention System (IPS) / Intrusion Detection System (IDS)	43
IPS konfigurálása SDM-en keresztül	44
IPS finomhangolása	47
Kriptográfia	51
Szimmetrikus és aszimmetrikus algoritmusok	52
Cipher típusok	52
Kulcsok mérete	52
DES, 3DES, AES és SEAL	53
Avalanche effektus	53
Hash	54
Digitális Aláírások (Digital Signatures)	55
RSA és digitális aláírások	56
Public Key Infrastructure (PKI)	56
SCEP (Simple Certificate Enrollment Protocol)	56
Site-to-Site VPN kapcsolatok	58
IPsec protokoll	58
Diffie-Hellman	58
IKE (Internet Key Exchange)	59
Authentication Header (AH) és Encapsulating Security Payload (ESP)	59
DMVPN	60
GRE (Generic Routing Encapsulation)	60
IPsec konfigurációja	61
IPsec konfigurálása SDM-en keresztül	63
IPsec monitorozása	69

Hálózati veszélyek forrásai:

- **Kivülről:** internetről hackerek, egyéb automatizált támadások (DoS)
- **belülről:** saját felhasználóink, azokra telepített rosszindulatú programok

Belső hálózatról jövő támadások sokkal veszélyesebbek, mint azok, amelyek kívülről jönnek:

- belső felhasználók jobban tudják, hogy néz ki a hálózatunk felépítése
- nekik vannak bizonyos jogosultságaik, amikkel a munkájukat is végezik
- hagyományos IPS és tüzfal rendszerek, nem a belülről érkező támadásokat figyelik, a belső felhasználók nem megfelelő magatartásának szabályozására alkalmatlanok

Három fő célja a hálózati biztonságnak:

- **bizalmasság** (confidentiality): illetéktelenek ne férjenek hozzá olyan adatokhoz, amikhez nem lenne szabad. Tüzfalakkal és ACL-ekkel, felhasználónév-jelszó párossal szabályozni lehet, hogy ki mihez férhet hozzá. Forgalom titkositásával pedig kiküszöbölnétejük, hogy az adatátvitel során lehallgatott adatok értelmezhetőek lesznek mások számára.
- **sértetlenség** (integrity): biztosítja, hogy adataink nem kerülnek módosításra illetéktelenek által, az adatátvitel során nem módosulnak az adatok. Továbbiakban biztosítja, hogy a forgalom ténylegesen attól jön, akitől jönne kell.
- **rendelkezésre állás** (availability): hálózat mindenkor működőképes legyen, egy támadás vagy valamely komponens rendellenes működése ne befolyásolja a hálózat többi részét.

Adatok klasszifikálása: a CCNA Security vizsgához az alábbi kormányzati klasszifikálást kell ismerni:

- Confidential
- Secret
- Top-secret

Természetesen ettől eltérhetünk, és saját osztályzási sémát is letrehozhatunk.

Sebezhetőségek:

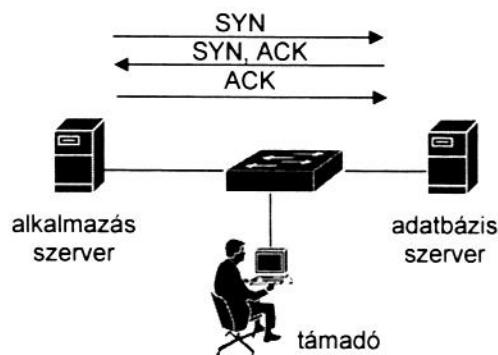
- fizikai események, katasztrófák (tűz, földrengés, tornádó, árvíz)
- design gyengesége: tüzfalak nem megfelelő elhelyezése
- protokollok gyengesége, amik használatban vannak: telnet - ssh
- szoftverek hibái, gyengeségei: buffer overflow (legnagyobb számban felfedezett szoftveres sérthatóségek)
- nem megfelelő konfiguráció: túl sok port van nyitva, helytelen tüzfalbeállítás, egyszerű jelszó engedélyezése
- ártalmas szoftverek: vírusok, spyware-ek
- emberi tényezők (akár szándék nélkül is): phishing, phising

IP spoofing

A támadó, valaki másnak az IP címét használja, valaki másnak adja ki magát.

Hijacking

TCP kapcsolatoknál ez nehezebb, mivel a TCP sorszámozásnál a szekvencia számot (sequence number) kitalálni, előre meghatározni nehéz lehet. Ha a támadó ACK csomagja érkezik meg hamarabb a célohoz, akkor az lesz továbbiakban a megbízható forrás.



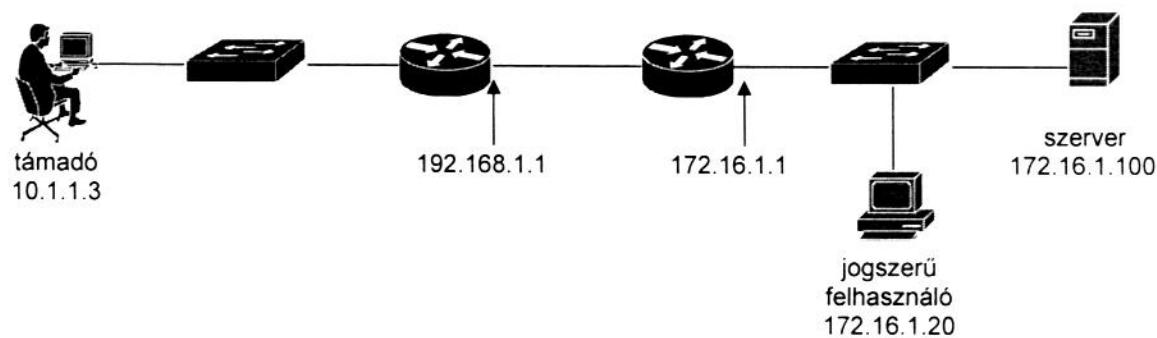
A képen látható két szerver kommunikációját eltérítve (hijack) a támadó az adatbázis szerverre a saját adatait injektálhatja be, saját tranzakciót rögzítheti le.

A TCP szekvencia szám meghatározásának két módja:

- **non-blind spoofing:** ha a támadó és a cél ugyanazon a hálózaton van, így egy packet capture szoftverrel képes a szekvencia számokat kinyerni.
- **blind spoofing:** a támadó és a cél nem ugyanazon a hálózaton van, így a TCP szekvencia számait sokkal nehezebb megszerezni, azonban a „IP source routing”-al erre is van lehetőség

IP source routing

A támadónak egy olyan csomagot kell küld a célnak, melyben meghatározza, hogy a válasz csomagoknak milyen útvonalat kell követniük visszafele. Ebben az útvonalban maga a támadó is benne lesz.



IP csomag source route-al:

Forrás IP	Cél IP	10.1.1.3	192.68.1.1	172.16.1.1	Adatok
172.16.1.20	172.16.1.100				

Egy source routeggal ellátott hamis IP csomaggal a támadó kényszeríteni tudja a szervert, hogy a válaszát a jogoszerű felhasználónak az IP címére a támadón keresztül küldje vissza.

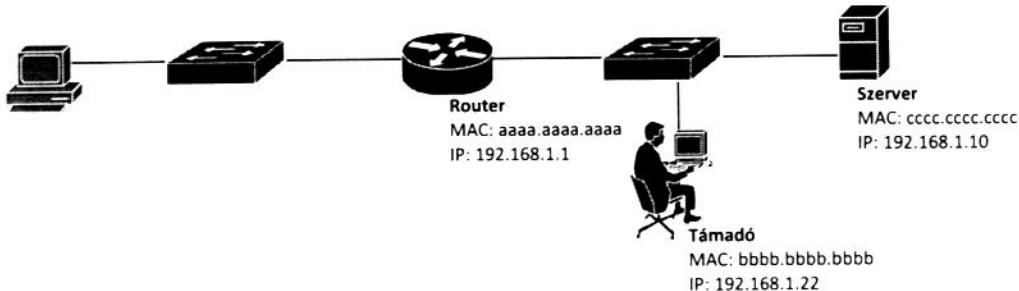
Source route opciói:

- **Loose:** a csomagnak a felsorolt IP címeken keresztül kell mennie, de más IP címek is benne lehetnek az útvonalban
- **Strict:** a csomagok csak a felsorolt IP címeken lehetnek keresztül, más IP címeken nem

Man-in-the-middle támadás

Ha a támadó a cél eszközzel egy subnetben van, akkor man-in-the-middle támadás indítására is van lehetőség. A támadó GARP üzenetek segítségével ráveszi az eszközöket, hogy rajta keresztül küldjék az adatokat.

Gratuitous ARP (GARP) üzenetek segítségével egy eszköz értesítheti a másik eszközt, hogy a MAC címe megváltozott. Persze más néven is ki lehet küldeni egy GARP üzenetet, és így az ARP táblát felülírni.



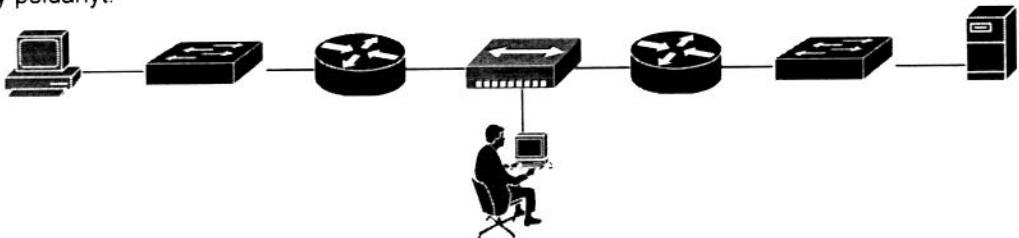
Szerver ARP táblájában a támadás előtt: 192.168.1.1 – aaaa.aaaa.aaaa (Router MAC címe)
Router ARP táblája a támadás előtt: 192.168.1.10 – cccc.cccc.cccc (Szerver MAC címe)

Támadó a routernek GARP üzenetben elküldi, hogy a 192.168.1.10 IP címhez tartozó új MAC cím bbbb.bbbb.bbbb, a szervernek pedig hogy a 192.168.1.1 IP címhez tartozó új MAC cím bbbb.bbbb.bbbb. Ezek után, ha a szerver egy csomagot akar küldeni a router felé, azt a támadó MAC címére küldi, majd a támadó az eredeti MAC címre továbbít. A válasz csomagokat a router szintén a támadó MAC címére küldi, és a támadó fogja a szervernek az eredeti címére tovább küldeni.

Az ARP táblák az alábbiak lesznek a GARP üzenetek után

Szerver ARP táblájában a támadás után: 192.168.1.1 – bbbb.bbbb.bbbb (Támadó MAC címe)
Router ARP táblája a támadás előtt: 192.168.1.10 – bbbb.bbbb.bbbb (Támadó MAC címe)

Másik lehetőség a man-in-the-middle támadásra, ha a támadó egy HUB-ot helyez el a hálózaton, vagy a switch SPAN (Switch port analyser) portjára tud kapcsolódni, így minden a switchen keresztülhaladó keretből kap egy példányt.

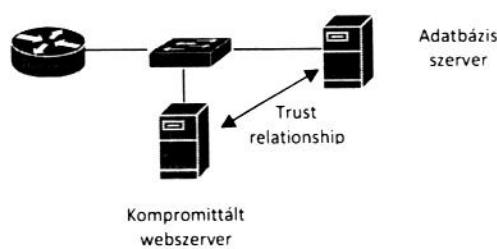


Védekezés az IP spoofing támadások ellen:

- **ACL-ek:** megszabhatjuk, hogy egy adott interfészben milyen bejövő forrás IP címek várhatóak. Ha egy router interfészén a 192.168.1.0/24-es hálózat van csak, akkor minden más forrás IP címmel bejövő csomag eldobható.
- **Titkositás:** man-in-the-middle támadásoknál a támadó ugyan megkapja az adatokat, azonban azok a számára értelmezhetetlenek, módosítás esetén az esetleges ellenőrző összegeket sem tudja újraszámítani.

Trust relationship támadás

Webszerver és az adatbázisszerver közötti „bizalmat” kihasználva a támadó bizalmas információkhöz juthat, mint például felhasználónevek-jelszavak, bankkártya információk, személyes adatok, stb. A támadó a webszerver gyengeségét kihasználva feltörí azt, majd onnan férhet hozzá az adatbázis szerverhez.



Támadások előkészületei

Élő hosztok és szolgáltatásaiak felderítése

- Ping sweep:** hálózaton levő IP címeket pingelve, vagy a hálózat broadcast címét pingelve, az élő hosztok választ küldenek, ezáltal megtudható, hogy milyen címekre érdemes támadásokat indítani
- Port scan:** az egyes hosztokon levő szolgáltatások (http, ftp, dns, smtp, stb...) felderítése, mely a használt operációs rendszer és az operációs rendszer sebezhetőségeiről is árulkodhat.

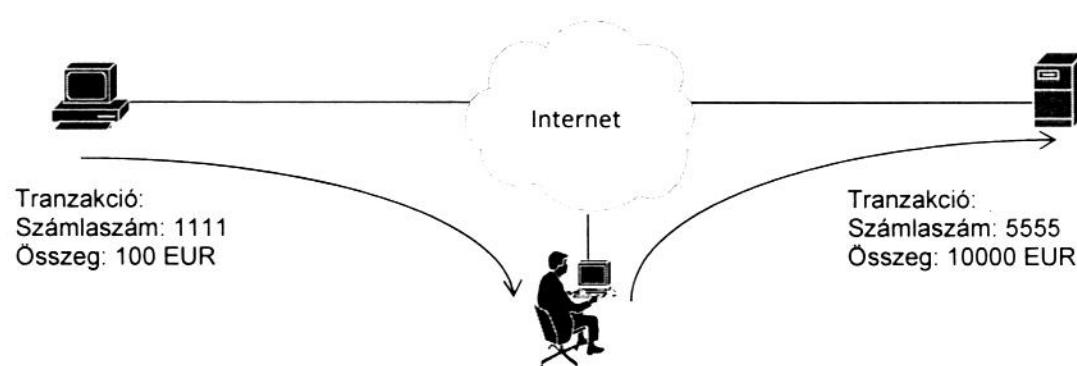
Social enginiering: bizalmas információk kiszedése személyektől

- phising:** hamis weboldalak, emailek bizalmas információ közlésére akarnak rávenni
- vishing:** telefonon keresztül bizalmas információra akarnak rávenni. Pl. telefonthívás alkalmával a támadó a bank vagy az informatikai részleg alkalmazottjának adja ki magát, és a bankkártya adatainkra vagy a felhasználói nevünkre és jelszavunkra kíváncsiak.

Backdoor: telepített alkalmazáson keresztül a biztonsági szabályok figyelmen kívül hagyásával, lehetővé teszi a kívülről történő behatolást (trójai)

Adatok elleni támadás

Ez esetben az adatokat a támadó az adatátvitel során próbálja módosítani. Banki tranzakció esetén a kliens egy átutalást küld 100 Euróról, azonban a bank már 10000 Euróról és más számlaszámra történő utalásról kapja meg a tranzakciót. Természetesen minden más támadás amely során az adatokat meg akarják változtatni ebbe a kategóriába sorolható.



Jelszavak lopása

- Trójai: alkalmazás mely hasznosnak tűnik, azonban gyűjt a beírt adatokat, így a jelszavakat is (pl mászkáló cica, szép képernyőkímélő)
- Keylogger: háttérben futó alkalmazás, mely a leütött billentyűket, a beírt adatokat gyűjt
- Brute force: összes lehetséges betükombináció végigpróbálása
- Jelszó kitalálása szótár/szószedet alapján (Dictionary attack)
- Packet capture: alkalmazás, mely a hálózaton keresztülmenő forgalmat gyűjt (pl Wireshark)

Bizalmasság elleni támadás

Leggyakrabban detektálhatlan marad, mivel csupán a felhasználó bizalmas adatainak (személyes adatok, jelszavak, bankkártya számok) megszerzésére irányul, nem módosít semmit az adatokon.

Botnet

Vírusral fertözött számítógépek egy helyről irányíthatóak. A gépek a „zombi” gépek, és robotként viselkednek. A botnet működtetője, egyszerre képes az összes hálózaton levő „gépének” a működtetésére. Felhasználási területei közé az email (spam) küldése és dDoS támadások indítása tartozik.

Kliensek biztonsága

A felhasználók oldaláról számos veszély jöhet. Ezért az alábbi irányelveket érdemesbetartani:

- **least privilege concept:** a lehető legkevesebb jogosultsággal rendelkezzenek a felhasználók, ami a munkájuk elvégzéséhez elegendő.
- **rendszeres update-ek:** frissítésekkel be lehet foltozni az alkalmazások és az operációs rendszer hibáit

Felhasználók oldaláról alkalmazott támadások:

- **backdoor:** biztonsági előírások átugrására irányuló támadás, ehhez szükséges alkalmazás telepítése, port megnyitása
- **buffer overflow (túlcordulás):** nem megfelelően formázott, speciális karaktereket tartalmazó, vagy túl hosszú input-ot adva egy programnak, az a normális végrehajtástól eltérhet, és az input-ban szereplő kódot hajthatja végre.
- **férgék (worms):** kártékony kódok, melyek képesek replikálni magukat, és terjedni felhasználói beavatkozás nélkül is, az operációs rendszer vagy más komponensek hibáinak kihasználásával
- **vírusok:** férgékhez hasonlóan kártékony kódok, azonban felhasználói beavatkozást igényelnek (kattintás, szoftverindítás, email megnyitás)
- **trójaiak (trojans):** hasznosnak látszó programok, azonban a felhasználó tudta kívül mást is csinálnak a háttérben. Akár billetpenz leütést figyelhetik, email-eket küldhetnek, vagy backdoor-t biztosíthatnak a felhasználó gépére.
- **privilege escalation:** egy másik szolgáltatás kompromittálása után, a sikeresen komromittált szolgáltatáson keresztül, annak jogával való támadás intézése a cél felé
- **footprint analysis:** információ gyűjtése (operációs rendszerek, szoftverek, domain nevek, IP címek, stb...)

Port-scan és ping-sweep alkalmazása

Korábban szó esett a kettőről. Alkalmazásuk az alábbiakra irányul:

- **port-scan**
 - operációs rendszer azonosítása
 - sebezhetőségek megállapítása
 - aktiv szolgáltatások azonosítása
- **ping-sweep**
 - élő hosztok megállapítása a hálózaton
 - eszközök azonosítása

Cisco Security Agent

A klienseken futó alkalmazás, mely az operációs rendszer biztonságát hivatott figyelni. Részei az alábbiak:

- **file system interceptor:** írási/olvasási műveleteket a policy-től függően engedélyezi vagy tiltja
- **network interceptor:** hálózati kapcsolatokat engedélyezi vagy tiltja
- **configuration interceptor:** konfigurációs módosításokat (registry, rc fájlok) engedélyezi vagy tiltja
- **execution space interceptor:** memóriaműveleteket engedélyezi vagy tiltja

HIPS (Host Intrusion Prevention System)

Kliensekre telepített, azokon futó biztonsági alkalmazás. Tipikusan vírusvédelmi és tüzfal és rendszerek. Nem feltétlen szükséges minden eszközre feltelepíteni. Sokkal hatásosabb ha a patchek telepítve vannak, nem szükséges szolgáltatások és portok le vannak tiltva, és erős, rendszeresen megújítandó jelszavakat használunk.

Email védelem: Cisco Ironport és Senderbase

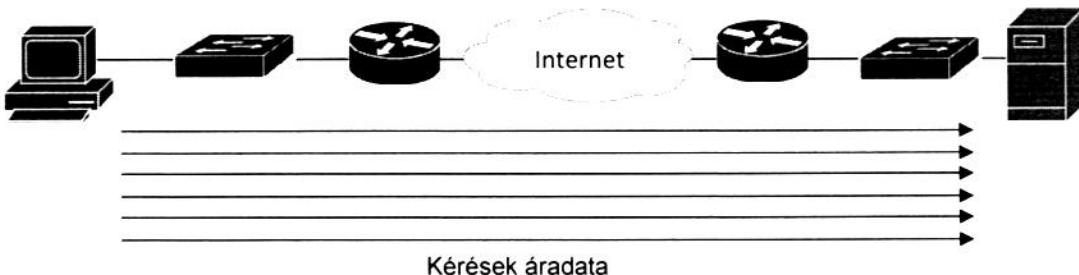
Senderbase a világ elsőszámú email forgalmat figyelő alkalmazása. A Cisco Ironport a Senderbase-t használja az e-mail forgalom figyelésére.

Rendelkezésre állás elleni támadások

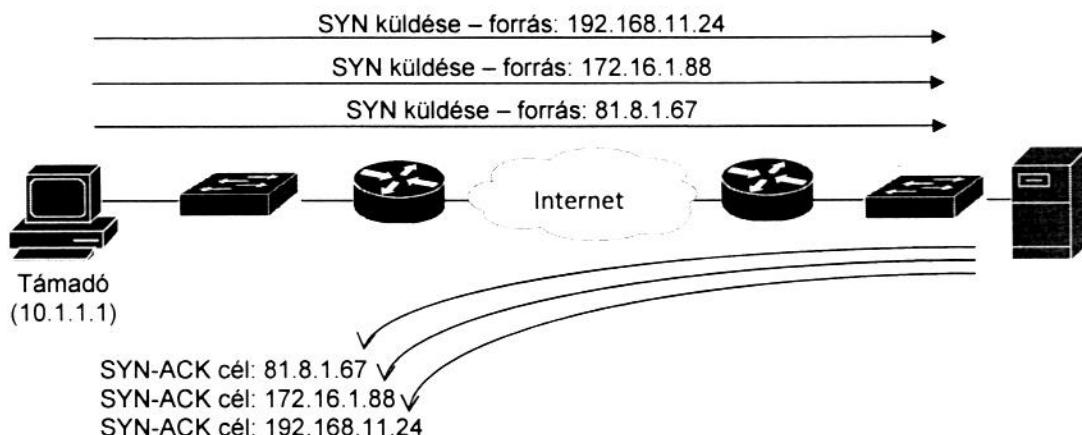
A támadó a rendszer elérhetőségét, használhatóságát próbálja korlátozni. Például ha a támadó el tudja érni, hogy a támadott rendszer memória és processzor kihasználtsága maximális legyen, akkor további kéréseket nem fog tudni kiszolgálni, így a legitim felhasználók nem lesznek képesek használni a rendszert. Kapcsolatok maximális számának megszabásával kiküszöböltető.

Denial of Service (DoS): adatok áradatával próbálja a támadó a szervert túlterhelni

Distributed DoS (dDoS): egyszerre több forrásból történő DoS támadás (Botnet hálózatok használatával)



TCP SYN flood: a támadó számos TCP kapcsolatot vesz fel a szerverrel, hamis forrás IP címmel, így azonban a TCP 3-way-handshake nem kerül sosem befejezésre, mivel az eszközök, akik kapják a szervertől a SYN-ACK csomagot nem kértek semmit a szervertől és nem is reagálnak rá semmit. A szerver egy bizonyos ideig várakozik, hogy a TCP handshake végbemenjen. Bizonyos számú kapcsolat után, újabb kapcsolat fogadására nem lesz képes, vagy akár az alkalmazás összeomlásához is vezethet.

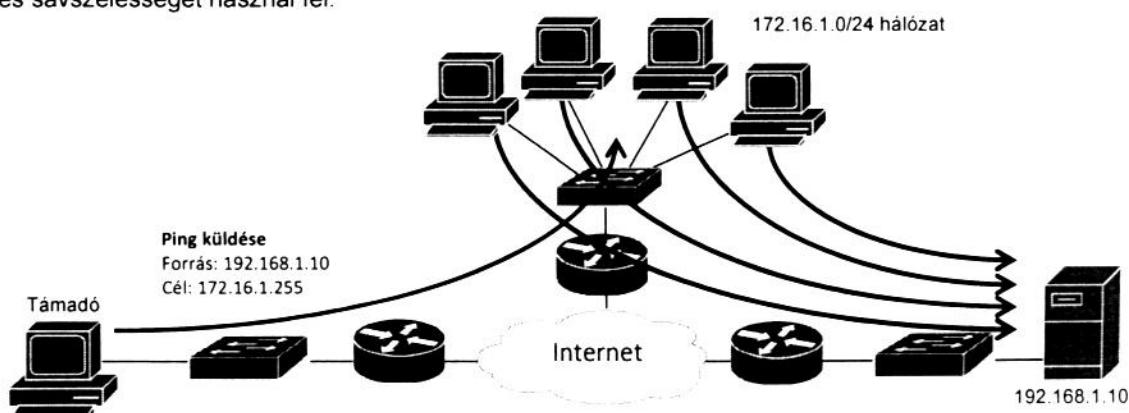


ICMP támadások

Számos hálózaton ICMP engedélyezve van, és hibaelhárításnál igen hasznos, azonban további támadási felületet is biztosít.

„Ping of Death” néven ismert támadásban a támadó túl nagy ping csomagot küld, melyet a célpont nem tud megfelelően feldolgozni és összeomlik.

Smurf attack: a támadó forrásként a támadni kívánt szerver IP címét adja meg, és egy subnet címére küld ping kéréseket, akkor a subneten levő összes hoszt a szervernek küldi vissza a ping választ, ezáltal azon CPU és sávszélességet használ fel.



Hálózati biztonság kiértékelése

- aktiv IP címek és nyitott portok keresése
- aktiv hosztokon ismert sebezhetőségek keresése
- jelszótörő alkalmazások futtatása
- log file-ok analizálása
- viruskereső futtatása
- nem biztonságos wifi hálózatok keresése

Biztonságos hálózat további jellemzői

- naprakész frissítések
- felhasználók értesítése a social engineeringről és egyéb veszélyekről
- dinamikus biztonsági szabály alkalmazása
- nem használt portok letiltása

Ezekkel a jellemzőkkel egy biztonságos hálózati platform építhető.

Üzleti érdekek (Business needs): legfontosabb amit szem előtt kell tartani a biztonságnál is.

Számos IOS alapú eszköz rendelkezik különböző biztonsági funkcióval, IOS és hardveres támogatással bővíthetők, azonban a Cisco ASA5500 szériájú eszköze kombinálja egybe a legtöbb lehetőséget:

- firewall
- IPS
- VPN
- antispyware-filter (DVS – dynamic vector streaming technológiával)
- antivirus
- antiphising

Host és hálózat alapú védelem

- **host based:** számítógépekre antivirus és tüzfal szoftvert telepítve fokozhatjuk a biztonságát a hálózatnak, azonban a skálázhatósága nem a legmegfelelőbb. A számítógépről induló minden adat ellenőrzésére használható.
- **network based:** hálózati eszközökön (router, firewall) elhelyezett antivirus és tüzfal funkciók biztonsági funkciókat nyújtanak anélkül, hogy a számítógépekre bármilyen szoftvert kellene telepíteni. Ez azonban SSL és TLS tunelek forgalmát nem képesek ellenőrizni. Inline védelemnek is nevezik.

Működési biztonság (Operations security): biztosítja, hogy nem lesz olyan pontja a hálózatnak, mely a teljes biztonságot veszélyeztetheti, se személyi, se pedig tárgyi eszköz téren. Magába foglalja az alábbiakat:

Cisco eszközök hozzáférés módjai

- Telnet/SSH (vty 0 4)
- Console (con 0)
- modemes / Aux (aux 0)
- HTTP/HTTPS (SDM)
- SNMP (Simple Network Management Protokol)

Biztonság fokozásának lépése egy erős jelszó választása, mely jó ellenáll egy brute-force vagy dictionary támadásnak. Kis- és nagybetűket, számokat, speciális karaktereket tartalmazó, hosszú jelszó, amely nem asszociálható a felhasználóhoz. A jelszó rendszeres cseréjének megkövetelése is fontos.

Minimális jelszó hosszúság az alábbi parancssal követelhető meg, globális konfigurációs módból:

```
security password min-length 10
```

IOS által használt jelszavak

enable password – Cisco algoritmusával van kódolva, ha a **service password-encryption** be van kapcsolva

enable secret – MD5-el titkosított jelszó

line con 0

password – konzolporton történő kapcsolatkor kérte jelszó

line vty 0 4

password – Telnet és SSH kapcsolódáskor kérte jelszó

line aux

password – aux porton történő kapcsolódáskor kérte jelszó

A login parancsot ne feledjük el kiadni, különben a bejelentkezések nem lesznek engedélyezve.

Ha az **enable password** és az **enable secret** is konfigurálva van, akkor csak az **enable secret** jelszó az érvényes.

Az **enable password** kompatibilitási okokból maradt meg, használata nem ajánlott.

Ha a „**service password-encryption**” nincs bekapsolva, akkor az **enable secret** jelszón kívül minden jelszó olvasható, **clear-text** formátumban tárolódik a konfigurációban.

Routerre kapcsolódva a legelső authentikáció, amit a router kér a kapcsolódástól függően a console, vty vagy aux jelszó. Ennek ismeretében user exec módba kerülünk, ahol nem minden parancs érhető el.

Ebből az **enable** parancssal lehet továbblépni globális konfigurációs módba, ami az **enable secret** vagy **enable password** jelszót kéri.

Fizikai hozzáférés esetén ROMMON módban lehetőség van a konfigurációs regiszter átállítására, hogy a router a startup-config-ot átugorja következő induláskor.

A ROMMON mód a „**no service password-recovery**” parancssal „kikapsolható”. A konfigurációs fájl nem másolható vissza.

Hibás bejelentkezések kezelése

Alapértelmezésben tíz hibás bejelentkezést követően, az eszköz nem enged 15 másodpercig újabb bejelentkezési kísérletet.

Ennek értéke a „**security authentication failure rate szám log**” opcionál megváltoztatható, és elérésekor syslog üzenet is generálódik.

Az érték 8-ra az alábbi módon változtatható meg:

```
Router(config)# security authentication failure rate 8 log
```

Inaktivitás idejének megszabása

Ha az adminisztrátor hosszabb időre elmegy, és otthagya a router konzolját bejelentkezve, komoly biztonsági problémákat vethet fel. Ha a router bizonyos ideig nem érzékel felhasználói aktivitást, akkor automatikusan megszakítja a kapcsolatot. Ez alapértelmezés szerint tíz perc. Megváltoztatni a vty con és aux-on lehet az **exec-timeout** parancssal. Példánkban 1 perc 45 másodpercre állítjuk (0 0-ra állítva, nincs inaktivitás figyelés)

```
Router(config-line)# exec-timeout 1 45
```

Privilégium szintek

Bizonyos felhasználók csak bizonyos parancsok kiadására lehetnek jogosultak, még mások több parancs vagy akár konfigurációs szintű parancsokat is kiadhatnak.

Cisco IOS privilégium szintekból 16-ot támogat (0-15). User exec mód 0-s szintnek felel meg, még az enable parancs kiadása után a 15-ös szint érhető el. A közte levő szintek konfigurálhatóak.

Szintenként külön jelszó állítható be, és megadható az adott szinten elérhető parancsok lista is.

A szinteken elérhetők az alacsonyabb szintek parancsai is.

Az elérhető parancsok és a szinthez tartozó jelszó az alábbi módon konfigurálható:

```
Router(config)# privilege exec level 5 debug  
Router(config)# enable secret level 5 otosjelszo
```

Példánkban a **debug** parancs lesz elérhető 5-ös szinten.

CLI view

Feladat alapú (Role-based) CLI view használatánál csak az adott view alá definiált parancsok érhetőek el a felhasználó számára. CLI view definiálásához az aaa-t engedélyezni kell az „**aaa new-model**” parancssal. Ezután a „root view”-t kell engedélyezni az „**enable view**” parancssal és az enable jelszót kell megadni. View létrehozása a „**parser view név**” parancssal lehetséges, mely alatt a view-hoz tartozó jelszó és a rendelkezésre álló parancsok adhatóak meg. Jelszó a „**secret 0 jelszo**” parancssal adható meg, ahol a „0” a jelszó clear-text formátumára utal. Az egyes parancsok a „**commands exec include parancs**” konfigurációs parancssal adhatóak meg. CLI view-be az „**enable view nev**” parancssal lehetséges.

```
Router(config)# aaa new-model  
Router(config)# end  
Router# enable view  
  
Password:  
  
Router# conf t  
Router(config)# parser view TEST  
Router(config-view)# secret 0 testjelszo  
Router(config-view)# commands exec include traceroute  
Router(config-view)# commands exec include ping  
Router(config-view)# end  
Router# enable view TEST  
  
Password:  
Router#
```

CLI view-nél csak az adott view-be konfigurált parancsok érhetőek el, még privilégium szinteknél az adott szintnél konfigurált és az összes alatta levő szint parancsai elérhetőek.

Telnet/SSH/HTTP elleni támadások védelem

A támadások irányulhatnak a router ellen, ha a támadó megpróbálja brute-force módon kitalálni a jelszót, vagy a router erőforrásait felemészteni. Ezt az alábbi eszközökkel lehet kiküszöbölni:

- bejelentkezési próbák közötti várakozási idő
- bejelentkezés felfüggesztése egy támadás esetén
- syslog üzenet generálása egy sikeres vagy sikertelen kísérlet esetén

A funkciót a „**login block-for**” parancssal lehet bekapcsolni, melynek paraméterei:

```
Router(config)# login block-for másodperc attempts kísérletek within másodperc
```

- **block-for másodperc**: bejelentkezések tiltásának ideje, **quiet-period**-nak is nevezik.
- **attempts kísérletek**: sikertelen bejelentkezések száma, ami után a bejelentkezések tiltva lesznek
- **within másodperc**: az az idő, amin belül a sikertelen bejelentkezéseknek meg kell történniük a tiltáshoz

Az alábbi parancs a bejelentkezést 60 másodpercre tiltja, ha 40 másodpercen belül 2 sikertelen bejelentkezés történik:

```
Router(config)# login block-for 60 attempts 2 within 40
```

Sikertelen bejelentkezést követően az alábbi log üzenetet adja:

```
*Nov  8 18:02:40.051: %SEC_LOGIN-1-QUIET_MODE_ON: Still timeleft for watching failures is 37 secs, [user: ] [Source: 192.168.1.2] [localport: 23] [Reason: Login Authentication Failed] [ACL: sl_def_acl] at 18:02:40 UTC Mon Nov 8 2010
```

A quiet-period letelte után pedig az alábbi üzenet tájékoztat arról, hogy a bejelentkezések újra lehetségesek:

```
*Nov  8 18:03:40.051: %SEC_LOGIN-5-QUIET_MODE_OFF: Quiet Mode is OFF, because block period timed out at 18:03:40 UTC Mon Nov 8 2010
```

A bejelentkezések tiltásának ellenőrzéséhez a „**show login**” parancs használható. Az alábbi példában a bejelentkezések még 55 másodpercig tiltva vannak:

```
R5# sh login
      A default login delay of 1 seconds is applied.
      No Quiet-Mode access list has been configured.

      Router enabled to watch for login Attacks.
      If more than 2 login failures occur in 50 seconds or less,
      logins will be disabled for 60 seconds.

      Router presently in Quiet-Mode.
      Will remain in Quiet-Mode for 55 seconds.
      Denying logins from all sources.
```

R5#

További lehetőségek a bejelentkezések szabályozására:

- | | |
|--|---|
| • Kivételek hozzáadása: | login quiet-mode access-class [acl-név acl-szám] |
| • Bejelentkezések közötti idő: | login delay másodperc |
| • Log generálása hibás bejelentkezéskor: | login on-failure log [every próbálkozások] |
| • Log generálása sikeres bejelentkezéskor: | login on-success log [every próbálkozások] |

A 192.168.1.0/24 hálózatra a bejelentkezések szabályozása nem terjed ki az alábbi példában illetve még néhány paraméter is megváltoztatásra kerül:

```
Router(config)# access-list 1 permit 192.168.1.0 0.0.0.255
Router(config)# login quiet-mode access-class 1
Router(config)# login block-for 60 attempts 2 within 40
Router(config)# login delay 3
```

A parancs csak az **aaa new-model** használata esetén működik.

Figyelemfelkeltő banner üzenetek

Mindenképp érdemes figyelemfelkeltő üzenetben közölni, hogy a router aktív monitorozás alatt áll, és a betörlési kísérletek ellen akár jogilag is fellép a router üzemeltetője, tulajdonosa.

A Cisco IOS routereken az alábbi banner üzeneteket lehet konfigurálni:

- **Message of the day (MOTD):** minden konzolablakon megjeleníti bejelentkezést
- **Login banner:** minden bejelentkezéskor megjeleníti
- **Exec banner:** sikeres bejelentkezést követően jeleníti meg

A bannerek kezdetét és végét a konfigurációnál kell megadni. Példában a ^C jelzi a banner kezdetét és végét is:

```
Router(config)# banner motd ^C
Message of the day: banner motd parancsal.
^C
Router(config)# banner login ^C
Login banner: banner login parancsal.
^C
Router(config)# banner exec ^C
Exec banner: banner exec parancsal.
^C
```

Fájlok védelme

A támadás irányulhat a Cisco IOS és a konfigurációs állomány törlésére is. A konfigurációs fájlról egy biztonsági másolatot készíthetünk, illetve az IOS-t elrejthetjük a könyvtálistázáskor az alábbi parancsokkal:

```
Router(config)# secure boot-image
Router(config)# secure boot-config
```

Működését a „show secure bootset” parancssal lehet ellenörizni.

Cisco Self-Defending Network jellemzői

- collaborative: hálózat egyes eszközei képesek együttműködni másik eszközzel
- integrated: a biztonsági funkciók be vannak építve az eszközökbe
- adaptive: új biztonsági veszélyek könnyen adaptálhatók

Konfiguráció SDM (Security Device Manager)

SDM segítségével webes felületen van lehetőség a routert konfigurálni. Az SDM telepíthető a számítógépre, vagy a routerre, vagy minden helyre. Ezt a telepítő a telepítés során kérdezi meg, hogy hova szeretnénk telepíteni.

SDM futtatásához a routeren az alábbiak szükségesek:

- lokális felhasználó konfigurálása (*username felhasznalo privilege 15 secret jelszo*)
- http szerver engedélyezése (*ip http server*)
- lokális http authentikáció konfigurálása
 - aaa new-model-en keresztül


```
Router(config)# aaa new-model
          Router(config)# aaa authentication login default local
```
 - http local authentikáció keresztül


```
Router(config)# ip http authentication local
```

A nyitó képernyön egy általános összefoglalás található a routerről. Konfiguráció és a konfiguráció megtekintése a **Configure** alatt lehetséges.

The screenshot shows the SDM interface for a Cisco 2621XM router. The top navigation bar includes File, Edit, View, Tools, Help, Home, Configure, Monitor, Refresh, Save, Search, and Help buttons. The main window has two main sections:

About Your Router (Left):

Hardware		Software	
Model Type:	Cisco 2621XM	IOS Version:	12.4(13d)
Available / Total Memory(MB):	51/128 MB	SDM Version:	2.5
Total Flash Capacity:	16 MB		

Configuration Overview (Right):

Interfaces and Connections		Up (1)	Down (1)
Total Supported LAN:	1	Total Supported WAN:	1(10/100Ethernet)
Configured LAN Interface:	0	Total WAN Connections:	1
DHCP Server:	Not Configured	No. of DHCP Clients:	0
DHCP Pool:	Not Configured		

Below this is a table for interfaces:

Interface	Type	IP/Mask	Description
FastEthernet0/0	10/100Ethernet	DHCP client	
FastEthernet0/1	10/100Ethernet	no ip address	

Routing (Bottom):

No. of Static Route:	0
Dynamic Routing Protocols:	None

SDM nyitóképernyő

Configure gombra kattintva, baloldalon egy újabb menüoszlop jelenik meg, mely a konfigurációs feladatokat kategorizálva tartalmazza.

Ebben a fejezetben tárgyalunk az **Additional tasks** menü alatt lehet elérni, amely az utolsó az oszlopban.

AAA – helyi adatbázis használatával

Az AAA elnevezés mögött az alábbiak állnak:

- **authentication:** ki az a felhasználó, aki be akar lépni a routerre, és beléphet-e
- **authorization:** mi az amit csinálhat a felhasználó
- **accounting:** mi az, amit csinált a felhasználó

Az aaa-t globálisan az **aaa new-model** parancssal lehet bekapcsolni. Ezután az authentikáció metódus listát (meghod list) lehet definiálni, vagy az alapértelmezett method list konfigurációját megadni. A method list szabja meg, hogy a bejelentkezni kívánó felhasználó nevét és jelszavát hogyan és milyen módon kívánjuk leellenőrizni. Az ellenőrzés módjai az alábbiak lehetnek:

- **local:** helyi adatbázis vizsgálata
- **group tacacs+:** TACACS+ authentikáció használata
- **group radius:** RADIUS authentikáció használata
- **none:** nincs authentikáció

Lokális adatbázis használatának konfigurálása a bejelentkezéskor az alábbi parancsokkal lehetséges:

```
Router(config)# aaa new-model
Router(config)# aaa authentication login default local
```

Ha egy bejelentkezási módra nincs más konfigurálva, akkor a **default** authentikációs method list-et használja.

Az alábbi példában a router telnet illetve ssh kapcsolatok authentikálására az authVTY method list-ben definiált enable jelszót fogja kérni bejelentkezéskor:

```
Router(config)# aaa authentication login authVTY enable
Router(config)# line vty 0 4
Router(config-line)# login authentication authVTY
```

Az AAA authentikációt a „**debug aaa authentication**” parancssal lehet debugolni:

```
Username:
*Nov 21 19:35:38.355: AAA/BIND(00000006): Bind i/f
*Nov 21 19:35:38.359: AAA/AUTHEN/LOGIN (00000006): Pick method list 'default'
Username: cisco
Password: 

Router>en
Router#
*Nov 21 19:35:47.291: AAA: parse name=tty0 idb type=-1 tty=-1
*Nov 21 19:35:47.295: AAA: name=tty0 flags=0x11 type=4 shelf=0 slot=0 adapter=0 port=0
channel=0
*Nov 21 19:35:47.295: AAA/MEMORY: create_user (0x6459EC74) user='cisco' ruser='NULL' ds0=0
port='tty0' rem_addr='async' authen_type=ASCII service=ENABLE priv=15 initial_task_id='0',
vrf= (id=0)
*Nov 21 19:35:47.295: AAA/AUTHEN/START (3679405441): port='tty0' list='' action=LOGIN
service=ENABLE
*Nov 21 19:35:47.295: AAA/AUTHEN/START (3679405441): console enable - default to enable
password (if any)
*Nov 21 19:35:47.299: AAA/AUTHEN/START (3679405441): Method=ENABLE
*Nov 21 19:35:47.299: AAA/AUTHEN(3679405441): can't find any passwords
*Nov 21 19:35:47.299: AAA/AUTHEN(3679405441): Status=ERROR
*Nov 21 19:35:47.299: AAA/AUTHEN/START (3679405441): Method=NONE
*Nov 21 19:35:47.299: AAA/AUTHEN(3679405441): Status=PASS
*Nov 21 19:35:47.303: AAA/MEMORY: free_user (0x6459EC74) user='cisco' ruser='NULL'
port='tty0' rem_addr='async' authen_type=ASCII service=ENABLE priv=15 vrf= (id=0)
Router#
```

Enable authentikáció lokálisan: az „**aaa authentication enable default local**” parancssal az enable jelszó ellenőrzése a lokális felhasználó adatbázissal konfigurálható.

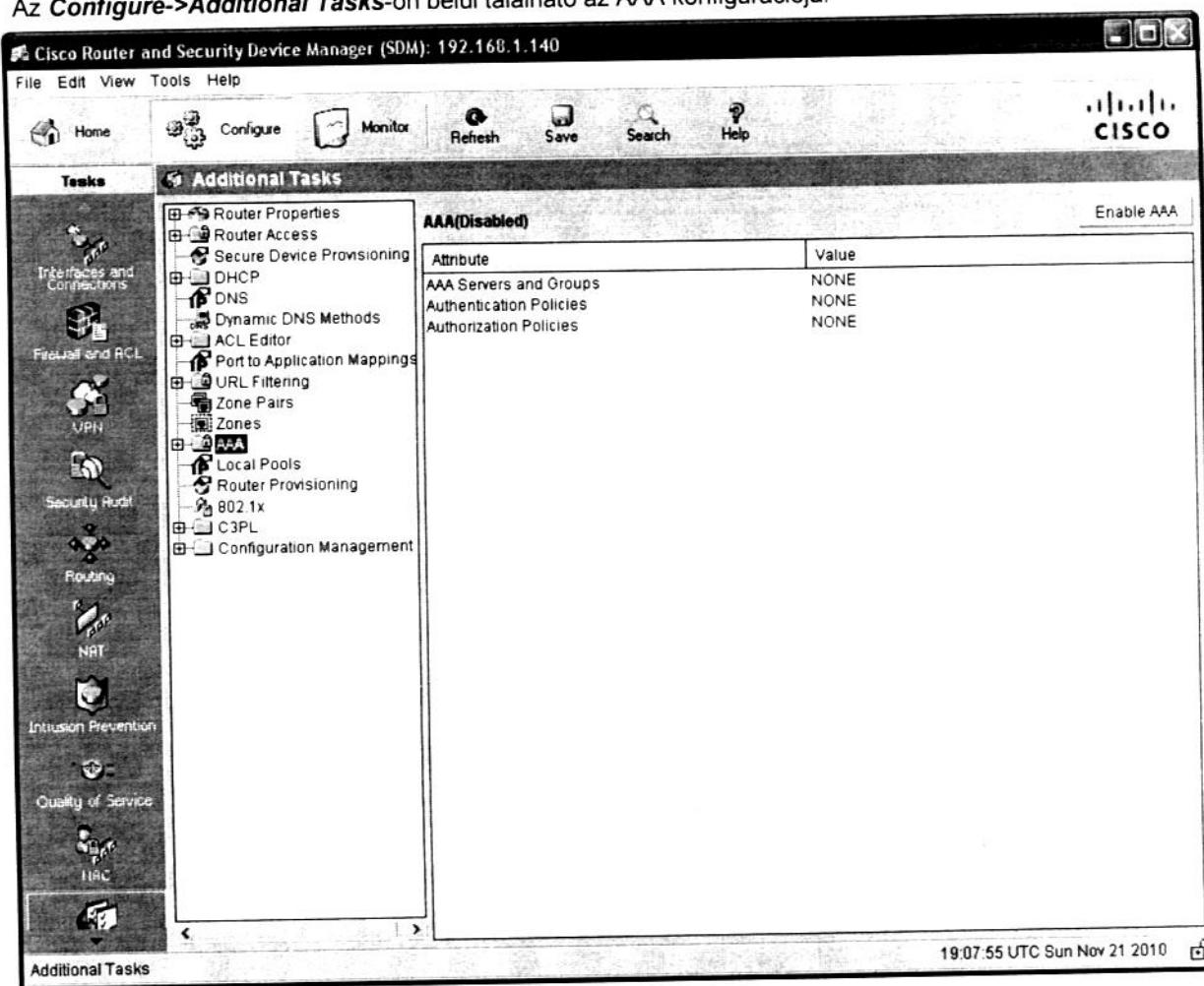
Accounting bekapcsolása:

```
Router(config)# aaa accounting exec start-stop tacacs+
```

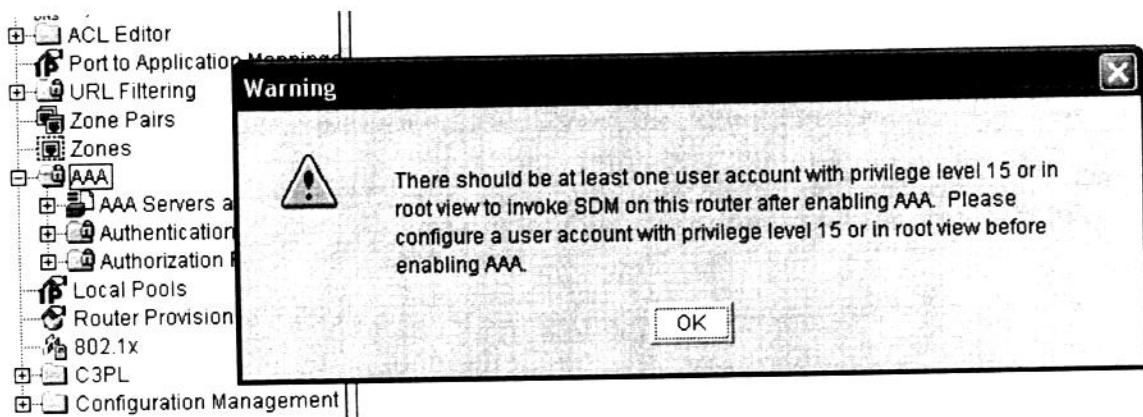
A felhasználó terminál sessionjének kezdetét és végét logolja.

AAA konfigurációja SDM-ben

Az *Configure->Additional Tasks*-on belül található az AAA konfigurációja.



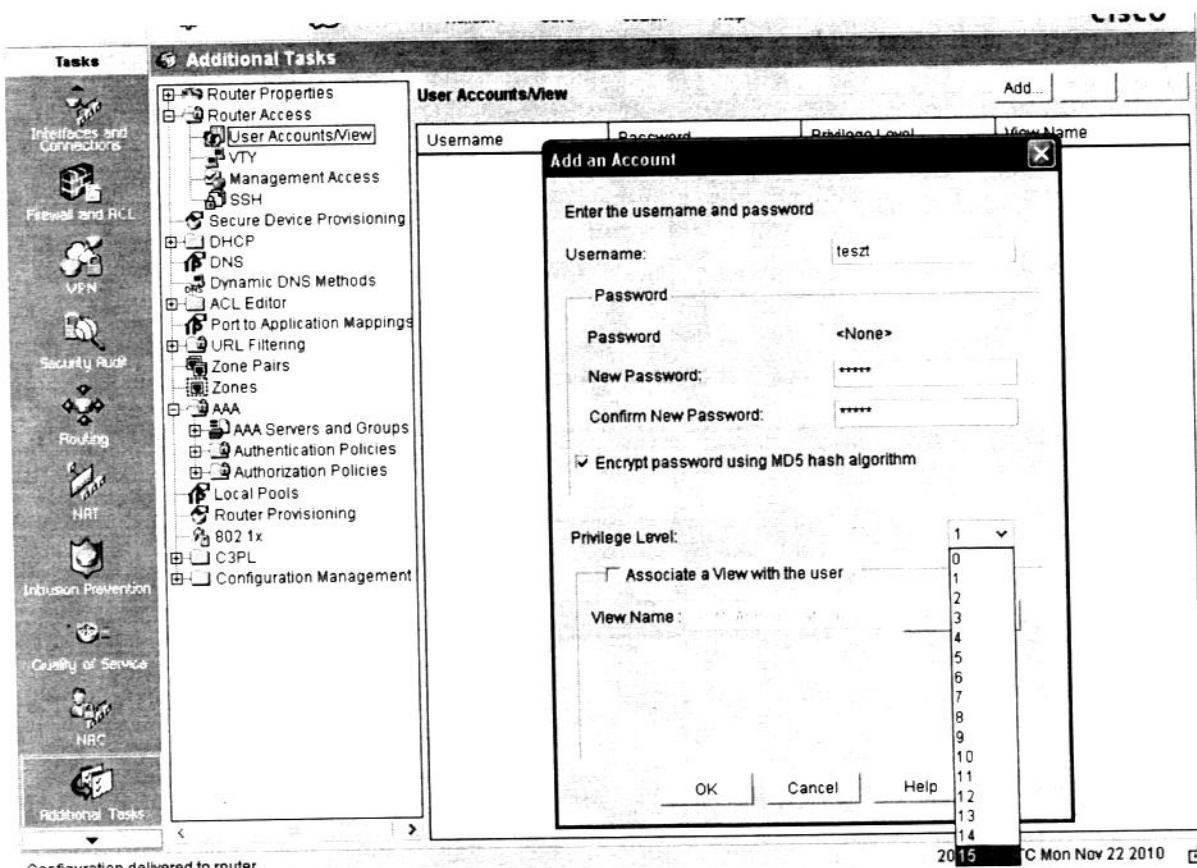
Ha az AAA még nincs engedélyezve, azt a jobb felső részen levő „**Enable AAA**” gomb megnyomásával engedélyezhetjük. Az AAA engedélyezéséhez legalább egy felhasználónak konfigurálva kell lennie a routeren. Ennek hiányát az alábbi hibaüzenet is jelzi:



Felhasználókat manuálisan az username parancssal lehet konfigurálni az alábbi módon:

```
Router(config)# username teszt privilege 15 secret jelsz0
```

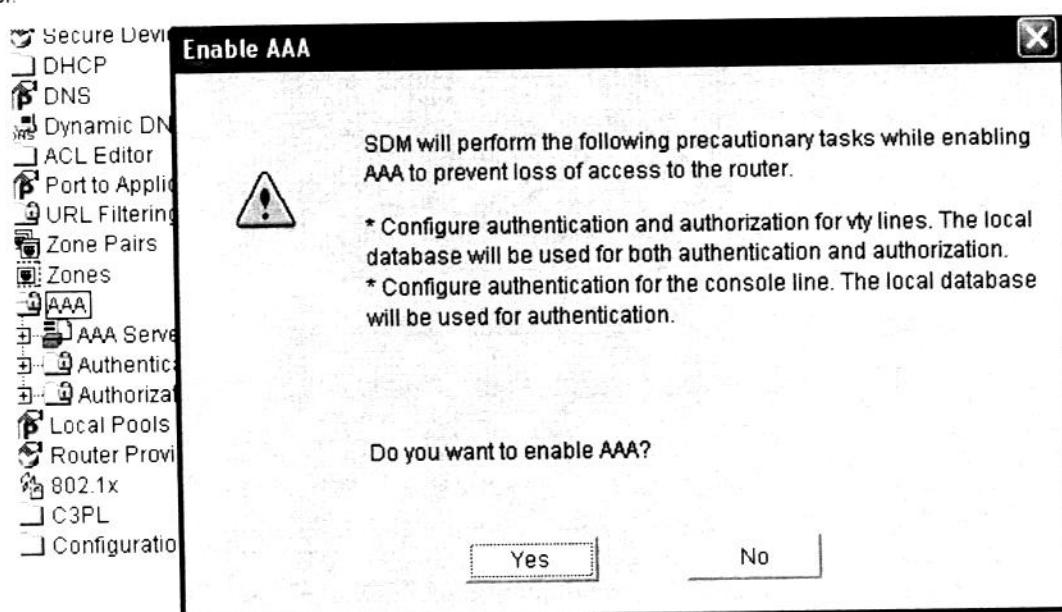
Az SDM-en keresztül a felhasználók konfigurálása az **Additional Tasks->Router Access->User Accounts/Views** alatt lehetséges. Az „Add...” gombra kattintva pedig új felhasználót adhatunk hozzá.



Configuration delivered to router.

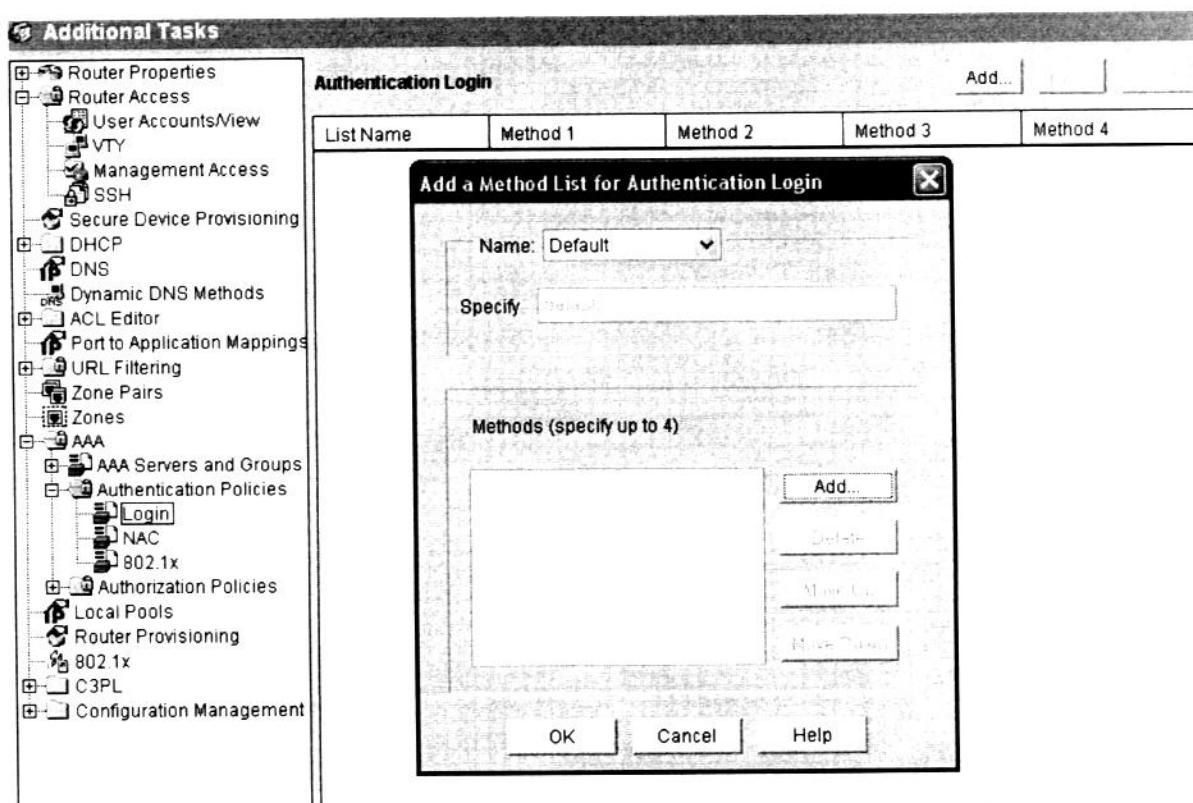
Felhasználót konfigurálása azért lényeges, mert ha az AAA login authentikációt bekapcsoljuk, azonban felhasználó nincs konfigurálva a routeren, akkor ha elveszítjük a kapcsolatot a routerrel, vagy kilépünk róla, akkor nincs felhasználó amivel be fogunk tudni jelentkezni vissza rá.

Miután beállítottunk egy felhasználót az „**Enable AAA**” gomb megnyomásával az alábbi megerősítő ablak ugrik fel:

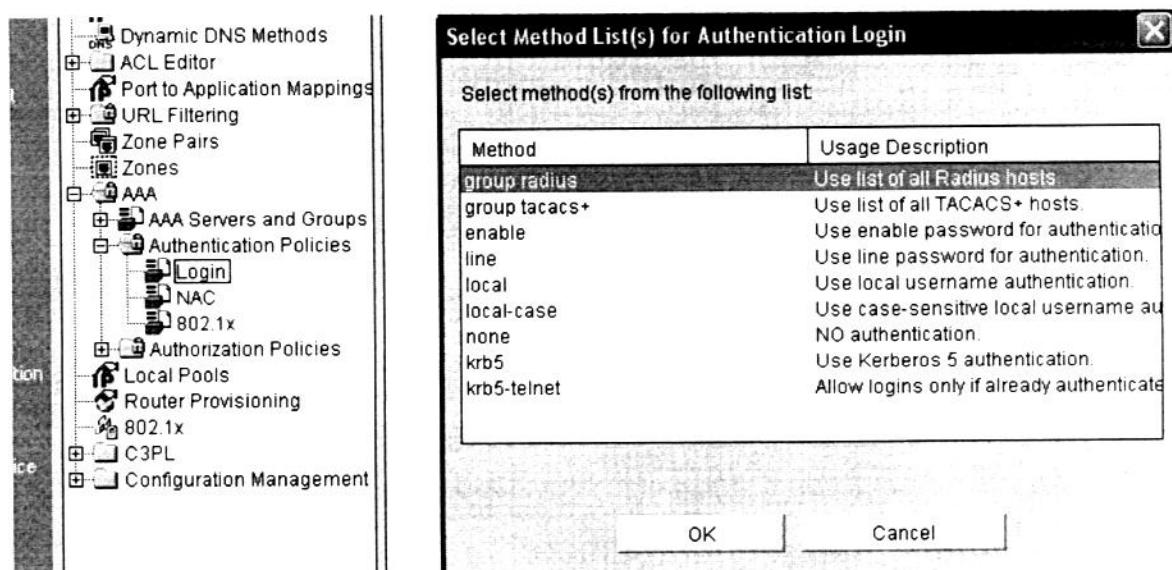


Lokális authentikáció konfigurálása SDM-ben

Az Additional Tasks->AAA->Authentication Policies->Login alatt lehet az „Add...” gombbal az alapértelmezett és egyéb authentication method list-eket konfigurálni.



A felugró ablakban szintén az „Add...” gombbal lehet a metódusokat hozzáadni:



A virtuális terminál vonalak (VTY) authentikációs beállítása az Additional Tasks->Router Access->VTY alatt állítható be.

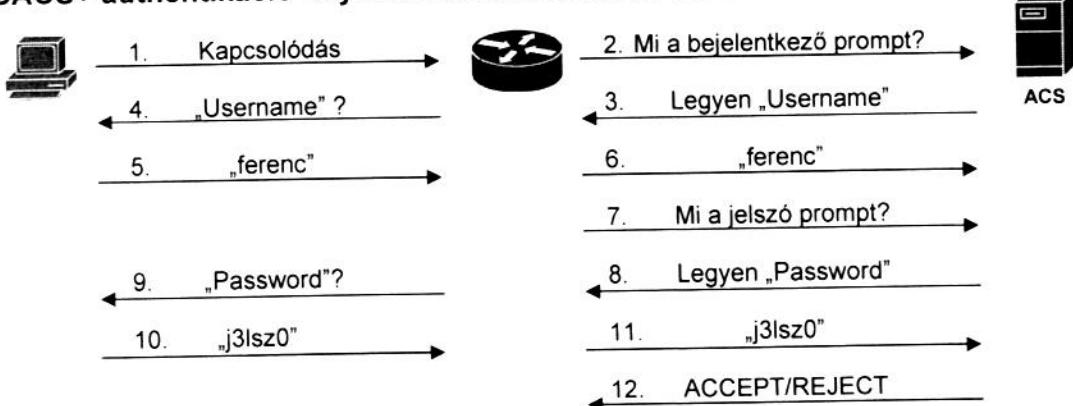
AAA RADIUS és TACACS+ authentikáció

Központositott felhasználó nyilvántartással elkerülhető az a adminisztrációs probléma, hogy a felhasználókat minden routeren konfigurálni és karbantartani kelljen. Ez esetben az authentikáció nem a routeren tárolt felhasználói név és jelszó adatbázis alapján megy végbe, hanem egy külső eszközön. Ez a külső eszköz jelez vissza a routernek, hogy a felhasználó jogosult-e belépni, vagy kiadni egy parancsot. Cisco által fejlesztett TACACS+ termék neve a **Cisco Secure ACS**.

RADIUS és TACACS+ által használt portok

Radius authentication és authorization	UDP 1645, UDP 1812
Radius accounting	UDP 1646, UDP 1813
Tacacs+	TCP 49

TACACS+ authentikáció folyamata az alábbi ábrán látható



TACACS+ üzenetek

- ACCEPT: elfogadva, a felhasználó beléphet.
- REJECT: nincs elfogadva, a felhasználó nem léphet be
- ERROR: hiba történt az authentikáció során, ez esetben a NAS általában egy másik authentikációs móddal folytatja az authentikációt
- CONTINUE: a felhasználónak további információt kell megadnia, mielőtt bejelentkezése elfogadásra, vagy elutasításra kerülne.

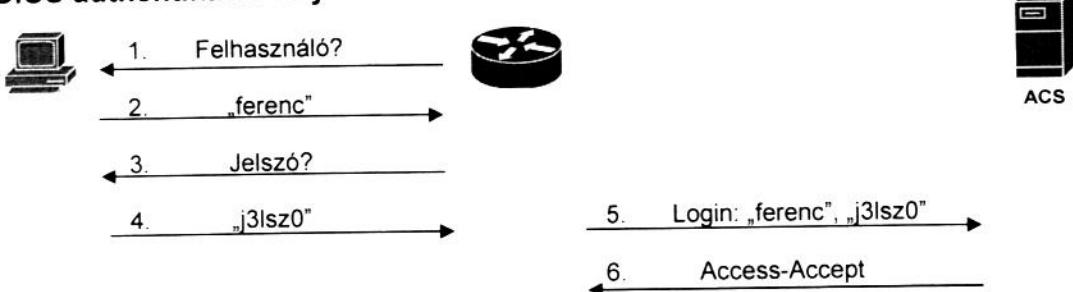
Supplicant: felhasználó, aki csatlakozni akar a hálózathoz/eszközökhöz

Authenticator/NAS: eszköz, amelyre/amelyen keresztül csatlakozni akar a felhasználó

Authentication server: az authentikációt végző szerver (TACACS+, RADIUS)

Számos esetben a tényleges authentikáció egy külső adatbázison keresztül történik meg, mint például Windows Active Directory-n keresztül.

RADIUS authentikáció folyamata az alábbi ábrán látható



RADIUS üzenet típusok

- Access-Request: RADIUS által encryptált felhasználónév-jelszó párost tartalmazza
- Access-Challenge: CHAP, MS-CHAP, EAP-MD5 authentikációnál használt
- Access-Accept: a felhasználó bejelentkezhet
- Access-Reject: a felhasználó nem jelentkezhet be

TACACS+ és RADIUS összehasonlítása

	TACACS+	RADIUS
Használt L4 protokoll	TCP	UDP
Encryptálás	az egész TACACS+ csomagot encryptálja	Csak a jelszót encryptálja

TACACS+ konfigurálása

Attól függően hogy egy vagy több TACACS+ szerver van, többféleképpen is konfigurálhatjuk.

```
Router(config)# tacacs-server host 192.168.9.100 single-connection
Router(config)# tacacs-server host 192.168.9.101
Router(config)# tacacs-server key Tj3lsz0
```

Ez esetben minden TACACS+ szerver elérésénél ugyanazt a jelszót fogja használni a router. A „single-connection” kulcsszóval egyetlen egy TCP kapcsolaton keresztül fog a router a TACACS+ szerverrel kommunikálni. Ha nem adjuk meg, akkor minden sessionhöz egy-egy új TCP kapcsolatot fog felépíteni.

A jelszavak szerverenként is megadhatóak, ahol pedig nincs konfigurálva, ott a „globális” jelszót használja a router:

```
Router(config)# tacacs-server host 192.168.9.100 key Tj3lsz0
Router(config)# tacacs-server host 192.168.9.101 key Tp@ssw0rd
Router(config)# tacacs-server host 192.168.9.102
Router(config)# tacacs-server host 192.168.9.103
Router(config)# tacacs-server key t@caCs
```

Bejelentkezéshez az authentikációt az alábbi parancsokkal adhatjuk meg:

```
Router(config)# aaa authentication login default group tacacs+ local
```

RADIUS konfigurálása

TACACS+ konfigurálásával teljesen megegyező módon, csak a „radius-server” kulcsszó használatával lehetséges.

```
Router(config)# radius-server host 192.168.9.100 key Rj3lsz0
Router(config)# radius-server host 192.168.9.101 key Rp@ssw0rd
Router(config)# radius-server host 192.168.9.102
Router(config)# radius-server host 192.168.9.103
Router(config)# radius-server key r@d1uS
```

RADIUS esetén a single-connection opció nem adható meg.

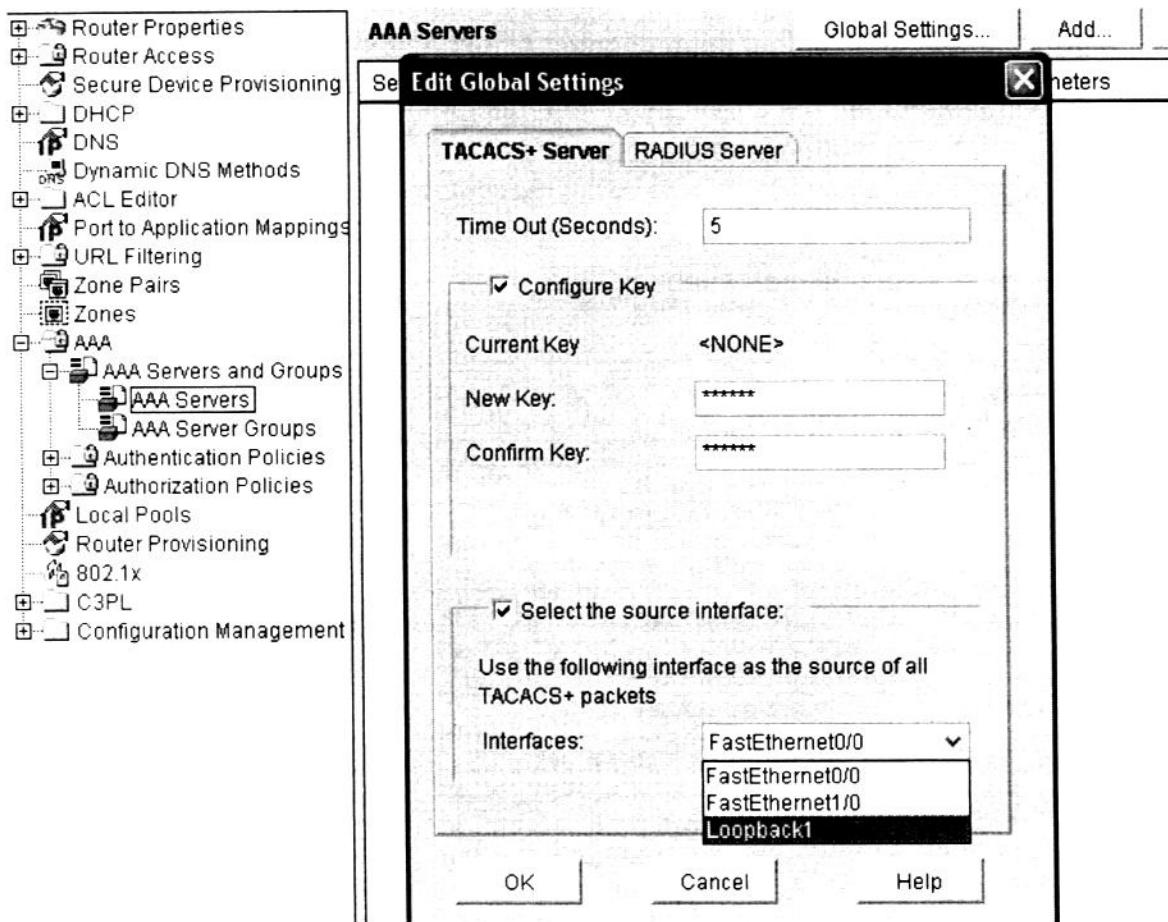
Amennyiben az alapértelmezett porttól eltérő portot szeretnénk használni a TACACS+ vagy RADIUS kapcsolódásnál, az az alábbi módon változtatható meg:

```
Router(config)# tacacs-server host 192.168.10.11 port 1001 key Tj3lsz0
```

```
Router(config)# radius-server host 192.168.10.12 auth-port 1022 acct-port 1033 key r@d1uS
```

TACACS+ és RADIUS konfigurálása SDM-ben

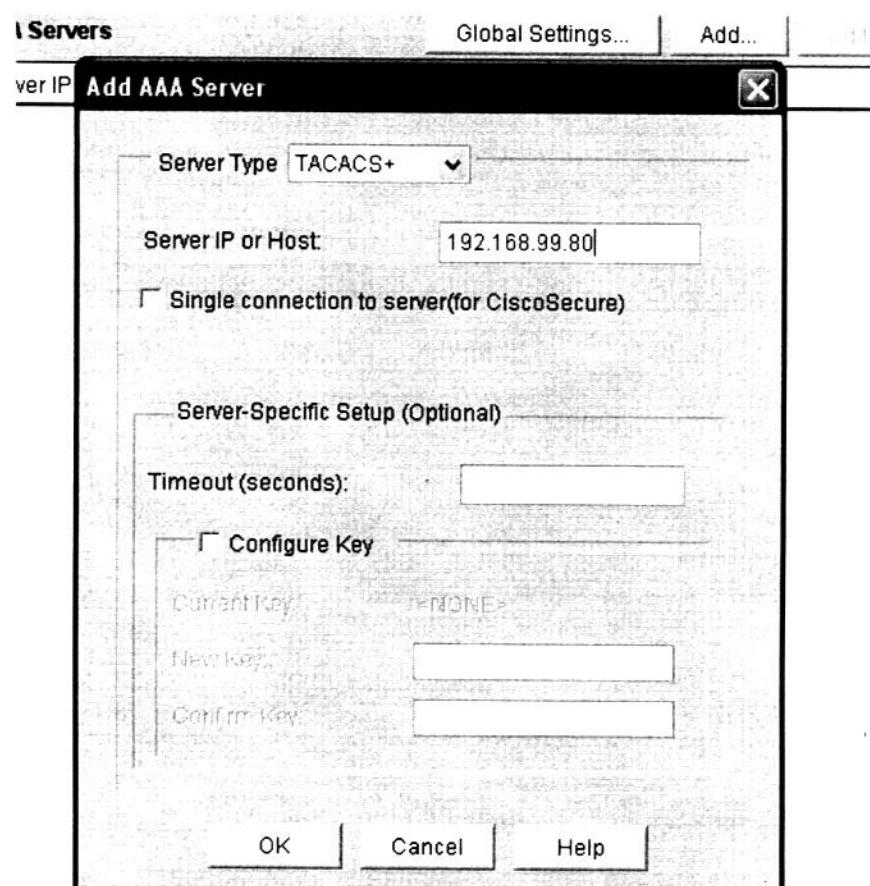
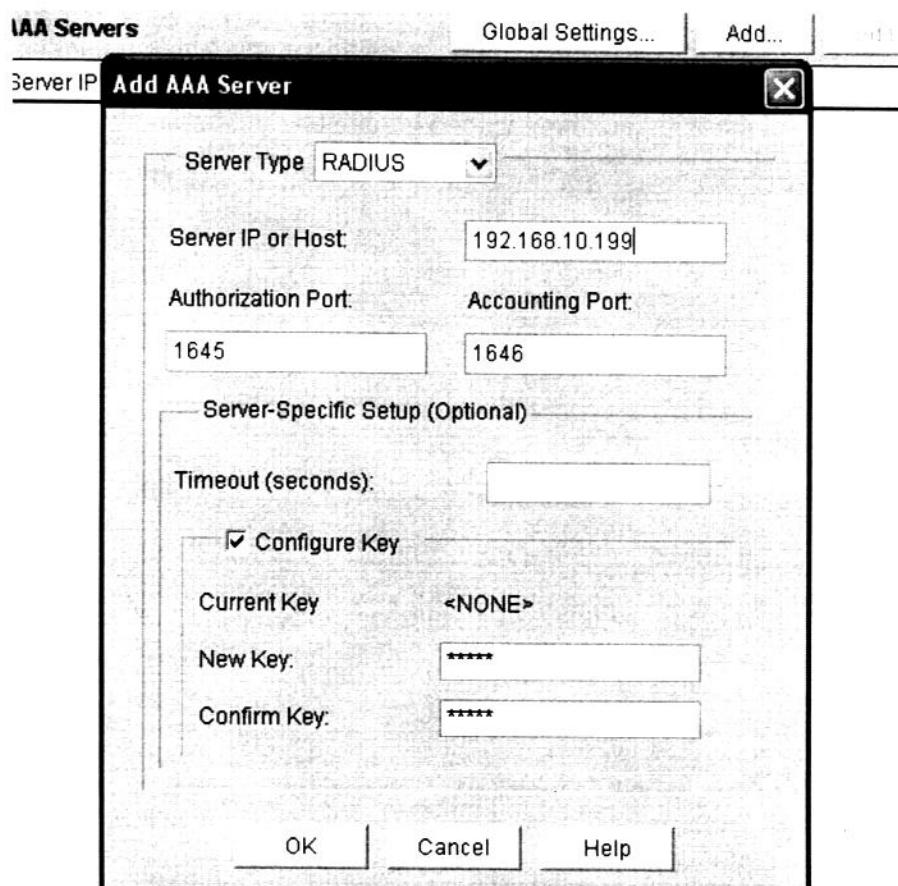
Authentikációs szervereket az **Additional Tasks->AAA->AAA Servers and Groups->AAA Servers** alatt lehet konfigurálni.



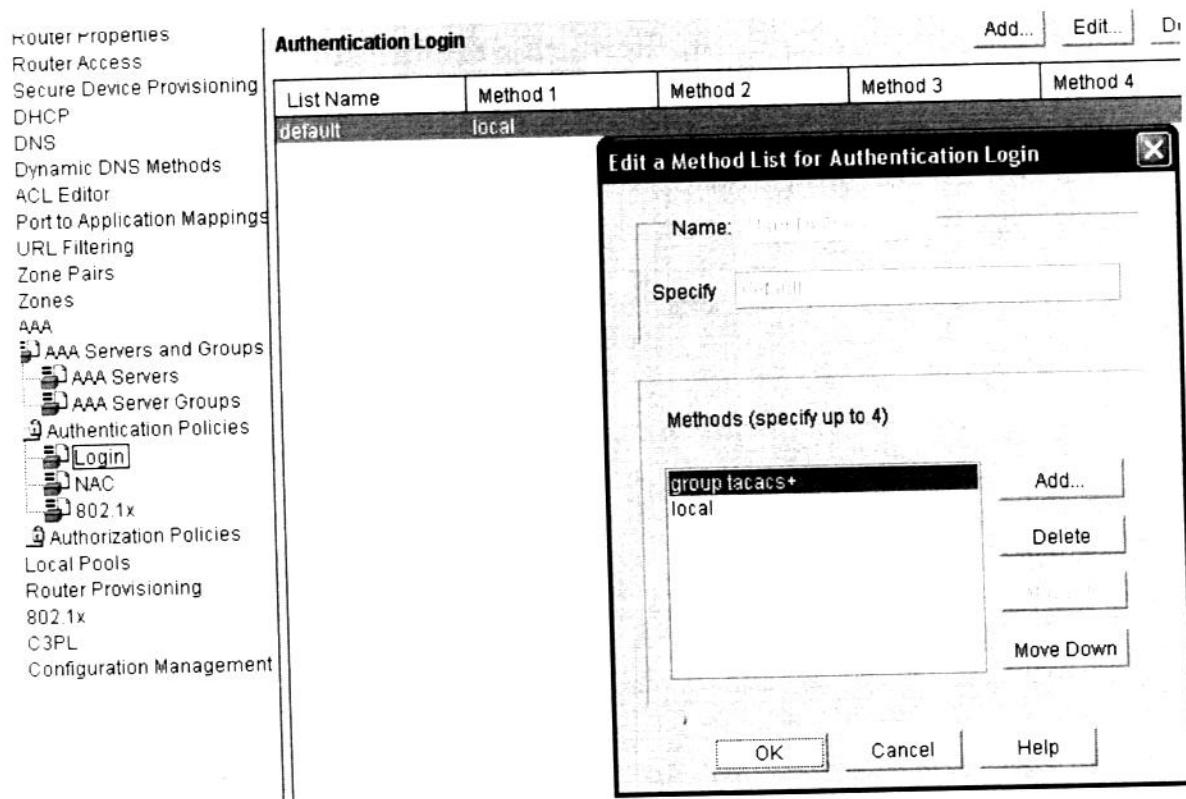
A „**Global Settings**” gomb az összes TACACS+/RADIUS szerverre érvényes beállításokat adja meg. (Kivéve, ha azt az egyes szervereknél külön megadjuk.) Itt van lehetőség a TACACS+/RADIUS üzenetek forrás interfészének kiválasztására is.

A RADIUS és TACACS+ globális beállításai ugyanazt a beállítási lehetőségeket tartalmazzák.

Authentikációs szerver hozzáadása az „Add...” gombbal lehetséges. Attól függően hogy RADIUS vagy TACACS+ szerver tipust választunk, a megfelelő tulajdonságok beállítására van lehetőség:

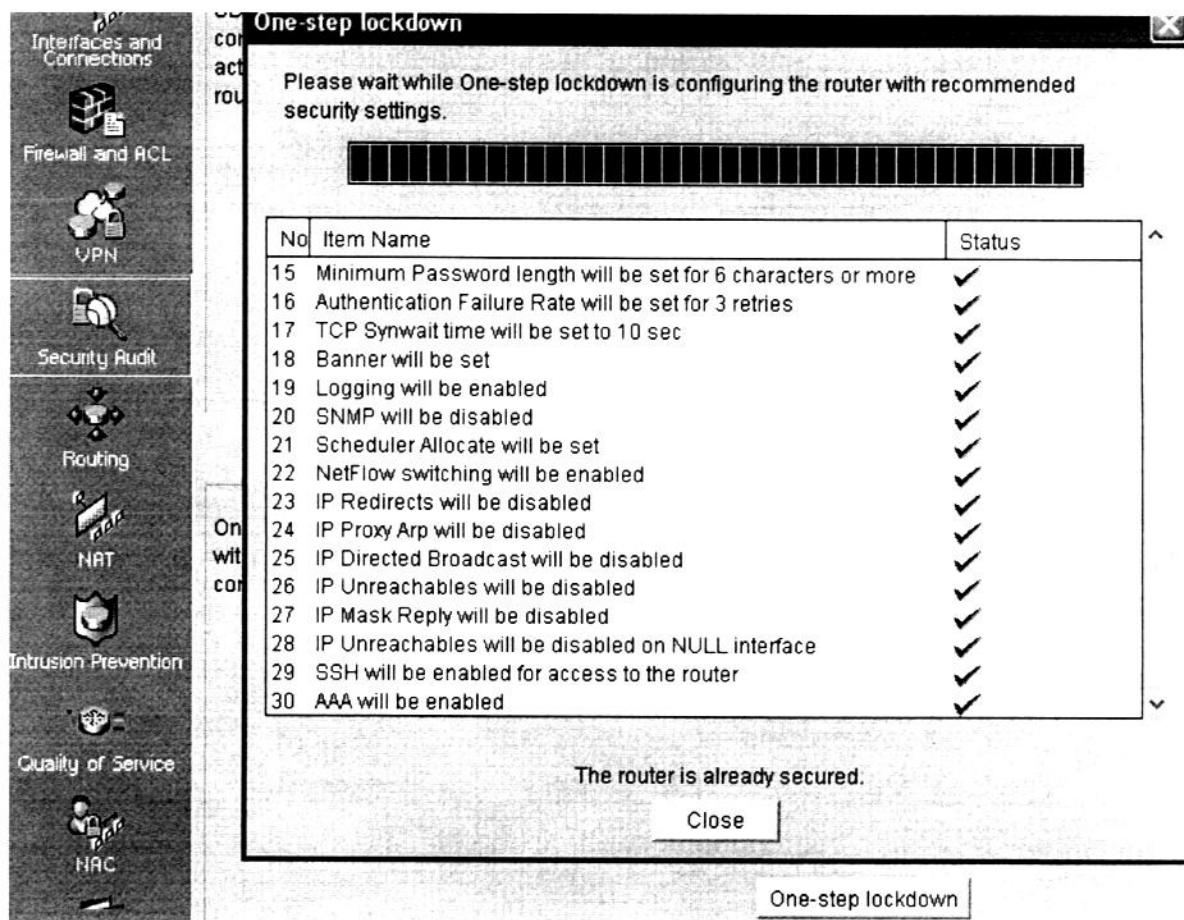


A TACACS+/RADIUS szerver konfigurációját követően meg lehet változtatni a bejelentkezés authentikálásának metódusát az **Additional Tasks->AAA->Authentication Policies->Login** alatt. Az „Edit...” gombra kattintva az előzőleg definiált alapértelmezett (default) method list-et is szerkeszthetjük és az „Add...” gombbal újabb methódusokat adhatunk hozzá a jelenlegihez. A „Move Up/Move Down” gombokkal az methódusok használatának sorrendjét állíthatjuk be.



Router Lock-Down

Számos funkció kikapcsolása és biztonsági beállítások megtétele biztonságosabbá teszi a routert. A Cisco IOS AutoSecure funkciója melyet parancssorból az „**auto secure**” parancssal indíthatunk, vagy SDM-ben a „**Configure->Security Audit->One step lockdown**” gombjával indíthatunk el egy-egy varázslót. A „**One step lockdown**” egy listát is megjelenít az elvégzendő konfigurációs módosításokról, még az „**auto secure**” egy interaktív „varázsló” után a konfigurációs parancsokat mutatja meg. A „**One step lockdown**” nem támogatja az összes lehetőséget, amit az „**auto secure**” igen.



Out-of-Band management (OOB)

Eszközök biztonságát növeli, ha úgynevezett **Out-of-Band** management hozzáférés is van hozzájuk, amely az adatforgalomtól el van szeparálva. Ez lehet egy külön management hálózat, vagy akár remote konzol hozzáférés is.

SSH vs. Telnet

Biztonság növeléséhez hozzátarozik a biztonságos management protokollok használata.

Telnet az összes adatot, beleértve a bejelentkezéshez szükséges felhasználói név és jelszó párost kódolatlanul küldi át a hálózaton. Az SSH ezzel szemben titkosított csatornát épít fel, így a külső szemlélő számára értelmezhetetlen adatfolyamnak tűnik minden nemű kommunikáció.

További fontos tényező a log üzenetek naplózása, egy külön szerverre történő mentése. A log üzenetek megfelelő értelmezéséhez elengedhetetlen a helyes idő és dátumbeállítás a routereken, ellenkező esetben, komoly időzavarba kerülhetünk. Az automatikus dátum és időfrissítést NTP protokollon keresztül kell elvégezni.

Az SNMP hozzáférésekkel is érdemes korlátozni, csak olvasási joggal rendelkező community-re, melyet egy további ACL is véd, mely megszabja mely hosztok csatlakozhatnak az eszközhöz.

Természetesen megfelelően gondoskodni kell a syslog, NTP, SNMP és egyéb management hosztok megfelelő védelméről, akár VLAN ACL-ekkel, hogy ne nyújtsanak újabb támadhatósági felületet.

Telnet és SSH engedélyezése

Telnet engedélyezéséhez a line vty 0 4 alatt kell csupán jelszót beállítani. AAA használata esetén pedig egy lokális felhasználónak kell konfigurálva lennie (vagy TACACS+/RADIUS authentikáció).

SSH engedélyezéséhez RSA kulcspárt kell generálni, melynek előfeltételei:

- domain név beállítása
- hosztnév beállítása

Majd ezek beállítása után lehet legeneráltatni a routerrel az RSA kulcspárt. A generálás során a router alapértelmezésként 512 bites RSA kulcsot generál, amit érdemes legalább 1024 bit-re változtatni.

```
Router(config)# ip domain-name ciscoworld.hu
Router(config)# hostname 7206VXR
```

```
7206VXR(config)# crypto key generate rsa
The name for the keys will be: 7206VXR.ciscoworld.hu
Choose the size of the key modulus in the range of 360 to 2048 for your
General Purpose Keys. Choosing a key modulus greater than 512 may take
a few minutes.
```

How many bits in the modulus [512]: 1024

% Generating 1024 bit RSA keys, keys will be non-exportable...[OK]

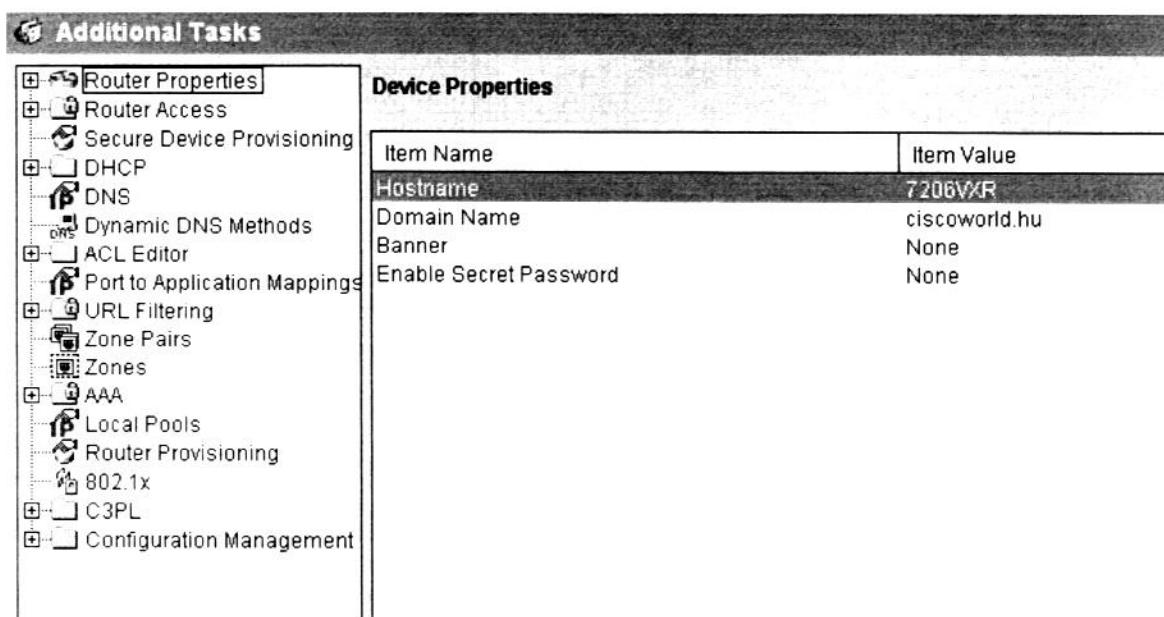
```
7206VXR(config)#
Dec 5 15:58:44.933: %SSH-5-ENABLED: SSH 1.99 has been enabled
```

Az utolsó log üzenet tájékoztat arról, hogy az SSH engedélyezve lett a routeren.

SSH engedélyezése SDM-ből

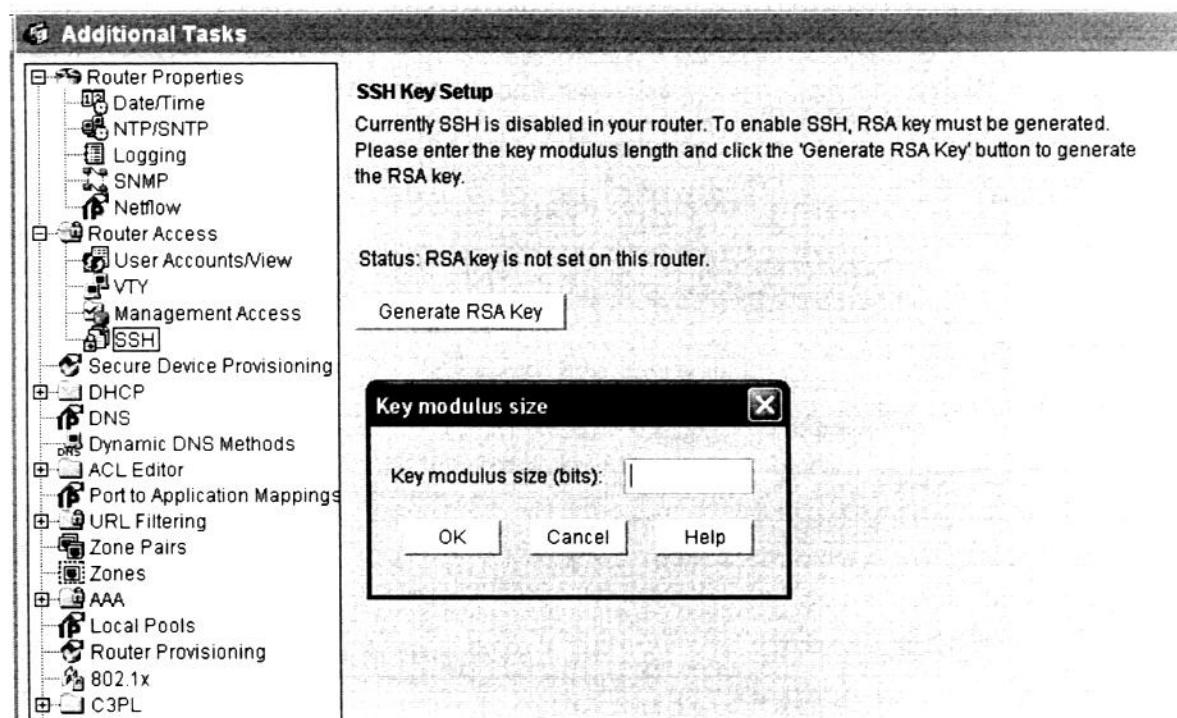
Az SSH-t az „**Additional Tasks->Router Access->SSH**” menüből lehet engedélyezni. Amennyiben még nincs RSA kulcs generálva a routeren, akkor az SDM felületén először ezt kell elvégezni.

A kulcs generálásához a hoszt- és domain név az „**Additional Tasks->Router Properties**” alatt állítható be:



További feltétele az „**aaa new-model**” bekapcsolása is, illetve egy 15-ös privilegium szinttel rendelkező felhasználó konfigurálása. (Konzol esetén nem szükséges)

Ezek után az „**Additional Tasks->Router Access->SSH**” menüben, a „**Generate RSA Key**” gombra kattintva lehet az RSA kulcsot generáltatni.



Az RSA kulcs méretének megadása után egy 15-ös privilegium szinttel rendelkező felhasználói név és jelszó megadása következik.

Amennyiben az authentikálása a felhasználónak sikeres, az SSH engedélyezve lesz az eszközön.

További biztonsági beállítások SSH-hoz

Használt protkoll megadása (telnet v. ssh) az alábbi módon lehetséges:

```
Router(config)# line vty 0 4
Router(config-line)# transport input ssh

Router(config-line)# transport input ssh telnet
```

Ha csak egy protokoltt adunk meg, akkor csak az lesz engedélyezve, ha minden protokoll engedélyezni szeretnénk, akkor fel kell sorolni minden kettőt, ahogy az a második esetben látható.

A hibás bejelentkezési lehetőségek szám az alábbi parancssal adható meg:

```
Router(config)# ip ssh authentication-retries 5
```

A bejelentkezéskor a login prompt várakozása az alábbi parancssal szabható meg (érتهke másodpercben):

```
Router(config)# ip ssh time-out 10
```

Meglevő SSH kulcspárt az alábbi módon törölhetünk:

```
Router(config)# crypto key zeroize rsa
```

Syslog

A router által generált log üzeneteket az alábbi helyekre küldheti az eszköz:

- konzolport: a konzolon minden log üzenet automatikusan megjelenik
- VTY: a „terminal monitor” parancs kiadása után a konzolporthoz hasonlóan minden log üzenet megjelenik
- Buffer: a megjelent log üzeneteket az eszköz úraindításáig a bufferben eltárolja (általában 4-8kbyte)
- SNMP szerver
- Syslog szerver

Syslog az üzeneteket komolyáguk szerint szintekre ossza, melyek az alábbiak:

0. Emergencies
1. Alerts
2. Critical
3. Errors
4. Warnings
5. Notifications
6. Informational
7. Debugging

Ez minden log üzenetben megtalálható. Az alábbi péda egy 5-ös szintű log üzenet:

*Nov 30 20:34:31.263: %LINEPROTO-5-UPDOWN: Line protocol on Interface Ethernet0, changed state to up

A bufferben tárolt logokat a „**show logging**” parancssal lehet megtekinteni.

Syslog server az alábbi parancssal konfigurálható:

```
Router(config)# logging 192.168.1.10
```

SNMP – Simple Network Management Protocol

SNMP részei:

- **SNMP agent:** egy eszközön futó szoftver
- **SNMP szerver:** szerveren futó alkalmazás, mely fogadja az SNMP üzeneteket és SNMP kéréseket indít
- **Management Information Base (MIB):** struktúrált objektumokon keresztül lehet az adatokat SNMP-n keresztül lekérni a routerről

Log üzeneteket SNMP-n keresztül is elküldhetjük a menedzsment szervernek. Egy esemény esetén az eszköz SNMP Trap formájában, üzenetet küld a menedzsment szervernek.

SNMP további lehetőségéhez tartozik, hogy az SNMP szerver kéréseket tud intézni az eszközhöz, melyben különböző adatokat (leginkább statisztikai) tud lekérni róla. Pl. forgalomi statisztika, interfész állapotok, stb...

SNMP „jelszó” gyanánt egy úgynevezett community-t használ. Erre a community-re hivatkozva lehet kéréseket küldeni az eszközre, illetve a trappek is egy community-t használnak a management szerver felé.

SNMP verziói:

- v1: első SNMPverzió, számos biztonsági kérdést felvetett
- v2: az SNMP átdolgozott verziója, nem kompatibilis az előző verzióval
- v2c: az SNMP v1-el kompatibilis megoldás
- v3: legújabb verziója az SNMP protokollnak, magas biztonsági szabályzásokkal (encryptálás, felhasználók)

CLI konfigurációja az alábbi módon lehetséges:

```
Router(config)# snmp-server community ciscoworldSNMP RO
```

```
Router(config)# snmp-server host 10.1.1.12 ciscoworldSNMP
```

Az első példában beállított csak olvasható community-t használva lekérdezhetjük a routert. A második esetben egy menedzsment szerver IP címét konfiguráltuk, melynek az SNMP trap üzenetek küldve lesznek.

Az SNMP hozzáférést érdemes ACL-el limitálni, megszabni mely hálózatok vagy hosztok csatlakozhatnak a routerre SNMP protokollen keresztül. Ez az alábbi módon lehetséges:

```
Router(config)# access-list 10 permit 10.1.1.12
```

```
Router(config)# snmp-server community ciscoworldSNMP RO 10
```

Mivel SNMP keresztül nem csak adatok kiolvasása lehetséges, hanem konfiguráció is, ezért biztonsági szempontból jól meg kell fontolni a Read-Write (RW) community konfigurálását.

SNMP konfigurálása SDM-ben

SDM-ben az SNMP-t az „Additional Tasks->Router Properties->SNMP” állíthatjuk be.

Layer 2-es támadások

Layer 2-es eszközök leginkább belső támadásoknak vannak kitéve, mivel a külső támadókat egy Layer 3-as eszköz választja el a belső hálózattól.

CAM overflowing – MAC spoofing és port security

Ha egy switch túl sok MAC címet jegyez meg, a memoriája betelte után HUB-ként viselkedik (fail-open mode), mely biztonsági problémákat vet fel. Számos program áll rendelkezésre (pl.: macof), hogy egy PC-ről keretek százezreit küldjük különböző forrás MAC címekkel, betelítve így a switch memóriakapacitását.

MAC spoofing ellen port-security konfigurálása ajánlat, melyel meghatározhatjuk, hogy hány darab és milyen MAC címek jelenhetnek meg egy porton forrásként. Konfigurálása során az alábbiak adhatóak meg:

- MAC címek maximális száma (alapértéke egy)
- MAC címek statikus konfigurálása
- MAC címek dinamikus megtanulása
- MAC címek dinamikus megtanulása és tárolása (sticky)
- maximális MAC cím szám túllépés esetének kezelése
 - shutdown (alapértelmezés): a portot inaktiválja és err-disabled státuszba teszi
 - protect: a riasztást kiváltó MAC címet nem engedi, de a korábbi MAC cím forgalmát igen
 - restrict: protect-től annyiban különbözik, hogy SNMP ill. syslog üzenetet küld

Konfigurálása az alábbiak szerint lehetséges:

```
Switch(config)# interface fa0/1
Switch(config-if)# switchport port-security
Switch(config-if)# switchport port-security mac-address aaaa.bbbb.cccc
Switch(config-if)# switchport port-security mac-address sticky
Switch(config-if)# switchport port-security maximum 5
Switch(config-if)# switchport port-security violation shutdown
```

A fenti példában a switchporton maximum 5 MAC cím lehetséges, ebből egy statikusan van konfigurálva, a többöt pedig dinamikusan tanulja meg és tárolja el a running-config-ban. Ha nem mentjük a konfigurációt, akkor elvesznek a **sticky address**-ek. Ha a sticky opción elhagyjuk, akkor a dinamikusan megtanult MAC címek az aging idejéig lesznek nyilvántartva, ennek letelte után más MAC címek is megjelenhetnek a porton.

Port-security ellenőrzése a „**show port-security**”, „**show port-security interface FastEthernet 0/1**”, és a „**show port-security address**”, parancsokkal lehetséges:

A „**show port-security address**” parancs megmutatja, milyen MAC címek szerepelnek az egyes portokon:

```
Switch#show port-security address
Secure Mac Address Table
-----
Vlan Mac Address Type          Ports          Remaining Age
                           (mins)
-----
1   00E0.8FD0.0001  DynamicConfigured  FastEthernet0/1  -
1   0090.2B76.0002  DynamicConfigured  FastEthernet0/2  -
-----
Total Addresses in System (excluding one mac per port) : 0
Max Addresses limit in System (excluding one mac per port) : 1024
Switch#
```

Az „**show port-security**” parancs, az aktuális beállításokat mutatja meg:

```
Switch#show port-security
Secure Port MaxSecureAddr CurrentAddr SecurityViolation Security Action
                           (Count)      (Count)      (Count)
-----
Fa0/1        1           1           0           Shutdown
Fa0/2        1           1           0           Shutdown
-----
Switch#
```

A „**show port-security interface fa0/1**” parancs az interfész státuszáról és érvényes beállításait mutatja:

```
Switch#show port-security interface fa0/1
Port Security           : Enabled
Port Status              : Secure-up
Violation Mode          : Shutdown
Aging Time               : 0 mins
Aging Type               : Absolute
SecureStatic Address Aging : Disabled
Maximum MAC Addresses    : 1
Total MAC Addresses      : 1
Configured MAC Addresses : 0
Sticky MAC Addresses     : 0
Last Source Address:Vlan : 00E0.8FD0.0001:1
Security Violation Count : 0
```

Egy idegen MAC cím megjelenése esetén az alábbi lesz a parancs kimenetele:

```
Switch#show port-security interface fa0/1
Port Security           : Enabled
Port Status              : Secure-shutdown
Violation Mode          : Shutdown
...
Last Source Address:Vlan : 00E0.8FD0.0003:1
Security Violation Count : 1

Switch#
```

A „**show interface fa0/1**” parancs mutatja a port státuszát: (err-disabled) ami port-security miatt lett inaktiválva.

```
Switch#show interface fa0/1
FastEthernet0/1 is down, line protocol is down (err-disabled)
```

Port-security MAC cím típusok:

- **Static secure MAC address:** kézzel konfigurált MAC cím a porton
- **Sticky secure MAC address:** switch által dinamikusan tanult, *running-config*-hoz adott MAC cím
- **Dynamic secure MAC address:** switch által dinamikusan tanult MAC cím. Ageing letelte után elvész.

VLAN hopping

VLAN hopping akkor következik be, ha egy felhasználó egy nem a számára kijelölt VLAN tagjává válik, vagy abba a VLAN-ba küld csomagot, anélkül hogy egy Layer 3-as eszközön route-olva lett volna a csomag. Két legelterjedtebb változata a *switch spoofing* és a *double tagging*.

DTP veszélyei

Ethernet trunk linkek alapbeállításukkal minden VLAN forgalmát szállítják. Ezért ha egy támadó trunk kapcsolatot tud létrehozni egy switchel, akkor bármely VLAN tagjává képes válni, illetve bármely VLAN forgalmát láthatja. Ezáltal akár felhasználói név és jelszó páros lopására is lehetősége nyílik.

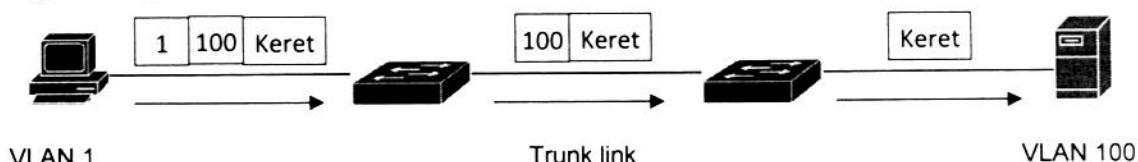
Számos Cisco Catalyst switch DTP auto módot használ a portjain. Ezáltal ha egy másik külső switchet csatlakoztatunk rá, a kettő közötti linket a DTP trunk módúvá formálhatja. Ezután a csatlakoztatott switch bármely portját a kívánt VLAN-ba lehet konfigurálni, így a trunk linken az adott VLAN-ra rá lehet csatlakozni.

Ennek megakadályozására minden porton ki kell kapcsolni a DTP-t és kézzel kell access vagy trunk módra konfigurálni a portot. Ez az alábbi parancsokkal lehetséges:

Switch(config)# interface fa0/1	
Switch(config-if)# switchport mode access	! trunk kikapcsolása
Switch(config-if)# switchport nonegotiate	! DTP kikapcsolása

Double tagging

A támadó két VLAN taggel ellátott keret esetén a trunk link native VLAN taggeletlenségét kihasználva keretekeket juttathat át egy másik VLAN-ba. Ennek feltétele, hogy a felhasználó és a native VLAN a trunk linken ugyanaz legyen.



Ez ellen az alábbiakat lehet tenni:

- a native VLAN egy nem használt VLAN-ra állítása

```
Switch(config-if)# switchport trunk native vlan 999
```

- a native VLAN taggolásának beállítása

```
Switch(config)# vlan dot1q tag native
```

- VLAN 1 mint native VLAN használatának mellőzése
- A nem használt portok egy nem használt VLAN-ba konfigurálása, vagy „admin down”-ba tevése
- felhasználó portok statikus access módra konfigurálása

Az újabb switchek már nem továbbítják azokat a keretekeket, melyek VLAN taggel érkeznek, azonban nem kellene azzal érkezniük.

Spanning-tree védelme – Root guard, BPDU guard

Egy alacsonyabb BID-vel rendelkező switch csatlakoztatásával az STP topológia átrendezhető, mely a hálózat lassulásához vezet, illetve az STP topológia ágai között áramló forgalom az új root bridgen keresztülhaladva adatlopásra ad lehetőséget.

BPDU guard

BPDU guard a portfast-ra konfigurált portokon használatos. Portfast beállításával a felhasználókhöz menő portok nem várakoznak az STP listening és learning státuszában, hanem egyből forwardingba mennek át. Felhasználókhöz menő portokon nem kell loopra és BPDU-ra számítani, mivel azok nem switchekhez kapcsolódnak.

Ha ezeken a portokon mégis BPDU érkezik, a BPDU guard a portot inaktiválja, és az esetet ki kell vizsgálni, hogy mi okozta a porton kapott BPDU-t. Legtöbb esetben a felhasználók csatlakoztatnak switcheket vagy egy HUB-on keresztül több uplink portot. Konfigurálása az alábbi módon lehetséges:

```
Switch(config)# interface Gi0/1
Switch(config-if)# spanning-tree portfast bpduguard
```

Root guard

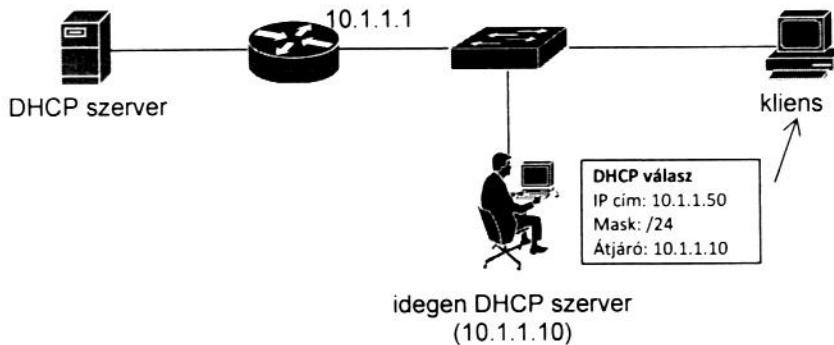
Ha egy porton egy jobb BPDU érkezik, amely miatt a port root porttá válhat, a port **root-inconsistent** státuszba kerül és mindenkoruknak a BPDU-knak az áradata meg nem szűnik abban marad. Ebben a státuszban semmilyen felhasználói forgalmat nem továbbít a switch.

Minden olyan porton, amelyen keresztül a root bridge elérése nem várható, be lehet kapcsolni ezt a funkciót az alábbi módon:

```
Switch(config)# interface Gi0/1
Switch(config-if)# spanning-tree guard root
```

DHCP snooping

Ha egy idegen DHCP szervert helyezünk el a hálózaton, akkor a kliensek DHCP kérésére az idegen és a legitim DHCP szerver is választ fog küldeni. Ha az idegen DHCP szerver válasza érkezik meg hamarabb a klienshez, akkor az általa ajánlott IP konfigurációs beállításokat fogja elfogadni a kliens. A veszély abban rejlik, hogy az idegen DHCP szerver önmagát megadva alapértelmezett átjárónak, a kliensről indított hálózaton kívüli forgalmat maga felé tereli, majd azt az igazi átjárónak továbbküldve, látja a forgalmat. Ez a legtöbb esetben sokáig észrevéten marad.



DHCP snooping használatával a Catalyst switchek trusted vagy untrusted port-ként kezelik a portokat. Ha egy untrusted porton DHCP válasz érkezik, akkor a port inaktiválva lesz. DHCP válaszok csak trusted porton keresztül jöhetnek. DHCP snooping alapértelmezésben minden portot untrusted ként kezel. DHCP snoopingot engedélyezni és trusted portokat az alábbi parancssal lehet kijelölni:

```
Switch(config)# ip dhcp snooping
Switch(config)# interface fa0/1
Switch(config-if)# ip dhcp snooping trust
```

Amennyiben csak egy néhány specifikus VLAN számára kell engedélyezni a DHCP snoopingot, akkor a VLAN-ok listáját a parancs után kell megadni az alábbi módon:

```
Switch(config)# ip dhcp snooping vlan 1, 10-15, 19
```

DHCP kiéheztetés (starvation)

Ha egy kliens több IP címet kér magának (különböző MAC címeket használva), a rendelkezésre álló összes IP címet lefoglalhatja, így a többi kliens számára nem marad szabad IP cím. Ezt elkerülendő szabályozhatjuk, hogy a DHCP szerver másodpercenként mennyi DHCP választ küldhet, az alábbi módon:

```
Switch(config)# interface fa0/1
Switch(config-if)# ip dhcp snooping rate limit 5
```

Dynamic ARP inspection (DAI)

DHCP snooping egy dinamikus táblázatot épít, hogy melyik MAC címhez milyen IP cím lett kiosztva. Ezt a táblázatot felhasználva az ARP válaszokban levő MAC és IP címet a switch össze tudja hasonlítani a nyilvántartásában levővel. DAI a DHCP snoopinghoz hasonlóan trusted és untrusted portokkal dolgozik. Ha egy ARP válasz érkezik egy untrusted portján keresztül, akkor a DHCP snooping által épített adatbázissal (DHCP binding table) összehasonlíta a MAC és IP címet. Amennyiben nem egyezik, az ARP választ eldobja, és a portot inaktiválja. GARP üzenet érkezése esetén is ugyanígy jár el. DAI engedélyezését követően minden port untrusted státuszú. DAI engedélyezése egy VLAN-on és a trusted port konfigurálása az alábbi parancsokkal lehetséges:

```
Switch(config)# ip arp inspection vlan 10
Switch(config)# interface fa0/1
Switch(config-if)# ip arp inspection trust
```

VLAN ACL-ek (VACL)

Layer 3-as határon IP ACL-ekkel lehet szűrni a forgalmat a különböző hálózatok között. Egy VLAN-on belül azonban a forgalom szabadon áramolhat. VLAN ACL-ekkel lehetőség van a VLAN-on belül ACL-eket kialakítani, és azokkal szabályozni a forgalmat.

Az alábbi példában az 1, 5-12-es VLAN-on belül csak az SSH forgalom engedélyezett a 192.168.1.21-es IP címre:

```
Switch(config)# access-list 100 permit tcp any host 192.168.1.21 eq 22
Switch(config)# vlan access-map VACL-ALLOW-SSH 10
Switch(config-access-map)# match ip address 100
Switch(config-access-map)# action forward
Switch(config-access-map)# exit

Switch(config)# vlan filter VACL-ALLOW-SSH vlan-list 1,5-10
```

Private VLAN

Másik módja a Layer 2-es biztonság növelésének a private VLAN-ok konfigurálása. Az azonos PVLAN-ba tartozó, de izolált portok egymás között nem kommunikálhatnak, csak a promiscuous porttal.

PVLAN-ok az alábbi portokat különbözteti meg:

- **isolated:** csak a promiscuous porttal kommunikálhat
- **community:** ugyanabba a community-ba tartozó portokkal és a promiscuous porttal kommunikálhat
- **promiscuous:** minden más porttal kommunikálhat

Private VLAN-ok konfigurációja a CCNP SWITCH része.

SAN (Storage Area Network)

A nagyobb tárhely méret igényének növekedtével a központositott adattárolás és hálózati adattárolási megoldások előtérbe kerülésével, egyre több támadás irányul ezen eszközök és technológiák ellen.

SAN-ok alkalmazásának az alábbi előnyei vannak:

- dinamikus skálázhatóság
- megtérülés növekedése
- telepítési és működtetési költségek csökkenése

SAN-ok használatával a vírusok és férgek által okozott károk azonban nem csökkennek, mivel a fájlok és fájlrendszerek ugyanolyan jogosultságokkal bírnak, mint a helyi adattárolás esetén, azzal a különbséggel, hogy az eszközök fizikailag másol helyezkednek el.

SAN-ok alapját az SCSI adja. Alkalmazás módjától függően több újabb technológia került kidolgozásra:

- FibreChannel és iSCSI: kliens és a SAN között
- FCIP: SAN-to-SAN kapcsolat

SAN-ok elleni leggyakoribb támadások:

- spoofing: bizalmasság és érintetlenség ellen
- snooping: bizalmasság ellen
- DoS: elérhetőség ellen

LUN (Logical Unit Number): SCSI-nál használt, az egyes disk-eket azonosítja

LUN masking: Az egyes LUN-okat elérhetővé lehet tenni csak bizonyos hosztok számára, a többieknek nincs hozzáférésük. Ez tipikusan a HBA (Host Bus Adapter) szintjén történik, ezért a HBA elleni támadások veszélyeinek van kitéve.

SAN zónázás

Zónázással a hálózati topológiában, switch szinten lehet a hozzáféréseket kontrollálni, hasonlóan egy ACL-hez.

Soft zónázás esetén a végpontok elől el lehet rejteni az eszközök elérhetőségét, azonban ez nem azt jelenti hogy nem éri el. Hard zónázás ténylegesen elzárja az eszközök egymással való kommunikációját.

WWN (World Wide Name): 64 bites felhasználó által hozzárendelhető cím. WWN spoofing-gal kijátszható az erre alapozott zónázás.

VSAN (Virtual SAN): a VLAN-okhoz hasonlóan a SAN-okat is el lehet egymástól szeparálni.

Port authentikációs protokollok

- DHCAP (Diffie-Hellman CHAP): FibreChannel-nél használt authentikációs protokoll
- CHAP: iSCSI-nál használt authentikációs protokoll
- FibreChannel Authentication Protocol (FCAP): digitális tanúsítványokkal (PKI) biztosítja az authentikációt, illetve ESP használatával titkosítja az adatokat. Emiatt körülmenyesebb az implementációja.

Digitális tanúsítványokról a későbbi fejezetekben lesz részletesen szó.

VOIP biztonság

VOIP technológia olcsósága és skálázhatósága miatt az egyik legdinamikusabban fejlődő hang alapú szolgáltatás.

Komponensei az alábbiak:

- **IP phone:** hagyományos telefonkészülékhez hasonlóan, IP alapú hangtovábbításnak a végpontjai
- **Call agent:** hagyományos telefontözpontokat váltja fel
- **Gateway:** különböző technológiák közötti átájárást biztosítja (pl VoIP és PSTN)
- **Gatekeeper:** sávszélesség monitorozásával, menedzselésével engedélyezik, illetve tiltják a hívásokat
- **Multipoint Control Unit (MCU):** konferenciahívások kezelése
- **Application server:** további szolgáltatások (pl. voice mail) biztosítása

VOIP protokollok adatai kódolatlanul mennek a hálózatban, mely lehetővé teszi a VOIP accountok ellopását, illetve a továbbított hanganyag visszajátszását. Ennek megoldásaként biztonságos, encryptálást használó protokollok alkalmazása ajánlott, vagy a titkositani kívánt adatokat egy IPsec tunel kell keresztül routeolni. További problémát jelenthetnek a DoS támadások, melyek főként egy call-center számára okoz nagy kiesést, hiszen a szolgáltatása a VOIP rendszerre épül.

Olcsósága miatt a Spam-over-IP Telephony (SOIT) is előtérbe került. Felhasználókat pedig Víshing során bizalmas adatok kiadására próbálják rávenni.

VOIP rendszerek esetén a Voice és a felhasználó adatforgalmat (Data) külön VLAN-ba teszik. A kábelezés egyszerűsítése végett, a Catalyst switchet az IP telefonnal köti össze, melyet továbbfűznek a PC felé.

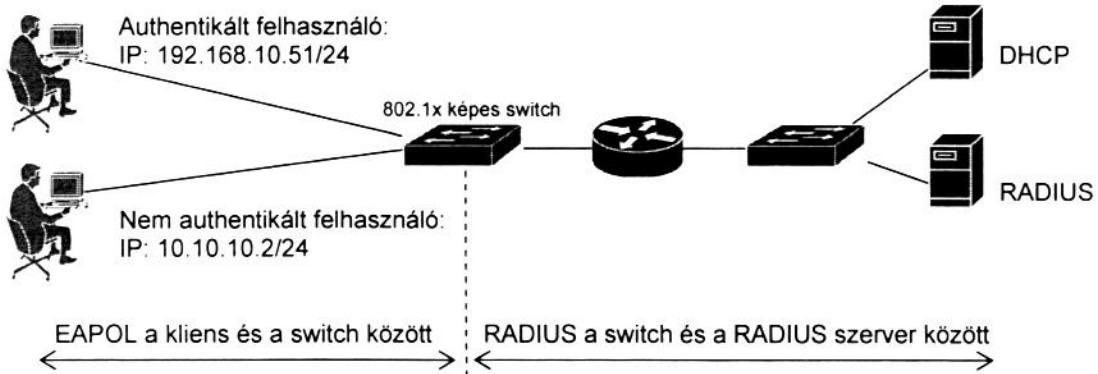


A telefon ésa switch között egy „trunk” kapcsolat jön létre, és a switch a Voice csomagokat más VLAN-ra helyezi, mint az adatcsomagokat. Ezáltal a két forgalom el van szeparálva két külön VLAN-ba. A telefon forgalma meg lesz taggalve a Voice VLAN-nal, PC forgalma azonban taggeletlen marad. A switch CDP-n keresztül érzékeli, hogy IP telefon lett csatlakoztatva, így át tudja neki küldeni a használandó Voice VLAN-t. Ha a switch nem érzékel a telefon jelenlétét, a PC taggeletlen csomagjai továbbra is a Data VLAN-on továbbítódik.

A Cisco IP telefonokon a web-es információs oldal alapértelmezésként engedélyezve van. Mivel semmilyen bejelentkezést nem igényel, további beállítások és szerver IP címek egyszerűen megtudhatóak. A támadást a szerverek ellen intézve pedig nem csupán egy IP telefon, hanem az egész rendszer kiesését okozhatják.

Cisco Identity-Based Network Services (IBNS)

Cisco IBNS az IEEE 802.1x és EAP protokollokra alapul. A kliensek authentikációja egy központi szerveren (RADIUS) történik meg, a switch pedig az authentikáció sikeresége alapján teszi bele a felhasználót egy adott VLAN-ba. Ez által, ha a kliens nem tudja magát megfelelően authentikálni (felhasználónév-jelszó páros, MAC cím, vagy certificate), akkor a kliens nem kerül bele egy VLAN-ba se, vagy egy nem használt VLAN-ba tehető. Sikeres authentikáció során a kliens a hozzá rendelt VLAN tagja lesz. Arra is lehetőség van, hogy a switch portjait ne kelljen egyenként konfigurálni, hanem a 802.1x authentikáció során a RADIUS, függetlenül hogy a kliens mely porton is csatlakozik fel, a neki megfelelő VLAN-ba fog bekerülni, annak megfelelő IP címet fog kapni.



Az authentikáció a switch (*authenticator*) és a kliens (*supplicant*) között egy **EAPOL** (EAP over LAN) csomaggal kezdődik. Az EAPOL csomag alapján a switch üzenetet küld a RADIUS szervernek (*authenticator*), amely visszajelzést ad a switchnek az authentikáció kimenetéről, és a használandó VLAN-ról. Ennek megfelelően a switch a klienshez menő portját, a megfelelő VLAN-ba rendeli bele, melyen keresztül a kliens IP címet tud kérni magának DHCP-n keresztül.

Ameddig a kliens nem authentikálja magát, a port **uncontrolled**, az után pedig **controlled** státuszba kerül.

Az **uncontrolled** (*unauthorized*) port csak az EAPOL, CDP és STP forgalmat továbbítja/dolgozza fel. minden más forgalom eldobásra kerül. Az authentikációt követően a **controlled** (*authorized*) port minden forgalmat továbbít.

Port beállítási opciók:

- **force-authorized:** a port mindenféle 802.1x authentikáció nélkül authorized lesz
- **force-unauthorized:** 802.1x-től függetlenül a port unauthorized lesz
- **auto:** 802.1x authentikáció sikeresége fogja előntení a port státuszát

Port-security-hoz hasonlóan a 802.1x csupán egy eszközt (MAC címet) engedélyez a sikeres authentikációt követően. Ha egyszerre több eszköz érhető el azon a porton, akkor a **multi-host** opciót is konfigurálni kell. Ez esetben az első sikeresen authentikáló kliens fogja a portot engedélyezni az összes több, akár nem EAPOL képes készülék számára is.

A 802.1x működését globális konfigurációs módból az alábbi parancsokkal kell engedélyezni:

```
Switch(config)# aaa new-model
Switch(config)# aaa authentication dot1x default group radius
Switch(config)# dot1x system-auth-control
```

Ez után interfészenként lehet engedélyezni a 802.1x használatát:

```
Switch(config)# interface fa0/1
Switch(config-if)# switchport mode access
Switch(config-if)# dot1x port-control auto
```

Multi-host opció az alábbi parancssal kell bekapcsolni:

```
Switch(config-if)# dot1x host-mode multi-host
```

Természetesen a dot1x esetén RADIUS szervernek is konfigurálva kell lennie. Ennek módja az előzőekben már tárgyalva volt.

Tűzfalak

A tűzfalak több generáción keresztül fejlődtek, kezdve a **Statikus tűzfalakkal**, melyek egyszerű Layer 2 és Layer 3 adatok alapján ACL-ekkel szűrték a csomagokat, szabályozták a hozzáféréseket. Második generációs **Circuit-level** tűzfalak már képesek voltak megállapítani, hogy egy csomag egy új kapcsolatnak az első csomagja, vagy egy meglévő adatfolyamhoz tartozik. Majd az **Application layer firewall** már képes lekezelni azon alkalmazásokat, melyek a payload-ban további információkat hordoznak, portokat nyitnak meg. NAT esetén ezeket az információkat dinamikusan frissítik. További ellenőrzéseket is elvégeznek, illetve alkalmazásszűrést (Java blocking) is végeznek.

Az Application layer firewall-okhoz hasonlóan a **statefull firewall** képes kezelní a dinamikusan nyitott portokat, illetve a state table-ben nyomon követik a kommunikációt.

Transzparens firewall: Layer 2-ben működik, nem látszik egy újabb hop-nak, azonban a lábai külön, elszeparált VLAN-ba tartoznak. A kliensek számára teljesen láthatatlan.

Proxy szerverek: kliens nevében eljáró, tartalomtárolást is megvalósító alkalmazás. A kliens a kérését a proxy szervernek küldi el, amely a kérést teljesíti, majd az eredményt visszaküldi a kliensnek. Ha egy másik kliens ugyanazt az adatot kéri le, akkor a proxy szerver nem kéri le újból az oldalt, hanem a lokálisan tárolt példányból gyorsabban vissza tudja azt küldeni.

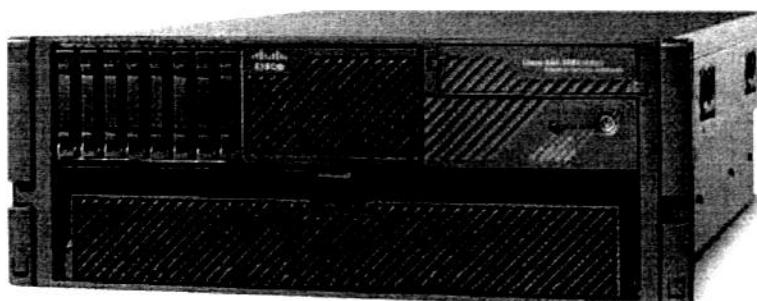
Az egyes tűzfalmegoldások tartalmazhatnak elemeket több generációból is.

Cisco tűzfal eszközei

A Cisco PIX és új generációja a **Cisco ASA** (Adaptive Security Appliance) képezi a Cisco tűzfal eszközeit.



Cisco ASA 5510



Cisco ASA 5580



Cisco ASA 5505

ACL-ek

ACL-ek alapjai a Cisco CCNA tananyag része. Cisco CCNA Security csupán az elemeket tárgyalja, amelyek túlnyúlnak a Cisco CCNA tananyagán.

Turbo ACL: minél nagyobb egy ACL annál nagyobb processzor teljesítményt, memóriát és időt igényel a vizsgálata. A Cisco 7200, 7500 és 12000 szérijű routerek támogatják a Turbo ACL funkciót, melyel az ACL-ek egy lookup táblába generálódnak le (hasonlóan a L3 switchekhez) biztosítva ezzel fix lépésszámú, gyors feldolgozást. Bekapcsolásával, minden három vagy több soros ACL ez esetben Turbo ACL lesz. Bekapcsolni az alábbi módon lehetséges:

```
Router(config)# access-list compiled
```

A Turbo ACL-eket pedig az alábbi parancssal lehet kilistázni:

```
Router# show access-list compiled
```

További funkcióját tekintve teljesen megegyezik a hagyományos ACL-ekkel.

ACL-ek további fontos jellemzői

- Router által generált csomagokat nem lehet ugyanazon a routeren, kimenő irányban az ACL-ekkel szűrni
- Extended ACL-eket a forráshoz lehető legközelebb ajánlott helyezni
- Standard ACL-eket a célhoz lehető legközelebb ajánlott helyezni

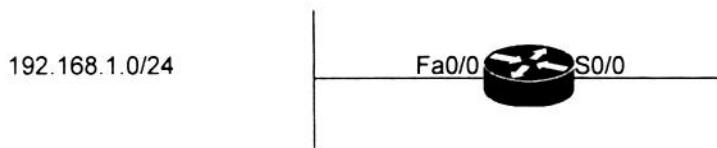
Legfontosabb portok és hozzájuk tartozó szolgáltatások:

20	FTP-DATA	TCP
21	FTP	TCP
22	SSH	TCP
23	Telnet	TCP
25	SMTP	TCP
53	DNS	TCP / UDP
67	BOOTP	UDP
69	TFTP	UDP
80	HTTP	TCP
110	POP3	TCP
137	NetBIOS-NS	TCP / UDP
138	NetBIOS-DGN	TCP / UDP
139	NetBIOS-SSN	TCP / UDP
177	XDMCP	UDP
443	HTTPS	TCP
445	NetBIOS	TCP
514	Syslog	UDP
515	LPR	TCP
1433, 1434	MS SQL	TCP
1900, 5000	Microsoft UPnP SSDP	TCP / UDP
2049	NFS	UDP
3306	MySQL	TCP
3389	RDP	TCP / UDP
6000-6063	X-Window System	TCP

A kiosztott portokról teljes lista a <http://www.iana.org/assignments/port-numbers> linken található.

IP spoofing kiküszöbölése ACL-el

IP spoofing iránya lehet befelé vagy kifelé menő. Mindkét irányban érdemes megszabni, hogy milyen forrás IP címek, melyik irányból nem várhatóak. Belső hálózat IP címeit a külső hálózat felől forrásként, vagy nem a belső IP cím tartományba eső IP címeket forrásként a belső hálózat felől tiltani lehet, az ilyen csomagokat el kell dobni.



Ebben a példában az alábbi ACL-eket lehet beállítani:

Minden forgalmat az S0/0 interfészen bejövő irányban, csak azon csomagokat engedni, melynek célja a 192.168.1.0/24 hálózat és a forrás cím különbözik tőle.

```

Router(config)# access-list 101 deny ip 192.168.1.0 0.0.0.255 any
Router(config)# access-list 101 deny ip 127.0.0.0 0.255.255.255 any
Router(config)# access-list 101 deny ip 0.0.0.0 0.255.255.255 any
Router(config)# access-list 101 deny ip 224.0.0.0 15.255.255.255 any
Router(config)# access-list 101 deny ip host 255.255.255.255 any
Router(config)# access-list 101 permit ip any 192.168.1.0 0.0.0.255

Router(config)# interface s0/0
Router(config-if)# ip access-group 101 in
  
```

Az első sor a belső hálózatot tiltja kívülről, 2-5 sorok minden forrásként nem lehetséges címet tiltanak, az utolsó sor pedig csak a 192.168.1.0/24 hálózatnak címzett csomagokat engedi át. Ezt a S0/0 interfészre kell felenni bejövő irányra.

Ugyanígy a belülről kifele menő forgalmat is lehet szűrni, hogy csak a 192.168.1.0/24-es hálózatból jövő forgalom legyen elfogadna az Fa0/0 interfészen bejövő irányban.

```

Router(config)# access-list 102 permit ip 192.168.1.0 0.0.0.255 any

Router(config)# interface Fa0/0
Router(config-if)# ip access-group 102 in
  
```

ICMP forgalom

Az ICMP számos hasznos dologban segítségünkre lehet, ezért ennek tiltására vagy engedélyezésére érdemes figyelmet szentelni. Bejövő Echo-request csomagokat lehet tiltani, de az Echo-reply-okat és a ttl-exceeded üzeneteket érdemes beengedni, hogy a ping és a traceroute belülről működjön, azonban kívülről ne legyünk elérhetőek. Az alábbi ICMP típusokat érdemes engedélyezni a külső hálózatból:

- echo-reply
- packet-too-big
- ttl-expired

A belső hálózatból érkező alábbi ICMP típusokat ajánlott letiltani:

- redirect

VTY hozzáférés

ACL-ekkel lehetőségünk van szabályozni, a telnet illetve SSH hozzáférést az eszközökhöz. Konfigurálni az alábbi módon lehetséges:

```

Router(config)# access-list 10 permit 192.168.1.99
Router(config)# access-list 10 permit 172.16.99.0 0.0.0.255

Router(config)# line vty 0 4
Router(config-line)# access-class 10 in
  
```

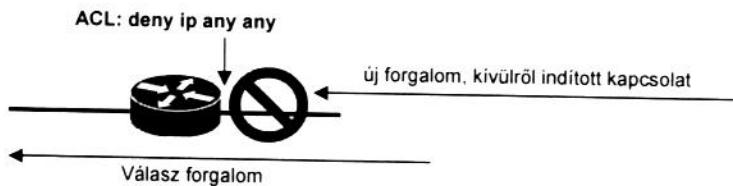
Classic Firewall

Cisco IOS Classic Firewall az alábbi lehetőségekkel rendelkezik:

- Forgalomszűrés:** számos akár Application layer-beli paraméteren alapulva szűri a csomagokat, ezáltal a dinamikusan nyitott portok kezelésére is alkalmas. Session táblában tárolt adatok alapján képes csak azt a forgalmat visszaengedni, ami belülről lett kezdeményezve és csak válasz forgalom.
- Forgalomfigyelés (traffic inspection):** forgalomszűréshez hasonlóan az Application layer-beli paramétereket is képes vizsgálni, megfelelő formátumú-e az adat, a sequence number megfelelő-e, ezek alapján a kimenő és a visszajövő forgalmat is tudja ellenőrizni.
- Alerts, audit trails:** figyelmeztetések és statisztikai információkat küldése

SPI és CBAC

Statefull Packet Inspection (SPI) a Cisco IOS szoftverben Context Based Access Control (CBAC) néven lett bevezetve. CBAC biztosítja, hogy egy ACL-en csak az a forgalom lesz átengedve, amelyet az ACL explicit módon megenged, vagy bármely belülről kifele menő forgalomra jövő válasz.



Konfigurálni az alábbi módon lehetséges:

! ACL 110 blokkol minden forgalmat

```
Router(config)# access-list 110 deny ip any any
```

! FW-in-out nevű tűzfal a tcp, udp és icmp forgalmat figyeli, majd a válasz forgalmat engedélyezi

```
Router(config)# ip inspect name FW-in-out tcp
Router(config)# ip inspect name FW-in-out udp
Router(config)# ip inspect name FW-in-out icmp
```

! FW-in-out a LAN interfészre kerül rá, a LAN felől jövő forgalmat figyeli

```
Router(config)# interface Fa0/0
Router(config-if)# ip inspect FW-in-out in
```

! minden bejövő forgalmat tiltunk a WAN felől. (A válasz forgalmat az ACL nem blokkolja)

```
Router(config)# interface S0/0
Router(config-if)# ip access-group 110 in
```

Az „ip inspect ...” parancsot nem feltétlen a bejövő interfészre „in” irányban kell konfigurálni, lehet a kimenő interfészen „out” irányban is.

Hagyományos, „statikus” csomagszűrés korábban csupán a Network és Transport (L3 és L4) layerre terjedt ki.

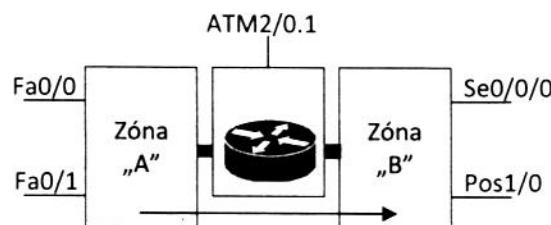
Kapcsolódások számának szabályzásával a férgek és automatikus támadások ellen lehet fellépni. Ezzel meg lehet szabni, hogy adott időn belül, adott hoszt hány kapcsolatot kezdeményezhet.

Zone-Based Firewall

A hagyományos interfész alapú tüzfal konfiguráció nem volt elég flexibilis, számos újítást nem lehetett benne megoldani. Ezért vezették be a zóna alapú tüzfal konfigurációt. Parancsai eltérnek az interfész alapú konfigurációtól, és a két módszert nem lehet egyszerre alkalmazni ugyanazon az interfészen. Ha egy interfész a CBAC tüzfallal van konfigurálva, nem lehet zone-based tüzfalhoz konfigurálni, nem lehet zóna tagja.

Egy interfész zóna tagságát az alábbiak jellemzik:

- mielőtt egy interfész egy zónához lehet rendelni, a zónának konfigurálva kell lennie
- azonos zónában levő interfések közötti forgalom implicit módon engedélyezve van
- egy zónában levő interfész, és egy zónához nem rendelt interfész közötti forgalom soha nem lesz engedélyezett, a forgalom mindenkor eldobódik. Zónák között lehet csak a pass, inspect vagy drop-ot beállítani.
- minden forgalom implicit módon tiltva van, kivéve az azonos zónán belüli interfések közötti forgalom, és a router interfészeire menő forgalom (self-zone)
- egy interfészt maximum egy zónához lehet hozzárendelni
- egy zónából menő és oda visszajövő forgalom engedélyezéséhez egy allow vagy inspect policy kell a két zóna közé
- a router interfészeire menő forgalom engedélyezve van
- Ha egy interfész nincs egyetlen zónához se rendelve, akkor tagja lehet a klasszikus CBAC tüzfalnak.
- Ha a forgalomnak minden interfész között engedélyezve kell lennie, akkor minden interfész tagja kell legyen valamelyik zónának.
- A zóna-pár (**zone-pair**) konfigurációja egyirányú (**unidirectional**). Ha „A” zóna forgalma engedélyezve van „B” zónába, akkor csak a válasz forgalom lesz engedélyezve „B” zónából „A” zónába. Ha a két zóna között kétirányú átjárásról akarunk biztosítani, akkor két unidirectional policy-t kell létrehozni.



A fenti példában az Fa0/0 és Fa0/1 az „A” zónába van konfigurálva, a Se0/0/0 és Pos0/0 interfések a „B” zónába, még az ATM2/0.1 interfész egyetlen zónához sincs hozzárendelve.

Zone-based firewall esetén a tüzfalszabály mindenkor a zóna-pár között van. Példánkban az „A” zónából indított forgalom a „B” zónába, illetve az ahhoz tartozó válaszforgalom lesz engedélyezve.

Zóna-párokhoz egy policy-t kell konfigurálni, melyben a class-okkal (vagy akár parameter map-ekkel) tovább lehet finomítani a tüzfal szabályait.

Tüzfal szabályok

Inspect: a class forgalmának figyelése és továbbengedése, és a válasz forgalom engedélyezése.

Pass: az adott class forgalmának továbbengedése. Válasz forgalom nincs automatikusan engedélyezve.

Drop: a class forgalmának eldobása

Inspect esetén nincs szükség két zóna-pár konfigurálására. Pass esetén mindenkor irányba egy-egy zóna-párt és hozzá tartozó szabályt konfigurálni kell, mivel a válasz forgalom nincs automatikusan engedélyezve. A forrás- és célpontok is ismertek kell legyenek.

A zone-based firewall konfigurálásának lépései

- zónák definiálása
- interfészek zónához rendelése
- class-map-ek konfigurálása
- policy-map konfigurálása
- zóna-párok konfigurálása

```

Router(config)# zone security A
Router(config)# zone security B

Router(config)# interface fa0/0
Router(config-if)# zone-member security A
Router(config-if)# interface fa0/1
Router(config-if)# zone-member security A
Router(config-if)# interface se0/0/0
Router(config-if)# zone-member security B
Router(config-if)# interface Pos1/0
Router(config-if)# zone-member security B

Router(config)# class-map type inspect match-any CM-web-traffic
Router(config-cmap)# match protocol http
Router(config-cmap)# match protocol https

Router(config)# policy-map type inspect PM-A-to-B
Router(config-pmap)# class type inspect CM-web-traffic
Router(config-pmap-c)# inspect
Router(config-pmap-c)# class type inspect class-default
Router(config-pmap-c)# drop

Router(config)# zone-pair security ZP-A-to-B source A destination B
Router(config-sec-zone-pair)# service-policy type inspect PM-A-to-B

```

Példánkban a http és https forgalom engedélyezett. minden más forgalom a class-default-ba esik, amire eldobódik.

Ellenőrzéshez a „show zone security”, „show zone-pair security”, „show policy-map type inspect”, parancsok használhatóak.

```

Router# show zone security
zone self
    Description: System defined zone

zone A
    Member Interfaces:
        FastEthernet0/0
        FastEthernet0/1

zone B
    Member Interfaces:
        Serial0/0/0
        POS1/0

Router# show zone-pair security
Zone-pair name ZP-A-to-B
    Source-Zone A Destination-Zone B
        service-policy PM-A-to-B

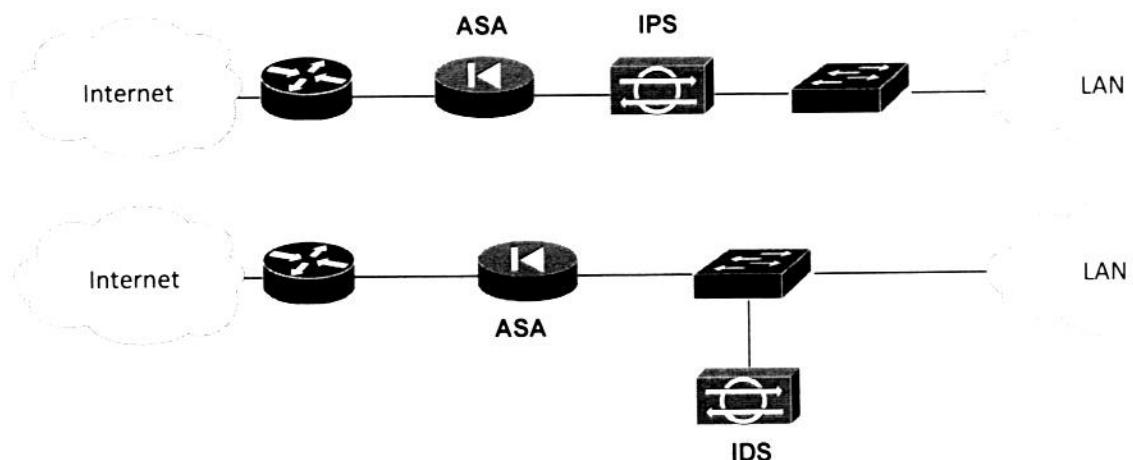
Router# show policy-map type inspect
    Policy Map type inspect PM-A-to-B
        Class CM-web-traffic
            Inspect
        Class class-default
            Drop

Router#

```

Intrusion Prevention System (IPS) / Intrusion Detection System (IDS)

Az IPS rendszerek a behatolást képesek megállítani, szemben az IDS-ekkel (Intrusion Detection System), melyek csupán érzékelni tudják, és erről egy riasztást generálnak. Ez természetesen nem az IDS rendszerek gyengeségét jelenti, más-más szerepe van egy IDS és IPS rendszernek a hálózatban. Az IDS-ek dinamikusan képesek a hálózati forgalmat figyelni, és abból következtetéseket levonni, azonban ez esetben a behatolás, káros tartalom már a hálózaton belülre kerül. Az IPS a hálózatban „inline” módon kerül, az IPS pedig csupán a forgalomról egy másolatot kap. Az IDS és IPS rendszerek elhelyezése a hálózatban az alábbi ábrákon láthatóak:



A hálózati (inline) IPS rendszerek előnye, hogy a hosztokon nem kell semmilyen szoftvert telepíteni, azonban a SSL csatornákban menő forgalom analizálására nem képesek. A hosztokra telepített IPS esetén az SSL tunnelbe küldött adatok kiértékelése még az encryptálás előtt megtörténik.

Hosztokra telepíthető IPS szoftver a Cisco Security Agent (CSA).

NIPS – Network Based Intrusion Prevention System

HIPS – Host Based Intrusion Prevention System

Detectálási módszerek

- **signature-based:** legelterjedtebb módszere a detectálásnak. Az eszközön átáramló adatot figyelve ismert kártevők kódjára, exploitokra egyezőséget keres. Például a WEB szerver elleni támadásokat a nem szabályosan megformázott URL-eken keresztül ismeri fel.
- **policy-based:** valamilyen szabály alapján érzékeli a behatolást. Ha egy hálózat nem kommunikálhat egy másik hálózattal, azonban ezt mégis megtesz, ez a policy-based módszerrel kiszűrhető.
- **anomaly-based:** átlagostól („normál”) eltérő forgalmi karakterisztikákat figyelve próbálja megállapítani a behatolást, a kártékony tevékenységet, behatolást. Ez lehet statisztikai, vagy a hálózati rendszergazda által, kézzel beállított érték.
- **honey pot:** egy éles rendszerhez hasonló, kedvező támadási felületet biztosít. Ha a támadó ezt a rendszert támadja meg, a viselkedésére, a támadás forrására, módjára lehet következtést levonni, és további szabályokat alkalmazni az éles rendszerhez.

IDS és IPS interfészei

Az IDS/IPS rendszereken az alábbi két fajta interfész található meg:

- **command and control:** eszköz menedzselésére, külső kommunikációra való, IP címmel ellátott interfész
- **monitoring:** legalább egy, a figyelendő hálózati forgalmat fogadó interfész

Monitoring interfész működési módjai:

- **promiscuous:** a hálózati forgalomról egy másolatot kap, és detektálás esetén riasztást küldhet, vagy más eszközöket utasíthat, hogy a vonatkozó forgalmat dobják el.
- **inline:** a forgalom útvonalába beékelődve figyeli a forgalmat. Legalább két interfész (logikai vagy fizikai) szükséges hozzá. Káros forgalmat a cél elérése előtt el lehet dobni.

Signature Definition File (SDF)

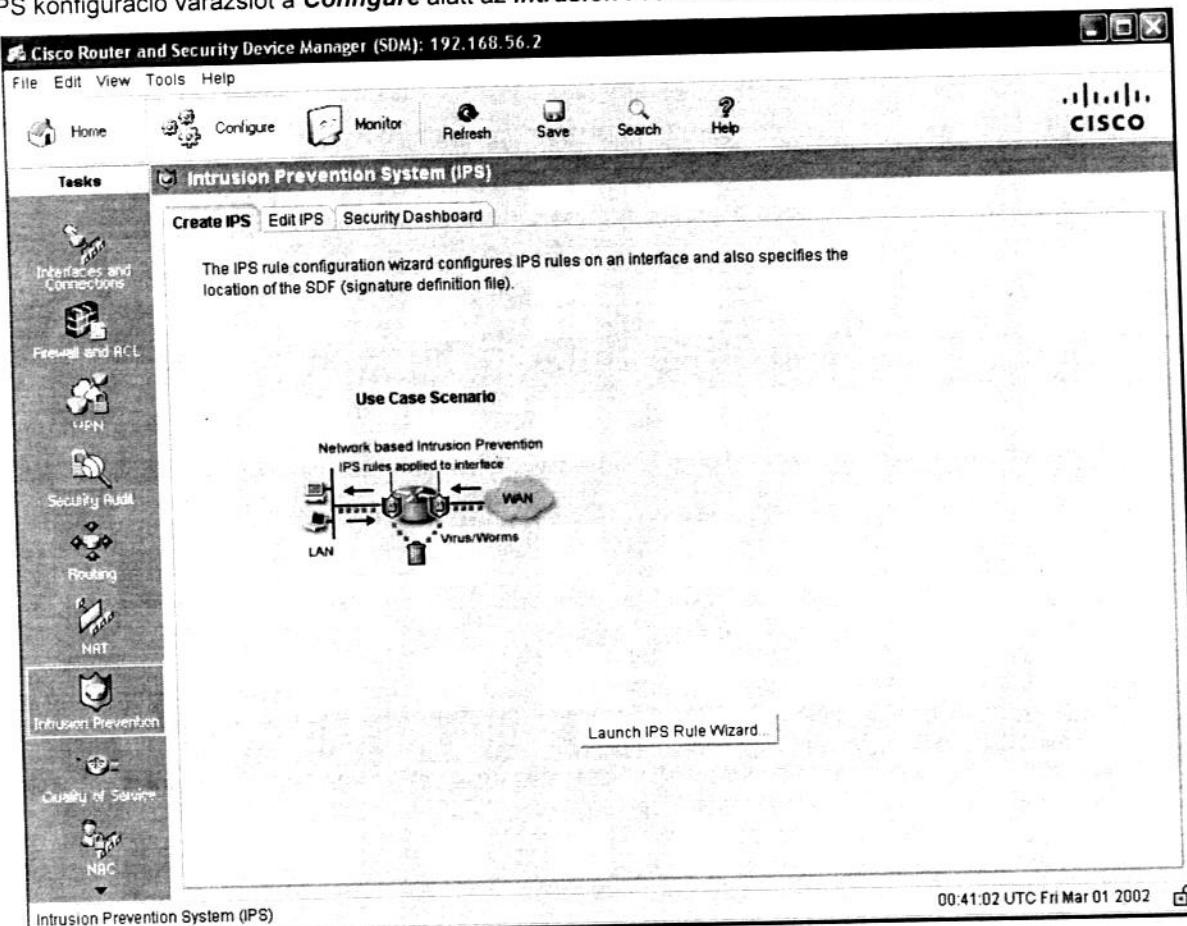
A signature-based detektálásnál az eszközök egyfajta adatbázist használnak a kártekny forgalom felismeréséhez. A Cisco IOS rendelkezik egy beépített signature-rel. Ezt ki lehet bővíteni külső signature fájlokkal, melyeknek .sdf kiterjesztése. Cisco két .sdf fájl ajánl, a router memóriakapacitásától függően. Ezek a 128MB.sdf (kb 300 signature) és a 256MB.sdf (kb 500 signature). Egy-egy signature-t finomhangolhatunk a (signature tuning) is, vagy akár ki is kapcsolhatunk, hogy jobban megfeleljön az aktuális hálózatnak. Új signature fájl építésére is van lehetőség.

Ha egy signature-nek alapján káros forgalom lesz érzékelve (signature firing), az alábbiakat teheti az eszköz:

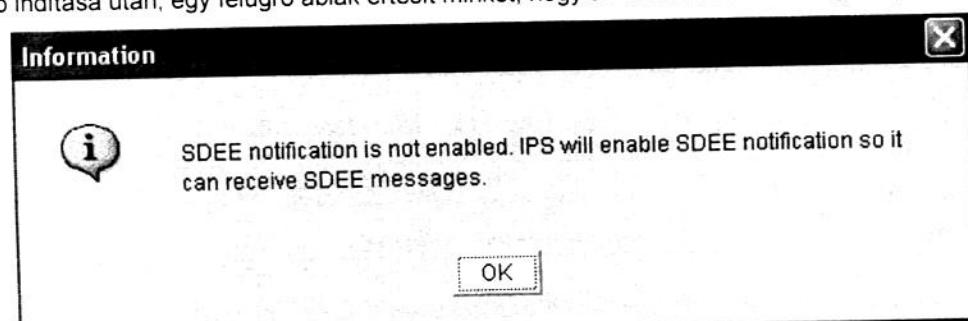
- log bejegyzés küldése
- eldobja a csomagot
- TCP kapcsolat resetelése: TCP reset-et küld minkét fél számára
- támadó IP címének blokkolása: minden forgalom blokkolva lesz az IP címről
- a kapcsolathoz tartozó forgalom blokkolása: csak a káros tartalomhoz tartozó forgalom lesz blokkolva

IPS konfigurálása SDM-en keresztül

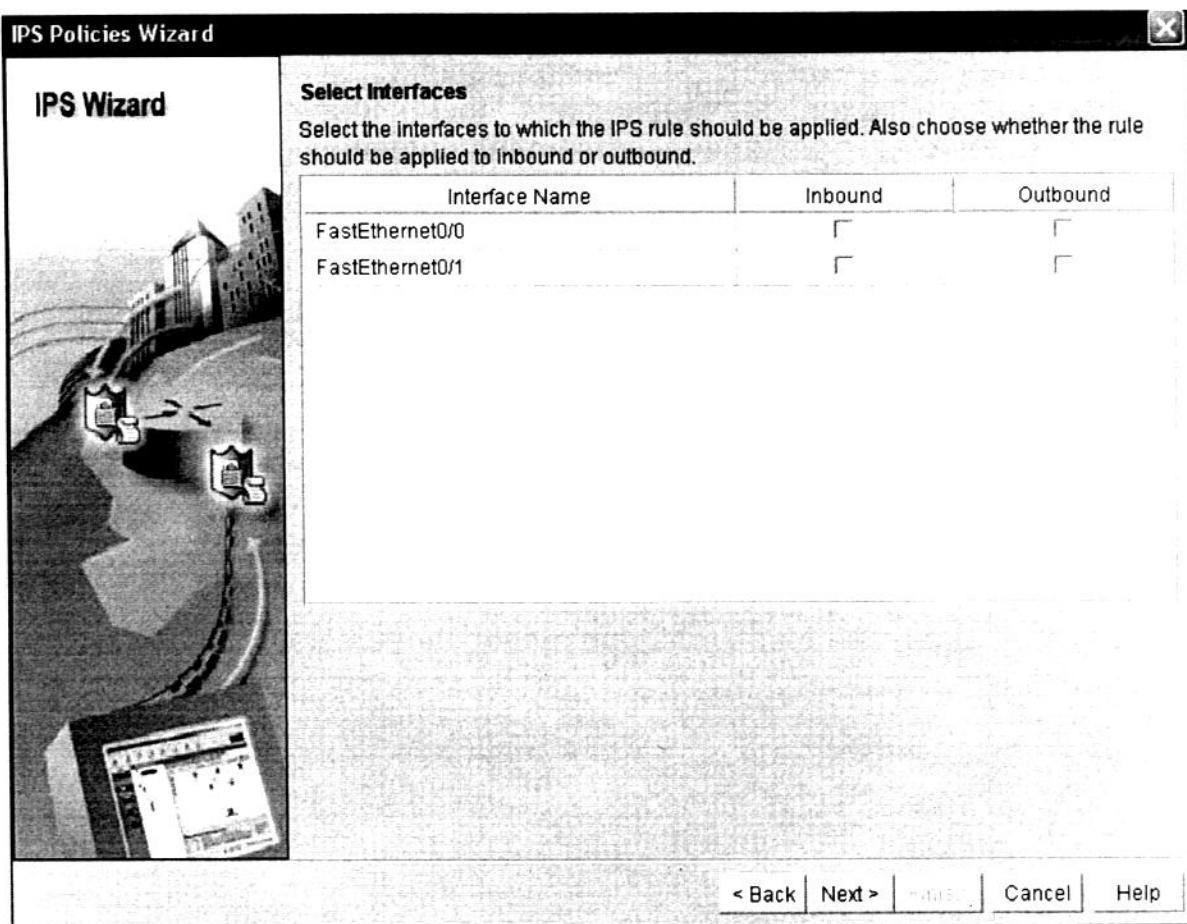
IPS konfiguráció varázslót a **Configure** alatt az **Intrusion Prevention** menüben lehet elindítani.



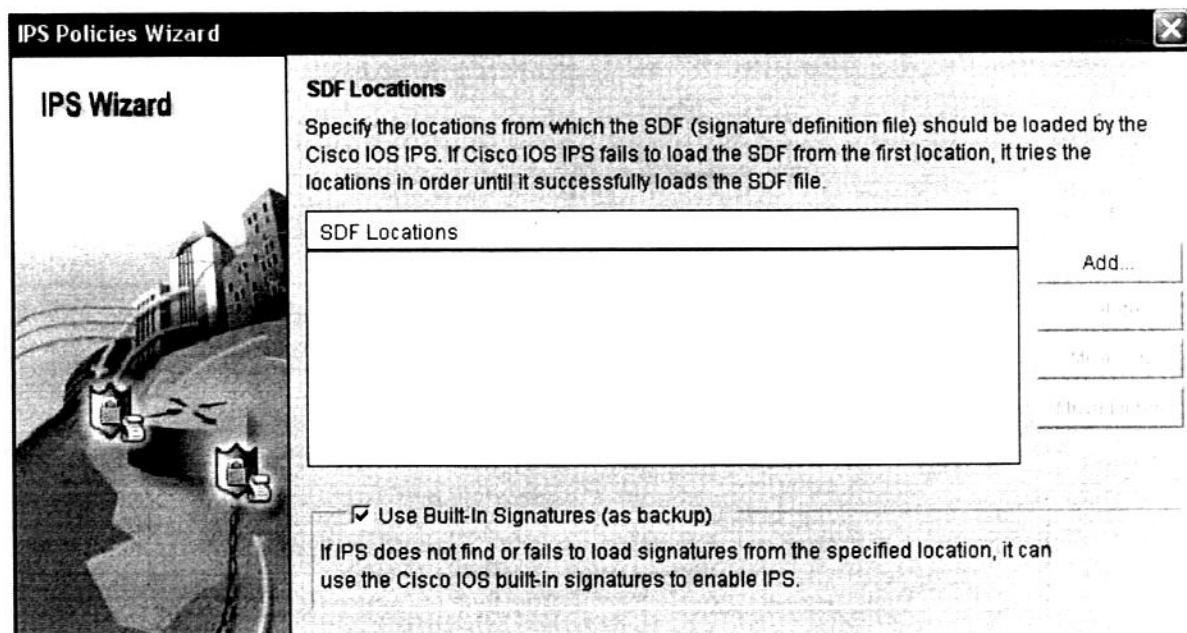
A varázsló indítása után, egy felugró ablak értesít minket, hogy az SDEE üzenetek engedélyezve lesznek:



A varázsló üdvözlő képernyője után ki kell választani, hogy mely interfészen, vagy interfészeken és milyen irányban legyen a figyelés engedélyezve.

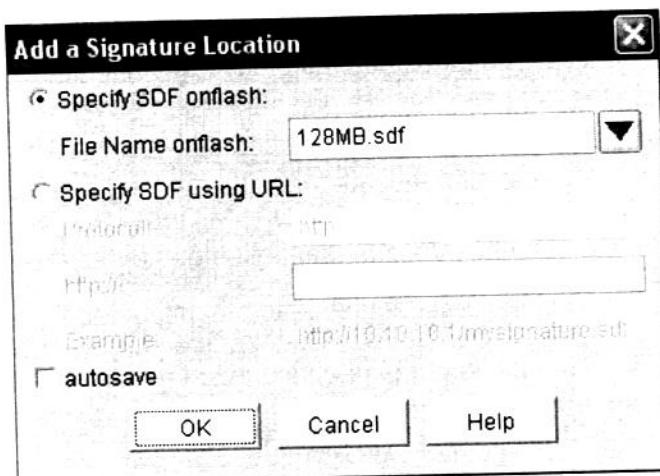


A következő ablakon az SDF fájlokat lehet megadni. Az „Use Built-In Signatures (as backup)” bejelölésével, ha nem sikeres az SDF fájlok betöltése, vagy nincs SDF fájl, akkor az IOS-be beépített signature-t fogja használni.

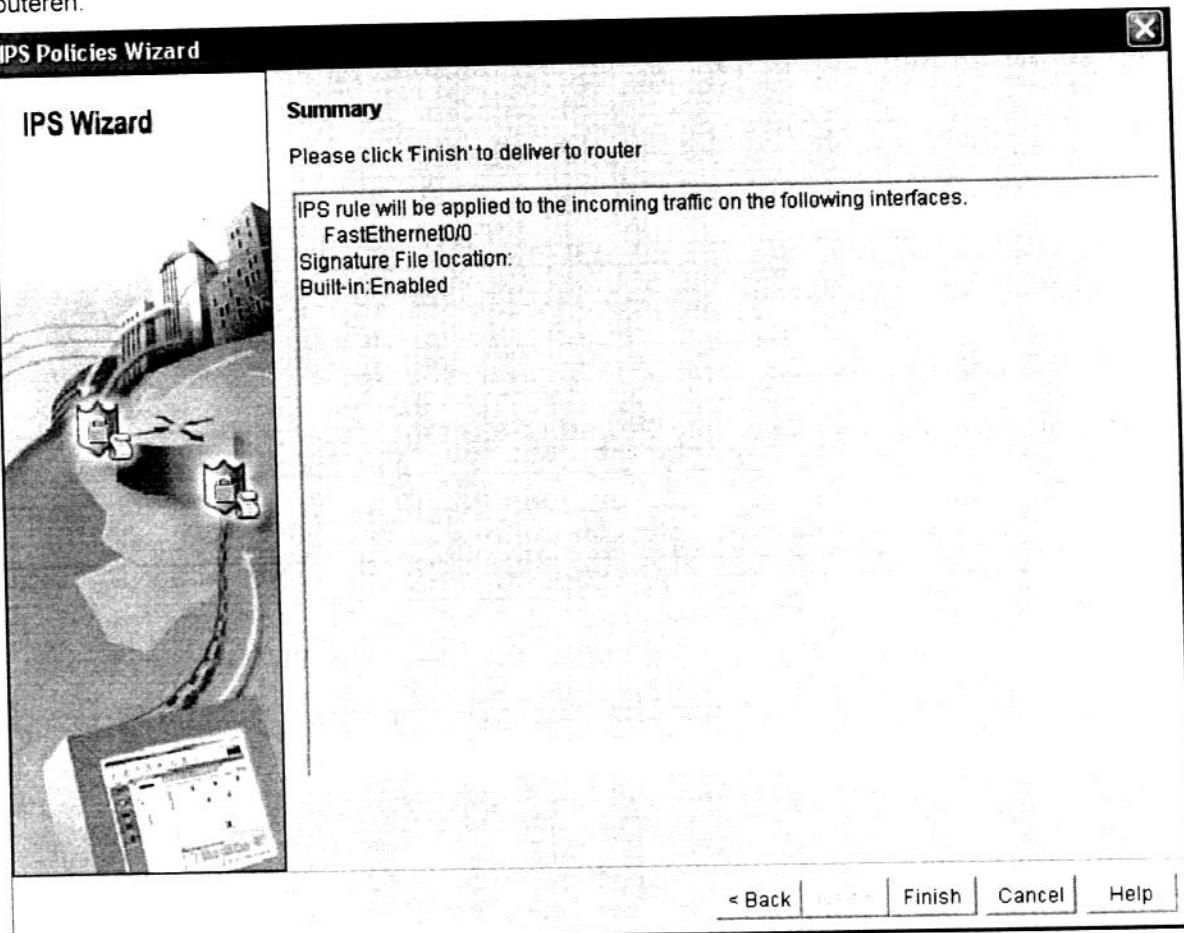


Amennyiben nem az IOS-be épített signature-t akarjuk használni, az „**Add...**” gombra kattintva adhatjuk meg, hogy honnan kívánjuk a használandó SDF fájlt betölteni. Erre az alábbi opciók vannak:

- flash memorián levő SDF fájl
- http, https, tftp, rcp, scp protokollon keresztül letöltendő fájl



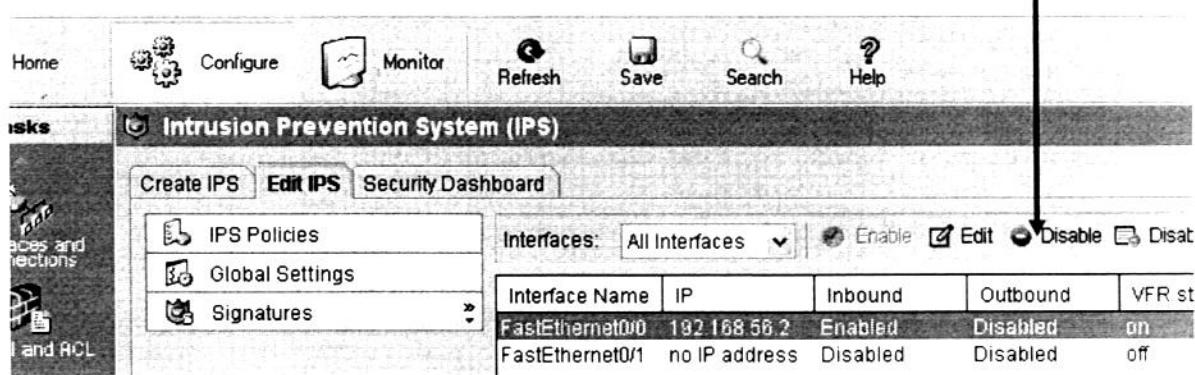
Utolsó lépésként egy összefoglalást látunk, majd a „**Finish**” gombra kattintva engedélyeztetjük az IPS-t a routeren.



SDEE üzenetek mellett a HTTPS is engedélyezve lesz, mivel az SDEE üzenetek cserélye HTTPS-en keresztül megy.

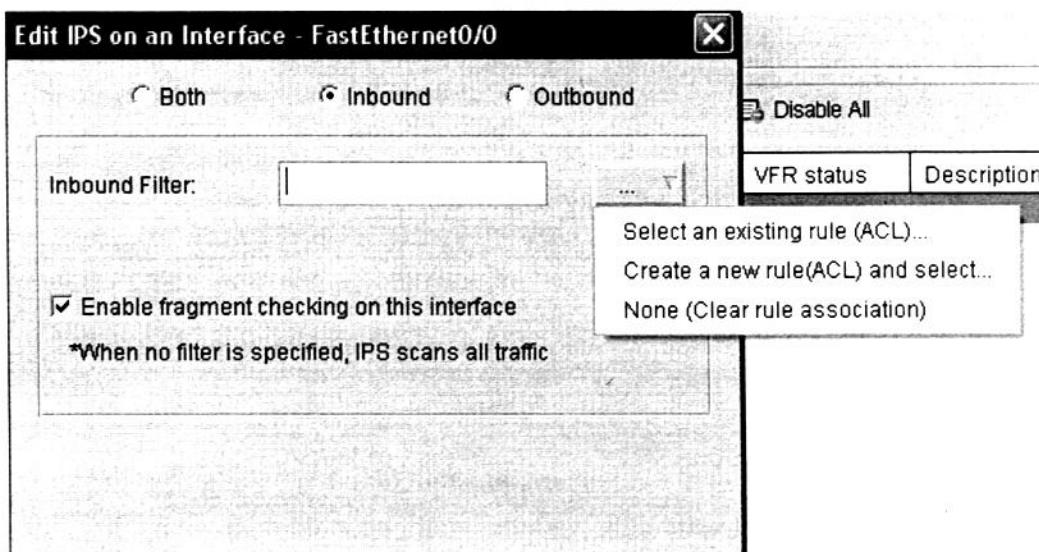
IPS finomhangolása

Alapértelmezésként minden forgalom az interfészen vizsgálva lesz. Szűrőket konfigurálni a **Configure-on belül** az „**Intrusion Prevention->Edit IPS**” fülön lehet. Ehhez az interfészt ki kell választani és az „**Edit**” gombra kattintva lehet a filterhez tartozó ACL-t konfigurálni.



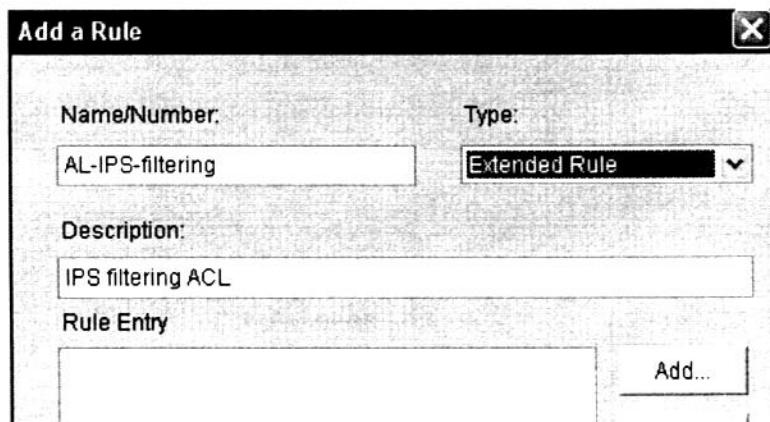
Ezek az ACL-ek szabják meg, hogy mely forgalom legyen vizsgálva. Konfigurációja SDM-ben az alábbi:

Az „**Inbound Filter**”-nél a jobb oldalon legördítve lehetőség van egy meglévő ACL-t (*Select an existing rule (ACL)...*) kiválasztani, egy újat készíteni (*Create a new rule(ACL) and select...*), vagy a meglévő szűrést leszedni.



Az „*Enable fragment checking on this interface*” bekapcsolása a *fragmentation* támadások ellen véd. A csomag újraépítésével a töredék csomagok is vizsgálatra kerülnek. (*ip virtual-reassembly*)

Új ACL létrehozásánál meg kell adni az ACL számát vagy nevét (*Name/Number*), típusát (*Type*) – Standard vagy Extended –, opcionálisan egy leírást az ACL-ről, és az „*Add...*” gombbal az ACL sorait létrehozni.



ACL bejegyzés létrehozásánál meg kell adni az akciót (Permit v. Deny) a forrás és cél hálózatot/hosztot, protokollt.

Add an Extended Rule Entry

Action	Description		
Select an action	Permit		
Source Host/Network			
Type:	A Network	Destination Host/Network	
IP Address:	192.168.21.0	Type:	A Host Name or IP Address
Wildcard Mask:	0.0.0.255	Host Name/IP:	172.19.1.44
(Mask bit 0 - Must match) (Mask bit 1 - Don't care)			
Protocol and Service			
<input type="radio"/> TCP <input type="radio"/> UDP <input type="radio"/> ICMP <input checked="" type="radio"/> IP			
IP Protocol			
IP Protocol ip ...			
<input type="checkbox"/> Log matches against this entry			
<input type="button" value="OK"/>		<input type="button" value="Cancel"/>	<input type="button" value="Help"/>

Amennyiben TCP vagy UDP a protokoll, lehetőség van a portot is megadni.

Protocol and Service			
<input checked="" type="radio"/> TCP <input type="radio"/> UDP <input type="radio"/> ICMP <input type="radio"/> IP			
Source Port (Rarely changed. See help)			
Service	=	any	...
Destination Port			
Service	=	telnet	...
<input type="checkbox"/> Log matches against this entry			
<input type="button" value="OK"/>		<input type="button" value="Cancel"/>	<input type="button" value="Help"/>

A „Log matches against this entry” bekapcsolásával, egyezés esetén log üzenetet generál a router.

IPS általános beállítások

A „Global Settings” alatt találjuk az IPS globális beállításait.

Item Name	Item Value
Syslog	Enabled
SDEE	Enabled
SDEE Alerts	200
SDEE Messages	200
SDEE Subscription	1
Engine Options	
Fail Closed	Disabled
Use Built-in Signatures (as backup)	Enabled
Deny Action on IPS interface	Enabled
Shun Event	
Timeout	30

A jobb felső részen levő Edit gombra kattintva szerkeszthetjük ezeket.

Az „**Enable Engine Fail Closed**” bekapcsolásával, ha az IOS új signature-t épít, és nem képes feldolgozni a csomagokat, azokat eldobja, ahelyett hogy továbbengedné.

Signature tuning

A „*Signatures*” alatt lehetséges egy-egy signature paraméterének megváltoztatása. Bal oldalon a különböző kategóriák szerint szűrhető a lista.

Enabled	Sig ID	SubSig ID	Name	Action	Severity
<input checked="" type="checkbox"/>	3153	0	FTP Improper Address	alarm	medium
<input checked="" type="checkbox"/>	2010	0	ICMP Info Rply	alarm	informational
<input checked="" type="checkbox"/>	3152	0	FTP CWD ~root	alarm	medium
<input checked="" type="checkbox"/>	5118	0	WWW eWave ServletExec File	alarm	high
<input checked="" type="checkbox"/>	3151	0	FTP SYST	alarm	informational

Egy Signaturet kiválasztva és az „*Edit*” gombra kattintva, megjelennek annak paraméterei:

Name	Value
SIGID:	3153
SigName:	FTP Improper Address
SubSig:	0
Alapértelmezett értékeknél zöld négyzet van	■ AlarmInterval: <input type="text"/>
Változtatott értékeknél pedig piros gyémánt	◆ AlarmSeverity: <input type="button" value="low"/> ■ AlarmThrottle: <input type="text"/> ■ AlarmTraits: <input type="text"/> ■ BadPortCmdAddress: <input type="text"/> ■ BadPortCmdPort: <input type="text"/> ■ BadPortCmdShort: <input type="text"/> ■ ChokeThreshold: <input type="text"/>

Zölde jelölést az engedélyezett, pirossat a kikapcsolt, sárga jelzést pedig a változtatott értékű Signature vannak ellátva.

Enabled	!	Sig ID	SubSig ID	Name	Action	Severity	Engine
<input checked="" type="checkbox"/>		2010	0	ICMP Info Rply	alarm	informational	ATOMIC.ICMP
<input checked="" type="checkbox"/>		3152	0	FTP CWD ~root	alarm	medium	STRING.TCP
<input checked="" type="checkbox"/>		5118	0	WWW eWave ServletExec File	alarm	high	SERVICE.HTTP
<input checked="" type="checkbox"/>		3151	0	FTP SYST	alarm	medium	STRING.TCP
<input checked="" type="checkbox"/>		5117	0	WWW eWave ServletExec File	alarm	high	SERVICE.HTTP

A változtatások érvényesítéséhez az „*Apply changes*” gombra kell kattintani. Ezek után kerülnek a változtatások a routerre. A sárga jelzés az alkalmazás után eltűnik, az csak a szerkesztés idején jelzi az értékek változtatását.

A 12.4(11)T vagy nagyon IOS-ek az 5.x verziójú signature adatbázist használják.

Kriptográfia

Évszázadokkal vagy akár évezredekkel ezelőtt is megvolt az igény arra, hogy a titkos üzeneteket más (az ellenség) ne tudja elolvasni, illetve biztosítsák az üzenet származását, és sértetlenségét. Rejtjelezésre az alábbi módszereket használták:

- **helyettesítés (substitution cipher):** egy-egy betűt egy másikkal helyettesítének, azonban a betük gyakoriságából következetéssel könnyen feltörhető
- **Vigenère cipher:** helyettesítéshez hasonló, azonban bonyolultabb algoritmus alapján kódolták az üzenetet
- **transpozíció (transposition):** a betük sorrendje kerül felcseréléésre
- **one-time pad:** kriptográfiai szempontból az egyik legjelentősebb lépés, azonban alkalmazása a valós életben nehézkes, főleg a számítógépes feldolgozás során. Az elgondolásban egy véletlen karakterszorozatot használnak, ami a számítógépek matematikai alapja miatt nem generálható.

Kódolás (encryption) folyamata

A bemenő adatokat egy kulcs alapján kódolják, majd a nem biztonságos hálózaton átküldve a fogadó felnél dekódolnak.



Titkosítást az OSI modellen három rétegen lehet elvégezni.

- application layer: secure email, secure database (Oracle SQL*net)
- session layer: SSL, TLS
- network layer: IPsec

Ezeket akár együtt is lehet használni, nem záraják ki egymást a konfigurációik. Routeren a session és network layerben lehet a titkosítást elvégezni. Az application layerben titkosítás csak a végpontokon lehetséges.

Kriptoanalízis

Amióta titkosítás létezik, a titkosítás feltörésére irányuló tevékenység is létezik.

Brute-force támadásoknál a lehetséges kulcsok átlagosan 50%-át kell végig próbálni, hogy sikeres legyen a támadás.

Szimmetrikus és aszimmetrikus algoritmusok

Szimmetrikus algoritmusok esetén a kódoláshoz és dekódoláshoz használt kulcs ugyanaz, még

aszimmetrikus algoritmusoknál különböző, és egyikből nem lehet a másikat kalkulálni.

Aszimmetrikus algoritmusokat *Publikus kulcs* (*Publik-key*) algoritmusoknak is szokták nevezni, mivel az egyik kulcs publikus, bárki számára elérhető, a másik kulcs pedig csak a tulajdonosa számára kell elérhetőnek lennie.

Aszimmetrikus algoritmusok akár ezerszer lassabbak is lehetnek, mivel nagy számok faktoriálisával vagy diszkrét logaritmusával számolnak. Nagy mennyiségű adat gyors kódolására szimmetrikus algoritmusokat használják.

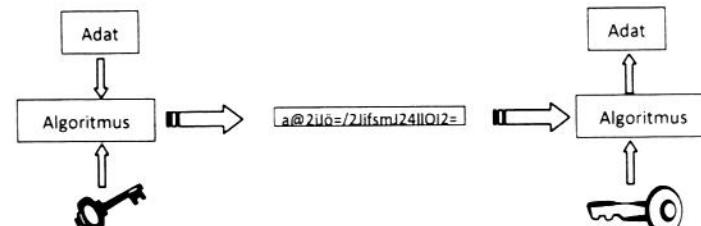
Szimmetrikus algoritmusok:

- DES, 3DES
- AES
- IDEA
- RC2, RC4, RC5, RC6
- Blowfish



Aszimmetrikus algoritmusok:

- RSA
- Diffie-Hellman
- Elliptical Curve



Szimmetrikus algoritmusokat a kódolásra, aszimmetrikus algoritmusokat pedig a szimmetrikus algoritmusok kulcs menedzsmentjére szokták használni.

Cipher típusok

- **Block cipher:** a kódolást az adatok összefüggő, fix hosszúságú blokkjain végzi az algoritmus. A blokk mérete adott, ezért a cipher-text mindig a blokk-méret többszöröse. Ha a kódolandó adat 28 bájt, a blokk mérete pedig 16 bájt, akkor a fennmaradó 4 bájtot (2 blokk használata miatt) automatikusan kiegészítí, így a cipher-text is hosszabb lesz miatta.
- **Stream cipher:** a kódolást sokkal kisebb általában bitenként végez el. Előnye a block cipherrel szemben, hogy gyorsabb és nem növeli meg a cipher-text méretét.

Block cipher algoritmusok: DES és 3DES (ECB, CBC módban), Blowfish, RSA, AES, IDEA, SAFER
 Stream cipher algoritmusok: RC4, DES és 3DES (OFB, CFB módban), SEAL

Kulcsok mérete

A szimmetrikus algoritmusok általában rövid, 40 és 256 bit közötti kulccsal dolgoznak. Az aszimmetrikus kulcsok esetén a kulcs mérete 512 és 4096 bit közötti szokott lenni.

DES és 3DES

DES 56 bites kulcsot használ. Ezt kiegészíti 8 bittel, amit paritásra használ. Blokk módban a blokk mérete a kulcsnak megfelelően így 64 bit lesz. DES képes blokk és stream módban is kódolni.

3DES tulajdonképpen a DES háromszori futtatása, három különböző kulccsal. A 3DES lépései az alábbi:

1. kódolás az első kulccsal
2. dekódolás a második kulccsal
3. kódolás a harmadik kulccsal

Ez egy 168 bit hosszú kulccsal történő kódolásnak felel meg. Ha az első és harmadik kulcs ugyanaz, akkor a kulcs hosszúsága csupán 112 bit. A decryptálásnál a folyamat fordítottját végzi el:

1. dekódolás a harmadik kulccsal
2. kódolás a második kulccsal
3. dekódolás az első kulccsal

Ennek a folyamatnak ***DES-encrypt-decrypt-encrypt (DES-EDE)*** a másik megnevezése.

DES és 3DES használata nem ajánlott, mivel könnyen feltörhető.

AES

Az AES általában 128, 192 vagy 256 bit hosszúságú kulcsot használ. A blokk vagy a klucz (vagy minden kettő) 32 bit többszörösére könnyen módosítható, emiatt a hardveres és szoftveres feldolgozása (32, 64 bit-es processzorok) jól támogatott.

Az AES a DES-hez képest erősebb és gyorsabban is fut. Ezért AES használata javasolt.

SEAL

Szoftveres kódolásnál a SEAL 160 bit-es kulccsal ideális választásnak tűnik. A CPU használata a DES, 3DES és AES algoritmusokhoz képest alacsonyabb.

Attól hogy egy algoritmus új, még nem biztos hogy biztonságos, még nem fedezték fel a hibáit. Egy régi algoritmus, ami az évek során bizonyította a sérthetetlenségét, megbízhatóbbnak mondható.

Avalanche effektus

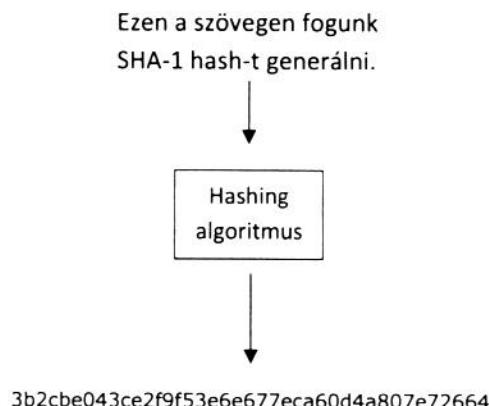
Egy vagy néhány bit megváltoztatásával a generált hash értéke teljesen más lesz.

További előnye egy algoritmusnak, ha az avalanche effektus jellemző rá.

Hash

A hashelés egy egy-irányú matematikai művelet. Visszafejteni szinte lehetetlen.

Hashelés során, egy tetszőleges hosszúságú adatból, egy fix hosszúságú hash érték generálódik. Ez a CRC érték számításához hasonlítható, csak kriptográfiailag sokkal erősebb. Ha bármely bit megváltozik az adatban, a hash értéke más lesz. Mivel a hash értéke fix hosszúságú, több adatnak lehet ugyanolyan hash értéke. Két ugyanolyan hash értékkal rendelkező adatot találni azonban meglehetősen nehéz, és egy kulcs hozzáadásával a hash visszafejtése szinte lehetetlen.



Hasheléssel az adatok sértetlenségét lehet biztosítani, illetve még authentikáció során használják. Ha az adatátvitel után az általunk generált hash értéke más mint az eredeti hash érték, akkor átvitel közben módosultak az adatok. Ez lehet sérülés, vonalhiba, de akár szándékos beavatkozás is. Egyfajta ujjlenyomathoz lehet hasonlítani a hash értékeket.

Leggyakrabban használt hash algoritmusok:

- MD5 (128 bit) – „salt” hozzáadásával komplikálható a hash értéke
- SHA-1 (160 bit)

Hash-based Message Authentication Code (HMAC)

Hash érték segítségével az adatok (üzenet – message) sértetlen megérkezésének tényét lehet megállapítani. Ha egy üzenet hash értékét egy titkos jelszó (amiről csak a két fél tud) hozzáadásával kiszámoljuk, majd a hash értékét és az üzenetet átküldjük, út közben, ha bárki módosítani akarja az üzenetet, tudnia kell a titkos jelszót is, hogy a hash értékét újra tudja számolni. E nélkül, mindenféle módosítás a hash értékből megállapítható.

Avalanche effektus a SHA-1 és MD5-nél

Szöveg	MD5	SHA-1
Ott egy pék.	02b2b5989bd9d4f703ed152f3fce1af3	d53a597008657cc2d9b0911cfb5093a225d93e43
Ott egy kép.	241621ea6cf6ff0528339d10ef3c95db	33e8271feb50091a810c78456ade9d4a435cdd10
Itt egy kép.	1c9ca1772aa0f6e32fd3d64ff2e176d	a8937c6cede7164e820cdc35160fe395c7ee082

Kulcs menedzsment

Kriptográfiai rendszereknél a kulcsok kezelése nagyon fontos. Ha a kulcs kiderül, vagy megfejtődik, akkor az adatokat nem lehet biztonságban, titkosítva átvinni a hálózaton. A támadások nagy része szintén a kulcs menedzsment ellen irányulnak, mintsem az alkalmazott algoritmusra.

Digitális Aláírások (Digital Signatures)

Digitális aláírásokat hasonlóan a kézzel írt aláírásokhoz, hitelesítésre használják. PKI (Public Key Infrastructure) rendszerek egyik alapeleme. Egy harmadik fél (CA) (mely minden két fél számára megbízható) közbeiktatásával történik meg a hitelesítés.

PKI rendszerek elemei:

- Certificate Authority (CA)
- Registration Authority (RA)

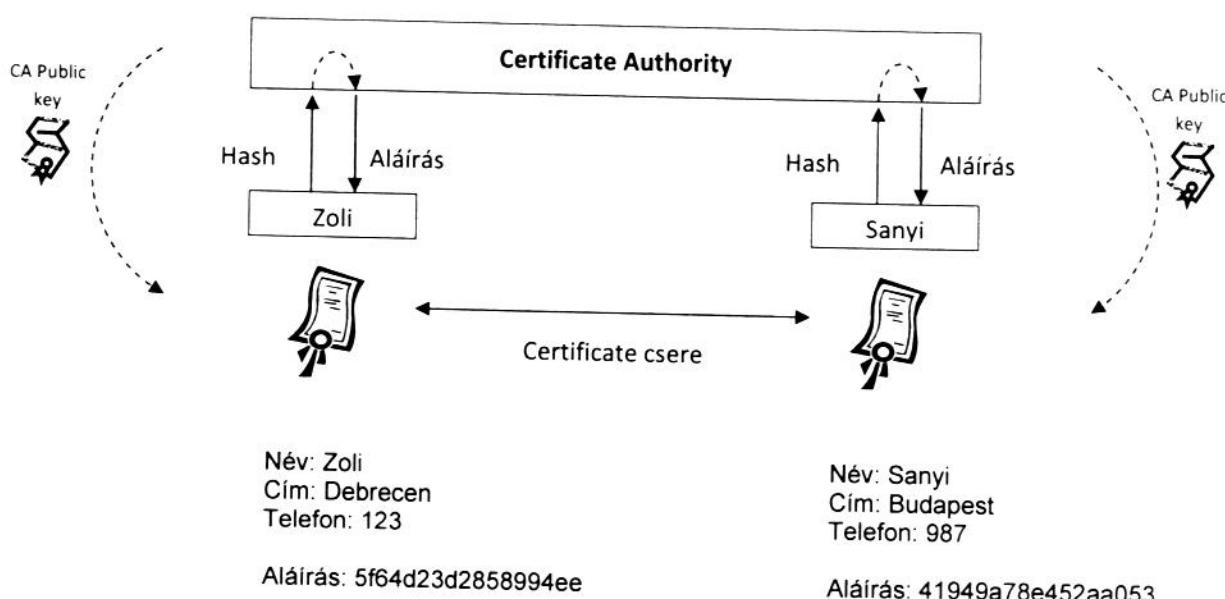
Az aszimmetrikus algoritmusok jellemzője, hogy a private-key-el kódolt adatokat csak a hozzá tartozó public-key-el lehet dekódolni, illetve fordítva, a public-key-el kódolt adatokat csak a hozzá tartozó private-key-el lehet dekódolni.

Digitális aláírás genrálásának folyamata: az egyik fél („A”) adataiból generált hash-t a CA kódolja a saját privát kulcsával. Az így kapott kódolt hash-t (digitális aláírás) pedig visszaküldi „A”-nak. Ha a CA megbízható valaki számára, a CA publikus kulcsával dekódolva „A” eredeti hash-ét kell megkapnia.

Authentikáció folyamata digitális aláírással:

1. „A” a CA-tól kapott kódolt hash értéket átküldi „B”-nek.
2. „B” legenerálja ugyanazokból az adatokból a hash értékét, majd az kapott digitális aláírást a CA publikus kulcsával dekódolja. Ha a két érték megegyezik, akkor „A” az, akinek mondja magát.

Kétoldali authentikáció esetén minden két félnek kell rendelkeznie digitális aláírással. Ennek hiányában az authentikáció csak egyoldalú lehet.



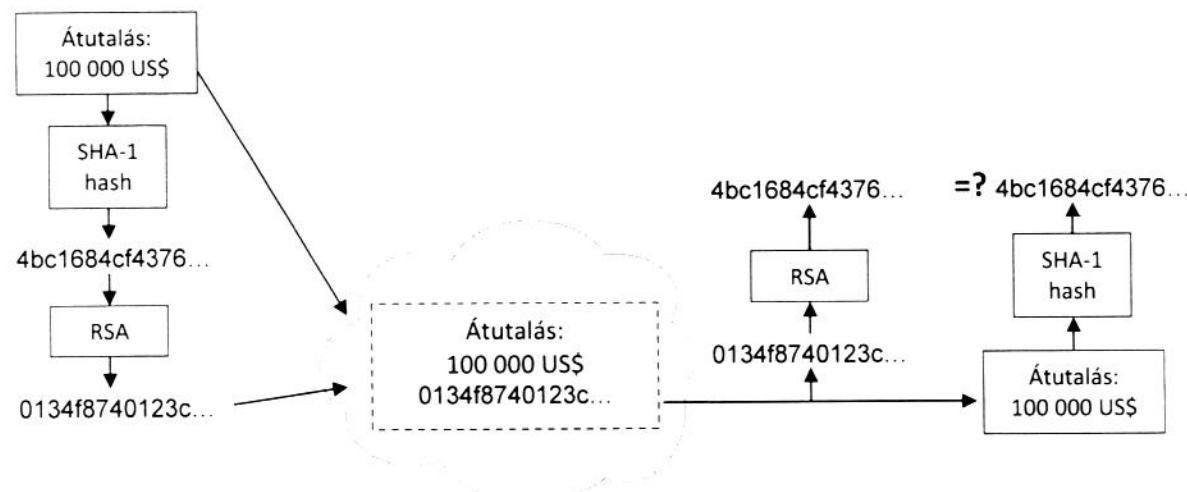
Egymás aláírását a CA publikus kulcsával dekódolva, ugyanazt az értéket kell kapniuk, mint a név, cím és telefon adatokból saját maguk által generált hash érték. Digitális tanúsítványoknál az adatok közé az RSA publikus kulcs is bekerül, biztosítva ezzel az RSA kulcs eredetiséget, hitelességét.

Ha az adatok titkosítására is szükség van, akkor a küldendő adatokat a másik fél publikus kulcsával kódolva, dekódolni csak a privát kulccsal lehet, mely csak a címzett birtokában van. Ezzel az elgondolással, ha valaki a privát kulcsával kódolja az adatokat, majd azt a publikus kulcsával küldi át a másiknak, biztosítható hogy az adatfolyam forrása ténylegesen az ahonnan jött, ugyanis a publikus kulcs csak a hozzá tartozó privát kulcsot tudja dekódolni. Ezáltal eredetiséget is lehet igazolni (*Origin authentication*).

RSA

Az RSA privát és publikus kulcsával dolgozik. A publikus kulcsot bárkihez lehet adni. A publikus kulccsal titkositott adatokat, csak a privátkulccsal lehet dekódolni. Az RSA az egyik olyan algoritmus, melyet mind aláírásra, minden pedig titkositásra is lehet használni. Tényleges alkalmazása azonban az aláírásokkal az authentikációra, és a szimmetrikus algoritmusokhoz kulcs-menedzsment (csere) terjed ki. **RSA sérthatóságát tekintve egyedül a BPA (Branch Prediction Analysis) és az adaptive chosen cipher text támadásokra érzékeny.**

RSA Digitális Aláírások használata



Az adatokból hash generálódik, melyet az RSA privát kulccsal encryptál, majd az encryptált hash értékkel kerül az adat átküldésre. A fogadó fél az adatokból legenerálja a hash értéket, a kapott encryptált hash értékét pedig a publikus RSA kulccsal decryptálja, majd a két értéket összehasonlíta.

Public Key Infrastructure (PKI)

CA (Certificate Authority): egy megbízható harmadik fél, mely a tanúsítványok (certificate) kiadásáért felelős.

RA (Registration Authority): CA több feladatait veheti át, azonban nincs annyi jogosultsága, mint a CA-nak.
Tanúsítvány (certificate): a publikus kulcsot és tulajdonosát köti össze, azonosítja be egyértelműen.

A CA is rendelkezik tanúsítvánnyal, amit saját maga ír alá (*self-signed*).

Ha a kliens nem képes magának kulcs-párt generálni, a CA vagy RA teszi meg helyette.

x509 szabvány: PKI rendszerek szabványosítása iránti igény miatt vezették be

PKCS (Public Key Cryptography Standards): public-key-en alapuló kriptografiát használó rendszerek közötti kompatibilitás miatt dolgozták ki. A CCNA Security vizsgához az alábbiakat érdemes megjegyezni.

Szabvány	Funkció
PKCS #7	Kriptográfiai üzenetek szintaksza (aláírások)
PKCS #10	Tanúsítvány kérési szintaksza

A PKCS#10 tanúsítványmérést (*certificate request*) a CA, ellenőrzés után x509-es formátumú aláírt tanúsítványt küld vissza.

SCEP (Simple Certificate Enrollment Protocol)

A különböző lehetőségek és megvalósítások miatt, szükségessé vált egy protokoll létrehozása, mely a tanúsítványok életciklusának számos elemét képes lekezelni (kérés, kibocsátás, jóváhagyás, visszavonás) Az IPsec protokollcsalád, tanúsítvány alapú authentikáció esetén is az SCEP protokollt használja a tanúsítványok kezelésére. Ez nagy előnyt jelent a hagyományos kézi, vagy fájl alapú tanúsítványkezeléshez. Pre-shared key használatával automatizálható a folyamat, nem kell a CA adminisztrátorak manuálisan jóváhagynia a tanúsítványt.

Certificate Revocation List (CRL)

Ha egy tanúsítványt valamelyen okból vissza kell vonni (ellopott privát-kulcs), a tanúsítvány felkerül a CRL listára. Ha egy CA tanúsítványa kerül fel a CRL listára, az összes általa kibocsátott tanúsítvány érvénytelenné kell tenni.

XAUTH (Extended Authentication)

Számos esetben a tanúsítványokkal történő authentikáció mellé további authentikációt megkövetelése szükséges. IPsec az XAUTH-al, one-time password tokennel vagy más módon történő további authentikációt követelhet meg.

Tanúsítványok használata

Digitális tanúsítványokat leggyakrabban az SSL (HTTPS), és biztonságos e-mail kapcsolatoknál használnak. Természetesen bárhol lehet használni őket, ahol authentikáció vagy enryptálás szükséges.

Site-to-Site VPN kapcsolatok

A bérelt vonalakkal költsége miatt, és az internet terjedésével az interneten kereszttüli VPN kapcsolatok kerültek előtérben. Mivel az internet egy nem biztonságos hálózat, ezért a bizalmas adatokat titkositva kell átküldeni rajta, illetve biztosítani, hogy módosítás nem történik rajtuk. Ezeket a kapcsolatokat VPN tunnelnek is nevezik, mivel mint egy csatorna halad át az interneten keresztül. A VPN tunnel több másik eszközön (hop-on) is kereszttühaladhat, azonban ez csak egy hop-nak látszik a hálózat számára.

VPN hálózatok elemei:

- **headend VPN device:** a központi telephelyen levő VPN eszköz
- **VPN access device:** távoli telephelyen levő VPN eszköz
- **Tunnel:** két telephelyet összekötő, virtuális kapcsolat
- **peer:** egymással összeköttetésben levő VPN eszköz

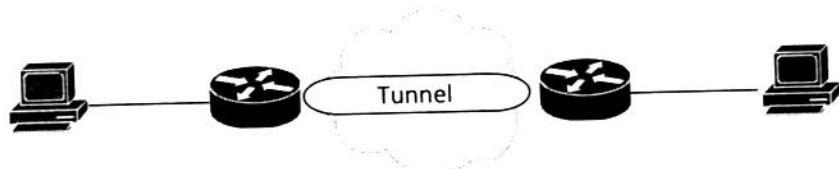
IPsec VPN

Az IPsec protokollcsalád (nyílt szabványokból) előnye a többi VPN kapcsolattal szemben, hogy titkításának köszönhetően az adatok bizalmas átvitelét is biztosítja.

IPsec előnyei:

- **bizalmasság (confidentiality):** adatok titkítása
- **sértetlenség (integrity):** checksum vagy hash értékkel az adatok sérzetlen megérkezését biztosítja
- **authentication:** megbizonyosodás a feladó kilétérből, az alábbi módok legalább egyikével:
 - felhasználói név-jelszó páros
 - one-time-password (OTP)
 - jelszó – Pre-shared key (PSK)
 - digitális tanúsítvány

IPsec az OSI modell harmadik rétegében működik, emiatt teljesen transzparens az alkalmazások számára.



A PC-k által küldött csomagokat a routerek encryptálják és küldik át a tunnellen. Ez a folyamat a PC számára teljesen észrevétenél zajlik. A routerek és a PC-k között az eredeti csomagok mennek, a két router között pedig az eredeti csomagok egy IPsec csomagba lesznek beágyazva. A PC-k számára a tunnel egy újabb direkt összeköttetésként látszik a két router között.

Azt a forgalmat, amelynek az IPsec kapcsolaton kell keresztlü mennie „*interesting traffic*”-nak nevezik. minden más forgalom azon kívül megy. Az „*interesting traffic*”-ot ACL-el lehet kiválasztani.

Kulccserét aszimmetrikus algoritmusok biztosítják. Még a titkosítást szimmetrikus algoritmusok végzik.

Diffie-Hellman key exchange (DH)

Diffie-Hellman algoritmust biztonságos kulcs-cserét valósít meg, nem biztonságos csatornán keresztlü. IPsec protokollcsaládot használó VPN hálózatok is a DH algoritmust használják az IKE (Internet Key Exchange) kulcs-cserénél. A folyamat indulásakor két nem titkos számban kell megegyeznie a két félnek (DH nonces).

IKE (Internet Key Exchange)

Az IPsec protokolcsalád egyik legfőbb protokollja az IKE. A kódoláshoz titkosító kulcsokra van szükség, melyet az IKE cserél ki rendszeres időközönként az authentikált felek (peer) között.

IKE módjai:

- **Main mode:**
 - a kapcsolat kezdeményezője (initiator) konfigurációs javaslatokat (proposal) küld a másik félnek (responder). A javaslatokban szerepel a támogatott enryptálási algoritmusok, authentikáció módja, és egyéb konfigurációs elemek. A responder ebből egyet kiválasztva visszaküldi az initiator-nak.
 - Következő lépésként a Diffie-Hellman-nal a nem biztonságos hálózaton keresztül kialakítanak egy titkos közös jelszót (szimmetrikus algoritmus számára).
 - az ISAKMP (*Internet Security Association Key Management Protocol*) session ezzel létrejött. Ezen a biztonságos csatornán keresztül létre lehet hozni az IPsec session-t.
- **Aggressive mode:**
 - ugyanazt az eredményt 3 üzenettel éri el, mint amit a Main mode. A három lépést három csomagba foglalja össze. Első csomagban az összes információt elküldi, ami a session létrehozásához kell. A második csomagban a responder küldi vissza az elfogadott paramétereket és az authentikációs adatokat, majd a harmadik csomag véglegesíti az ISAKMP session létrejöttét.
- **Quick mode:**
 - az ISAKMP session védelme alatt az IPsec session paramétereit egyezteti le és létrehozza az IPsec session-t.

IKE fázisai:

1. Main és Aggressive módot szokták Phase 1-nek nevezni
2. Quick mode pedig a Phase 2

Az IKE-nek van egy opcionális Phase 1.5-nek nevezett fázisa, amely XAUTH az ISAKMP csatorna védelmében további authentikációt követelhet meg, illetve a dinamikus IP konfigurációs beállítások is ekkor történnek meg.

Az IKE Phase 1 kétirányú csatorna, ugyanazt a csatornát használja minden irányba. Az IKE Phase 2 azonban egyirányú (unidirectional), így egy-egy csatornát épít fel a két fél között.

SA (Security Association): azon paraméterek, amelyek alapján a tunnel felépül illetve működik.

Authentication Header (AH) és Encapsulating Security Payload (ESP)

Az IKE-n kívül az IPsec az AH és ESP protokollokat is használja. Mindkettő biztosítja az authentikációt és az sértetlenséget a csomagnak (hash), azonban az ESP enryptálás miatt az adatok bizalmasságát is biztosítja. Mindkét protokoll képes tunnel és transport módban működni. A kettő közötti különbséget az alábbi ábrák mutatják:

Eredeti csomag:

IP header	Adatok
-----------	--------

Transport mód:

IP header	ESP header	Adatok	ESP trailer	ESP auth
-----------	------------	--------	-------------	----------

IP header	AH header	Adatok
-----------	-----------	--------

Tunnel mód:

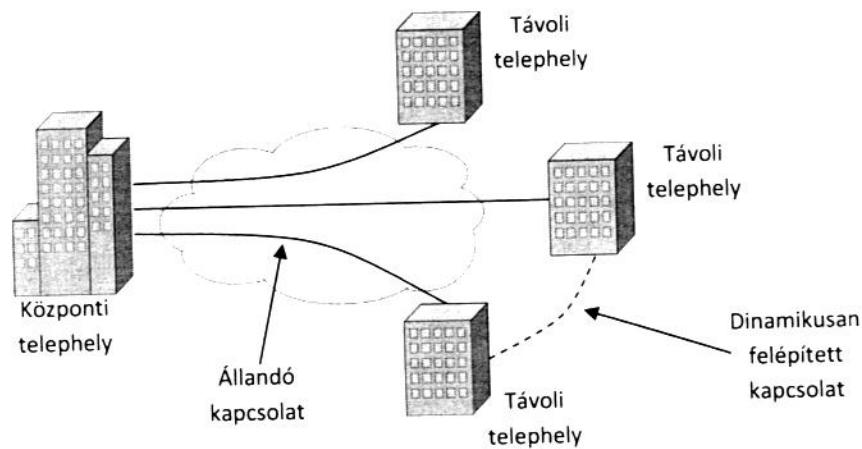
Új IP header	ESP header	IP header	Adatok	ESP trailer	ESP auth
--------------	------------	-----------	--------	-------------	----------

Új IP header	AH header	IP header	Adatok
--------------	-----------	-----------	--------

Dynamic Multipoint VPN (DMVPN)

A távoli telephelyek számának növekedtével, ha minden telephely, minden telephellyel össze kell legyen kötve, a konfigurálandó kapcsolatok száma drasztikusan meg tud növekedni. Tíz telephely esetén 45 IPsec tunnel szükséges, amely 90 IPsec peer konfigurációját jelenti (tunnelenként kettő, végpontonként kilenc másik végpont).

Ennek megoldásaként, a távoli telephelyek, egy vagy több központi telephellyel vannak állandó IPsec tunnellel összekötve. Ha két távoli telephely kommunikálni akar egymással, a központi telephelyen keresztül tehetik meg, ameddig közöttük dinamikusan ki nem alakul egy ideiglenes IPsec tunnel. Ha az ideiglenesen kialakított tunnelre nincs szükség, az automatikusan lebomlik.



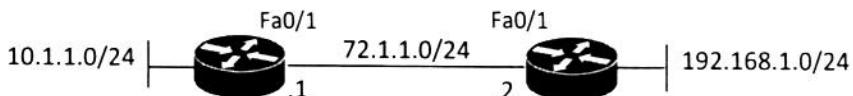
GRE (Generic Routing Encapsulation)

Számos protokoll broadcastet és multicastet használ. Az IPsec egyik nagy hátránya hogy csak unicast csomagokat tud átvinni. GRE használatával, a broadcast és multicast (vagy bármilyen más típusú) csomagok unicast GRE csomagként átküldhetők az IPsec tunnellen.

IPsec transport mód esetén, ha az adatokat GRE tunnelen keresztül küldjük át, akkor a GRE tunnel IP headerjét fogja használni a transport mód. Tulajdonképpen az adatok egy IPsec tunnelen keresztül küldött GRE tunnelen mennek keresztül..

IPsec használatával számolni kell az újonnan hozzáadott headerek és trailerek méretével is. Tunnel mód esetén ráadásul egy újabb IP headerrel növekszik a csomag.

IPsec konfigurációja



Első lépésként az ISAKMP kapcsolatot kell konfigurálni az alábbi módon:

```

Router1(config)# crypto isakmp key VPNjelsz0 address 72.1.1.2
Router1(config)# crypto isakmp policy 1
Router1(config-isakmp)# authentication pre-share
Router1(config-isakmp)# hash sha
Router1(config-isakmp)# encryption aes 128
Router1(config-isakmp)# group 2
Router1(config-isakmp)# lifetime 86400
  
```

A másik routert ugyanezzel a konfigurációval, azonban a másik router IP címével kell konfigurálni:

```

Router2(config)# crypto isakmp key VPNjelsz0 address 72.1.1.1
Router2(config)# crypto isakmp policy 1
Router2(config-isakmp)# authentication pre-share
Router2(config-isakmp)# hash sha
Router2(config-isakmp)# encryption aes 128
Router2(config-isakmp)# group 2
Router2(config-isakmp)# lifetime 86400
  
```

Az hash sha lesz, kódolás 128 bites aes, és Diffie-Hellman group 2-est fogja használni a router. A lifetime az ISAKMP életciklusát határozza meg, letelte után újra kell építeni.

Második lépében az IPsec paramétereket kell konfigurálni, mindenről mindenről az alábbi módon:

```

Router1(config)# crypto ipsec transform-set TS-ipsec esp-aes esp-sha-hmac
Router2(config)# crypto ipsec transform-set TS-ipsec esp-aes esp-sha-hmac
  
```

IPsec kapcsolat szintén aes titkosítást és sha hashinget fog használni.

Az így létrehozott Phase 1 és Phase 2-es konfigurációt egy **crypto map** köti össze, a forgalmat pedig egy ACL választja ki. Konfiguráció a két routeren az alábbi:

```

Router1(config)# access-list 101 permit ip 10.1.1.0 0.0.0.255 192.168.1.0 0.0.0.255
Router1(config)# crypto map CM-R1-R2 10 ipsec-isakmp
Router1(config-crypto-map)# set peer 72.1.1.2
Router1(config-crypto-map)# match address 101
Router1(config-crypto-map)# set transform-set TS-ipsec

Router2(config)# access-list 101 permit ip 192.168.1.0 0.0.0.255 10.1.1.0 0.0.0.255
Router2(config)# crypto map CM-R1-R2 10 ipsec-isakmp
Router2(config-crypto-map)# set peer 72.1.1.1
Router2(config-crypto-map)# match address 101
Router2(config-crypto-map)# set transform-set TS-ipsec
  
```

A két routeren a az ACL-t tükrözni kell a forgalom irányának megfelelően, illetve a peer címe a másik router címe.

A létrehozott crypto map-et pedig egy interfészre mindenről mindenről az alábbi módon kell rátenni:

```

Router1(config)# interface Fa0/1
Router1(config-if)# crypto map CM-R1-R2

Router2(config)# interface Fa0/1
Router2(config-if)# crypto map CM-R1-R2
  
```

Ellenőrzéshez a „show crypto session”, „show crypto isakmp sa”, „show crypto ipsec sa” parancsok használhatóak:

```

Router1# show crypto session
Crypto session current status

Interface: FastEthernet1/0
Session status: UP-ACTIVE
Peer: 172.16.1.2 port 500
IKE SA: local 72.1.1.1/500 remote 72.1.1.2/500 Active
IPSEC FLOW: permit ip 10.1.1.0/255.255.255.0 192.168.1.0/255.255.255.0
Active SAs: 2, origin: crypto map

Router1# show crypto isakmp sa
IPv4 Crypto ISAKMP SA
dst          src          state      conn-id slot status
72.1.1.2     72.1.1.1    QM_IDLE   1001    0 ACTIVE

IPv6 Crypto ISAKMP SA

Router1# show crypto ipsec sa

interface: FastEthernet1/0
Crypto map tag: CM-R1-R2, local addr 72.1.1.1

protected vrf: (none)
local ident (addr/mask/prot/port): (10.1.1.0/255.255.255.0/0/0)
remote ident (addr/mask/prot/port): (192.168.1.0/255.255.255.0/0/0)
current_peer 72.1.1.2 port 500
    PERMIT, flags={origin_is_acl,}
#pkts encaps: 4, #pkts encrypt: 4, #pkts digest: 4
#pkts decaps: 4, #pkts decrypt: 4, #pkts verify: 4
#pkts compressed: 0, #pkts decompressed: 0
#pkts not compressed: 0, #pkts compr. failed: 0
#pkts not decompressed: 0, #pkts decompress failed: 0
#send errors 11, #recv errors 0

local crypto endpt.: 72.1.1.1, remote crypto endpt.: 72.1.1.2
path mtu 1500, ip mtu 1500, ip mtu idb FastEthernet1/0
current outbound spi: 0x2D072368(755442536)

inbound esp sas:
spi: 0xA46F8439(2758771769)
transform: esp-aes esp-sha-hmac ,
in use settings ={Tunnel, }
conn id: 1, flow_id: 1, crypto map: CM-R1-R2
sa timing: remaining key lifetime (k/sec): (4435041/3141)
IV size: 16 bytes
replay detection support: Y
Status: ACTIVE

inbound ah sas:

inbound pcp sas:

outbound esp sas:
spi: 0x2D072368(755442536)
transform: esp-aes esp-sha-hmac ,
in use settings ={Tunnel, }
conn id: 2, flow_id: 2, crypto map: CM-R1-R2
sa timing: remaining key lifetime (k/sec): (4435041/3141)
IV size: 16 bytes
replay detection support: Y
Status: ACTIVE

outbound ah sas:

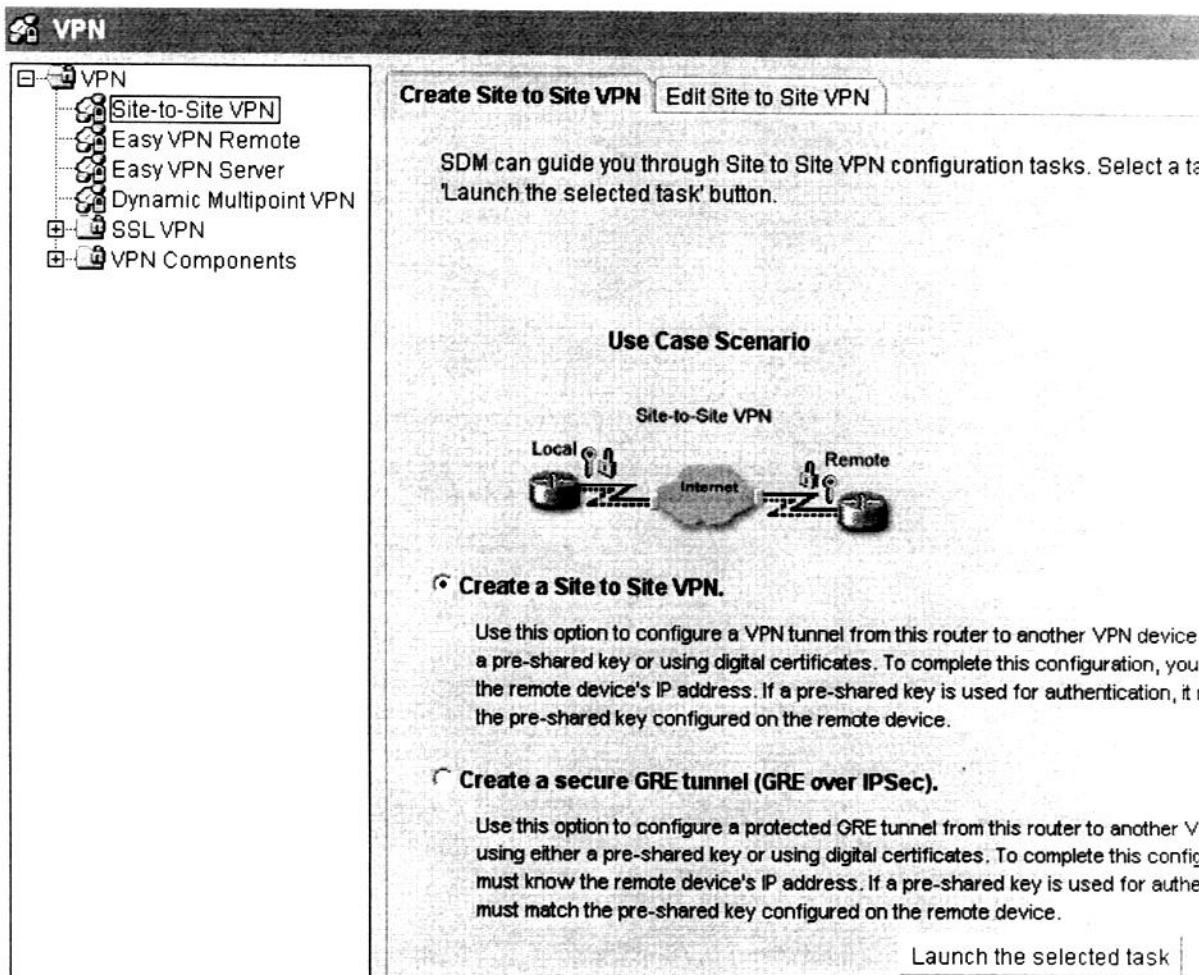
outbound pcp sas:
Router1#

```

Az ISAKMP SA-nak a QM_IDLE státusz jelenti azt, hogy a Phase 1 rendben van.

IPsec konfigurálása SDM-en keresztül

A Configure alatt a VPN-en belül lehet új VPN kapcsolatot létrehozni, vagy a meglevőket szerkeszteni. Új VPN kapcsolat létrehozása az „VPN->Site-to-Site VPN”-en belül a „Create a Site to Site VPN” kiválasztásával majd a „Launch the selected task” gomb megnyomásával az alábbi módon történik:



Következő lépésben kétféle konfigurációs mód, a **Quick Setup** és a **Step by Step wizard** közül választhatunk.

A **Quick Setup** esetén alapértelmezett értékekkel, csupán a felhasználófüggő adatok megadásával lehet konfigurálni a VPN kapcsolatot, még a Step by Step wizard egy részletesebb konfigurációs opciókkal is magunk adhatunk meg.

„Quick Setup” az alábbi képeken látható, ahol a következő paramétereket kell megadni:

- kimenő interfész kiválasztása (Select the interface for this VPN connection)
- távoli peer IP címe a Peer Identity-nél
- authentikáció konfigurálása (Pre-shared és a használandó jelszó)
- Traffic to encrypt-nél a forgalom definiálása, amit enryptálni kell (forrásnál az interfészt kéri)

VPN Connection Information

Select the interface for this VPN connection: Details...

Peer Identity

Select the type of peer(s) used for this VPN connection:

Enter the IP address of the remote peer:

Authentication

Authentication ensures that each end of the VPN connection uses the same secret key.

Pre-shared Keys pre-shared key: Digital Certificates
 Re-enter Key:

Traffic to encrypt

The traffic between the source and the destination specified here will be protected by the transforms (encryption algorithms) defined in the default transform set.

Source	Destination
Select a source interface where traffic to be encrypted originates: <input type="button" value="FastEthernet2/0"/> Details...	Enter the IP address and subnet mask of the destination where encrypted traffic terminates: IP Address: <input type="text" value="192.168.12.0"/> Subnet Mask: <input type="text" value="255.255.255.0"/> or <input type="text" value="24"/>

[< Back](#) [Next >](#) [Finish](#) [Cancel](#) [Help](#)

Az ezt követő képernyön egy összefoglalót kapunk a konfigurációról, majd a „Finish” gombra kattintva az a routerre kerül.

Az „Edit Site to Site VPN” fülön, alul a „Generate mirror...” gombbal lehet elkészíteni a másik routerhez tartozó konfigurációt.

Create Site to Site VPN **Edit Site to Site VPN**

Add... Delete

	Status	Interface	Description	IPSec Policy	S
<input checked="" type="checkbox"/>	Down	FastEthernet1/0	Tunnel to 172.16.1.2	SDM_CMAP_1	1

[Test Tunnel...](#) [Generate Mirror...](#)

A generált konfiguráció csak irányadó, az interfészek és az elnevezések mások is lehetnek, érdemes ellenörizni, hogy nem létezik-e már olyan ACL.

Step by Step wizard esetén a konfigurációs dialógusok az alábbiak:**VPN Connection Information**

Select the interface for this VPN connection:

FastEthernet1/0

Details...

Peer Identity

Select the type of peer(s) used for this VPN connection:

Peer with static IP address

Enter the IP address of the remote peer:

172.16.1.2

Authentication

Authentication ensures that each end of the VPN connection uses the same secret key.

 Pre-shared Keys Digital Certificatespre-shared key: Re-enter Key:

< Back | Next > | Finish | Cancel | Help

„Quick Setup”-tól eltérően, a következő ablakon lehetőségünk van más az alapértelmezett értékektől eltérő IKE paramétereket is megadni:

IKE Proposals

IKE proposals specify the encryption algorithm, authentication algorithm and key exchange method that is used by this router when negotiating a VPN connection with the remote device. For the VPN connection to be established with the remote device, the remote device should be configured with at least one of the policies listed below.

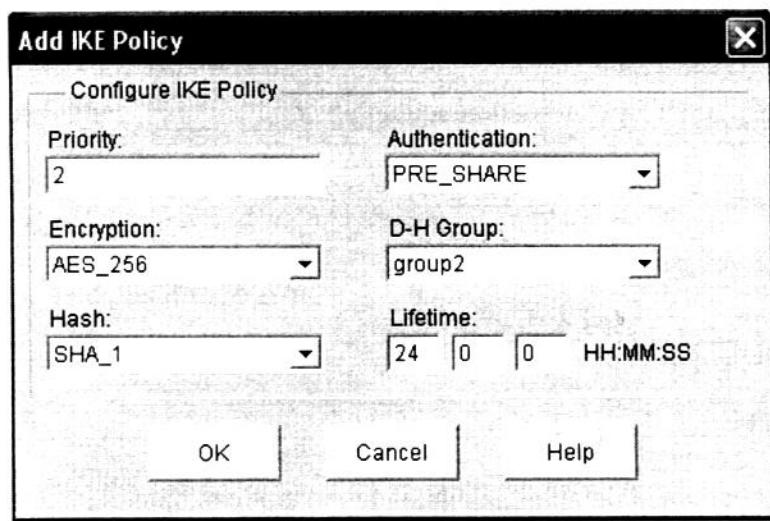
Click the Add... button to add more policies and the Edit... button to edit an existing policy.

	Priority	Encryption	Hash	D-H Group	Authentication	Type
1	3DES	SHA_1	group2	PRE_SHARE	SDM Default	

Add...

Edit

Az „**Add...**” vagy „**Edit...**” gomb megnyomásával az alábbi dialógus alapján lehet újabb paramétereket hozzáadni, vagy a meglévőket módosítani:



Priority értéke a preferenciáját jelzi a policy-nek. A legalacsonyabb prioritás értékkel rendelkező policy lesz a preferáltabb. Az *Authentication* alatt választhatjuk ki hogy *pre-shared* vagy *RSA* authentikáció legyen. *Encryption* és *Hash* alatt a használandó encryptálás és hash algoritmus, a *D-H Group* alatt pedig a kulccseréhez használt Diffie-Hellman group adható meg. A *Lifetime* értéke a létrejött SA maximálisan engedélyezett élettartama.

Következő lépésben az IPsec paramétereit lehet változtatni, vagy újat hozzátenni:

Select Transform Set:

SDM Default Transform Set

	Name	ESP Encryption	ESP Integrity	AH Integrity
ESP-3DES-SHA	ESP_3DES	ESP_SHA_HMAC		

Add...

Az „**Add...**” illetve az „**Edit...**” gombra kattintva lehet újabb IPsec proposalt felvenni, vagy a meglevőt szerkeszteni. Az alap képernyön és a „*Show Advanced*” gombra kattintva további beállítások adhatóak meg, melyek az alábbiak:

- ESP vagy AH használata
- integritás algoritmusa
- enryptálás algoritmusa (ha van)
- IPsec tunnel vagy transport mód

Add Transform Set

Name:

Data integrity with encryption (ESP)

Integrity Algorithm:

Encryption Algorithm:

Data and address integrity without encryption (AH)

Integrity Algorithm:

Mode

Tunnel (Encrypt data and IP header)

Transport (Encrypt data only)

IP Compression (COMP-LZS)

Ezt követően az IPsec tunnellen átküldendő adatokat lehet meghatározni, a forrás és cél subnet megadásával, vagy egy ACL létrehozásával/kiválasztásával.

Traffic to protect

IPSec rules define the traffic, such as file transfers (FTP) and e-mail (SMTP) that will be protected by this VPN connection. Other data traffic will be sent unprotected to the remote device. You can protect all traffic between a particular source and destination subnet, or specify an IPSec rule that defines the traffic types to be protected.

Protect all traffic between the following subnets

Local Network

Enter the IP address and subnet mask of the network where IPSec traffic originates.

IP Address:

Subnet Mask:

or

Remote Network

Enter the IP Address and Subnet Mask of the destination Network.

IP Address:

Subnet Mask:

or

Create>Select an access-list for IPSec traffic

< Back

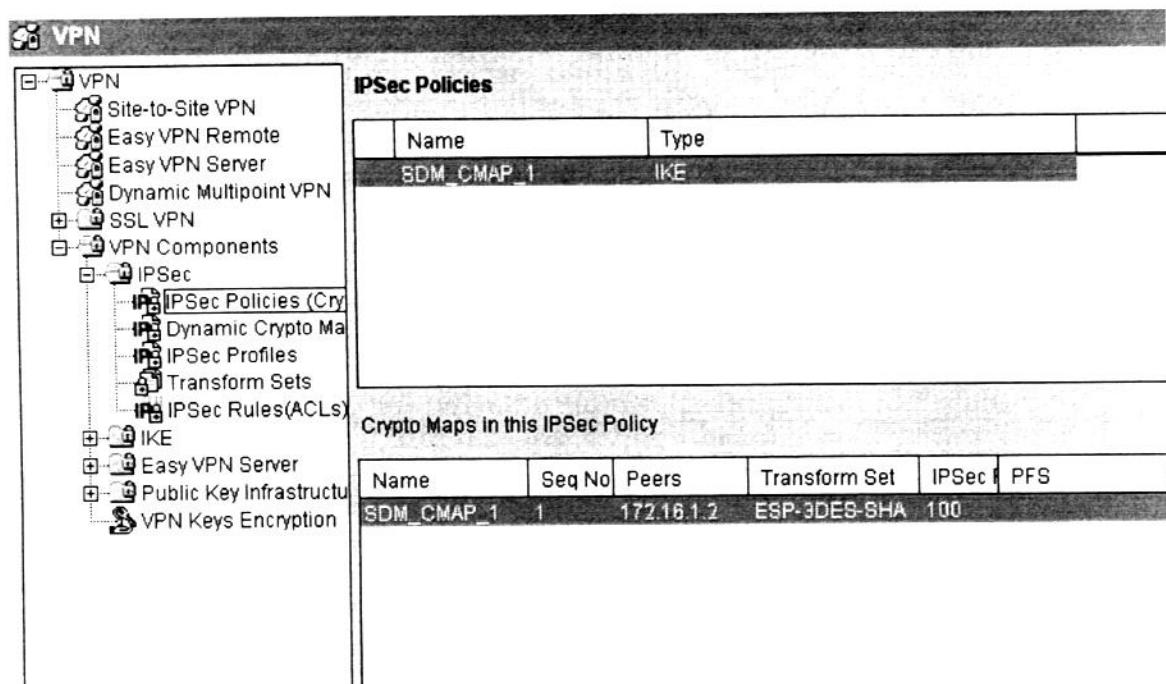
Ezt követően a konfiguráció alkalmazása előtt egy összefoglaló táblázat jelenik meg a konfigurációról.

Meglevő IPsec konfiguráció beállítása és szerkesztése a „**Configure->VPN->Site to Site VPN->Edit Site to Site VPN**” alatt lehetséges.

Az „**IPSec Rule**” a forgalmat kiválasztó ACL neve vagy száma.

		Status	Interface	Description	IPSec Policy	Seq N	Peers	Transform Set	IPSec Rule	Type
Up	FastEthernet1/0	ciscoworld	SDM_CMAP_1	1	172.16.1.2	ESP-3DES-SHA	100			Static

További IPsec-el kapcsolatos konfiguráció még a „**Configure->VPN->VPN components->IPSec Policies**”, „**Configure->VPN->VPN components->Transform Sets**”, és „**Configure->VPN->VPN components->IPSec Rules(ACLs)**” alatt található.



	Name	ESP Encryption	ESP Integrity	AH Integrity	IP Compr	Mode
	ESP-3DES-SHA	ESP_3DES	ESP_SHA_HMAC			TUNNEL

-to-Site VPN
y VPN Remote
y VPN Server
amic Multipoint VPN
VPN
Components
IPSec
IPSec Policies (Cry
Dynamic Crypto Ma
IPSec Profiles
Transform Sets
IPSec Rules(ACLs)
E
asy VPN Server
ublic Key Infrastructu
PN Keys Encryption

IPSec Rules

	Name/Number	Used by	Type	Description
◀	100	crypto map SDM_CMAP_1 1	Extended	

Action	Source	Destination	Service	Log	At
<input checked="" type="checkbox"/> Permit	192.168.11.0/0.0.0.255	192.168.12.0/0.0.0.255	ip		

IPsec monitorozása

A „Monitor”-on belül lehet az IPsec tunnellen átküldött forgalmat és a counterek értékét nyomon követni:

Each row represents one IPsec Tunnel

Stop Monitoring | Test Tunnel... | Update

Local IP	Remote IP	Peer	Tunnel Status
172.16.1.1	172.16.1.2	172.16.1.2.500	Up

Select Item to Monitor

Encapsulation Packets
 Decapsulation Packets
 Send Error Packets
 Received Error Packets

Tunnel Status

View Interval: Real-time data every 10 sec

Encapsulation Packets Decapsulation Packets Send Error Packets Received Error Packets

Encapsulation Packets

Decapsulation Packets

Send Error Packets

Received Error Packets

Erre legtöbb esetben külön alkalmazás használata javasolt, amely SNMP-n, vagy NetFlow-on keresztül kapja az adatokat.