

Paroolide murdmise vastavate tabelitega (rainbow table).
„Sool“ paroolide krüpteerimisel. Protokoll IPSec,
võtmevahetus IKE protokolliga.

Erika Matsak, PhD

Brute-force rüünded

- Kontrollitakse kõikvõimalikud paroolid kuni sobiv parool on leitud.
- Lihtne murda lühikest parooli. Parooli pikendamisega kasvab ülesande keerukus eksponentsiaalselt

Parooli/võtme pikkus bittides	Permutatsioonide arv	Brute-force aeg murdmiseks (2^{56} permutatsiooni sekundis)
8	2^8	0 millisekundit
40	2^{40}	0.015 millisekundit
56	2^{56}	1 sekund
64	2^{64}	4 minutit ja 16 sekundit
128	2^{128}	149,745,258,842,898 aastat
256	2^{256}	50,955,671,114,250,072,156,962,268,275,658,377,807,020,642,877,435,085 aastat

Brute Force Attack

Trying aaaa : failed

Trying aaab : failed

Trying aaac : failed

...

Trying acdb : failed

Trying acdc : success!

Vikerkaare tabelid (Rainbow tables)

- Moodustatakse tabelleid, mis sisaldavad kõiki võimalikke paroole, mida on võimalik genereerida etteantud sümbolitest
- Tabelis on võimalik arvestada paroolide pikkusega, suurte ja väikeste tähtedega, numbritega ja erisümbolitega (character-set)
- Tabelite genereerimiseks kasutatakse vastavat tarkvara, protseduur on palju aega nõudev, tabelite suurus on GB-des.
- Igast paroolist genereeritakse hash, mida töödeldakse redutseerimisfunktsiooniga
- Redutseerimisfunktsioon taastab parooli kasutades hash-koodi

Vikerkaare tabelid (Rainbow tables).

Näited tähestiku valikutest

- numeric = [0123456789]
- alpha = [ABCDEFGHIJKLMNOPQRSTUVWXYZ]
- alpha-numeric = [ABCDEFGHIJKLMNOPQRSTUVWXYZ0123456789]
- loweralpha = [abcdefghijklmnopqrstuvwxyz]
- loweralpha-numeric = [abcdefghijklmnopqrstuvwxyz0123456789]
- mixalpha = [abcdefghijklmnopqrstuvwxyzABCDEFGHIJKLMNOPQRSTUVWXYZ]
- mixalpha-numeric = [abcdefghijklmnopqrstuvwxyzABCDEFGHIJKLMNOPQRSTUVWXYZ0123456789]
- ascii-32-95 = [!"#\$%&'()*+,-./0123456789:;<=>?@ABCDEFGHIJKLMNOPQRSTUVWXYZ[\]^_`abcdefghijklmnopqrstuvwxyz{|}~]
- ascii-32-65-123-4 = [!"#\$%&'()*+,-./0123456789:;<=>?@ABCDEFGHIJKLMNOPQRSTUVWXYZ[\]^_`{|}~]
- alpha-numeric-symbol32-space = [ABCDEFGHIJKLMNOPQRSTUVWXYZ0123456789!@#\$%^&*()-_+=~`[]{}|\:;'"<>.,?/]
- oracle-alpha-numeric-symbol3 = [ABCDEFGHIJKLMNOPQRSTUVWXYZ0123456789#\$_]

Vikerkaare tabelid (Rainbow tables).

- Kui tegu on parooliga, mille pikkus on kuni 8 sümbolit, mis koosneb tähtedest, numbritest ja erisümbolitest, siis tõenäosus, et õnnestub hash-koodi murda on 75.42%. Vastavate vikerkaare tabelite suurus on 596GB. Tabelite genereerimine Pentium 3 protsessoriga arvutis võtab aega 3 aastat. Hash-koodi murdmine vastavate tabelitega mitte rohkem kui 22min.
- Tabelite genereerimiseks saab kasutada korraga mitmeid arvuteid, sel juhul saab tabelid valmis kiiremini. Näiteks, kui genereeritakse saja arvutiga, siis on tabelid valmis 11 ööpäevaga.

Vikerkaare tabelid (Rainbow tables).

- Iga hash-funktsiooni tüübi jaoks luuakse eraldi tabelid. Kui tegu on md5, siis tabelid, peavad olema genereeritud just md5 jaoks, kui sha1-ga, siis just sha1 jaoks jne.

```
C:\Users\Erika\vanad failid\kriptograafia\rainbowcrack-1.5-win32>rtgen md5 mixal
pha-numeric 1 12 2 3800 33554432 0
rainbow table md5_mixalpha-numeric#1-12_2_3800x33554432_0.rt parameters
hash algorithm:      md5
hash length:        16
charset:             abcdefghijklmnopqrstuvwxyzABCDEFGHIJKLMNOPQRSTUVWXYZ0123
456789
charset in hex:      61 62 63 64 65 66 67 68 69 6a 6b 6c 6d 6e 6f 70 71 72 73
74 75 76 77 78 79 7a 41 42 43 44 45 46 47 48 49 4a 4b 4c 4d 4e 4f 50 51 52 53 5
4 55 56 57 58 59 5a 30 31 32 33 34 35 36 37 38 39
charset length:      62
plaintext length range: 1 - 12
reduce offset:       0x00020000
plaintext total:     14082680407012460778

sequential starting point begin from 0 (0x0000000000000000)
generating...
```

Vikerkaare tabelid (Rainbow tables).

- Olgu, et meil on olemas hash-funktsioon H ning lõplik hulk paroole P .
- Eesmärgiks on seostada $H(p)=h$, et oleks võimalik edaspidi h järgi tuvastada p , või vähemalt tuvastada, et sellist p pole meie tabelis olemas.
- Kõikide paroolide ja hash-koodide sõnastiku moodustamise korral on tulemuslik tabel väga mahukas.
- Hash-ahelad (hash-chains) on spetsiifiline tehnika $H(p)=h$ seostamiseks. Teostatud reduktsioonifunktsioonide abil. Tabelis hoitakse esimest ja viimast parooli ahelast

aaaaaa \xrightarrow{H} 281DAF40 \xrightarrow{R} sgfnvd \xrightarrow{H} 920ECF10 \xrightarrow{R} kiebgt

Sool paroolides

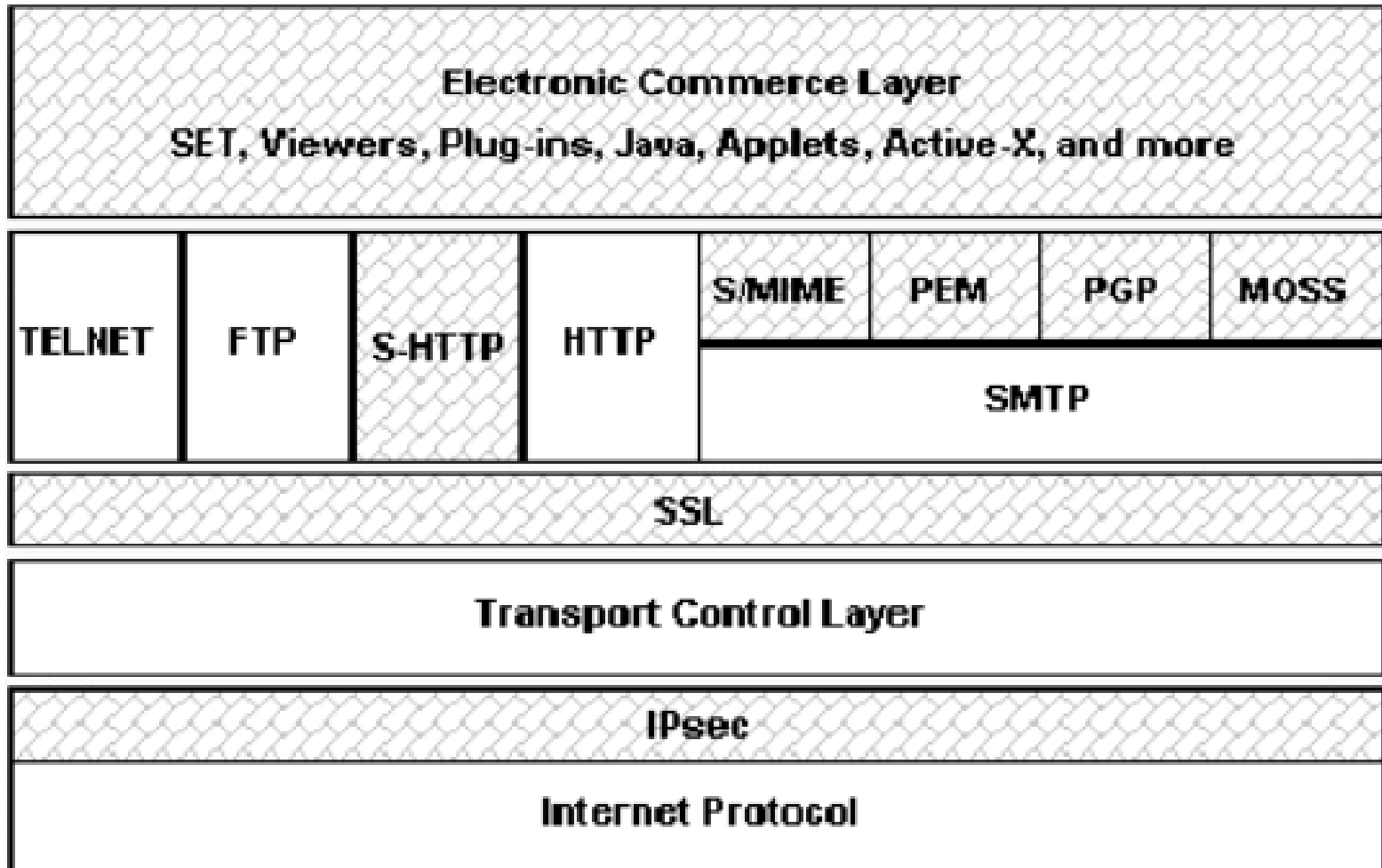
- Krüptograafias on sool – juhuslike sümbolite jada, mida sisestatakse hash-funktsioonidesse koos parooliga
- Sool kaitseb vikerkaare tabelitega seotud murdmiste eest
- Sool on 8-12 sümbolite pikk. Vikerkaare tabelid ei ole mõeldud nii pikkade paroolide murdmiseks
- Op. süsteemide näited, mis kasutavad soola:
 - Linux (koos MD5, SHA-256 või SHA-512)
 - Wordpress (koos MD5)
 - Joomla alates versioonist 1.0.13 (koos MD5)

Sool peab olema genereeritud kasutades: Cryptographically Secure Pseudo-Random Number Generator (CSPRNG)

<https://crackstation.net/source/password-hashing/PasswordHash.java>

<https://crackstation.net/source/password-hashing/PasswordHash.php>

Protokollid ja nende hierarhia



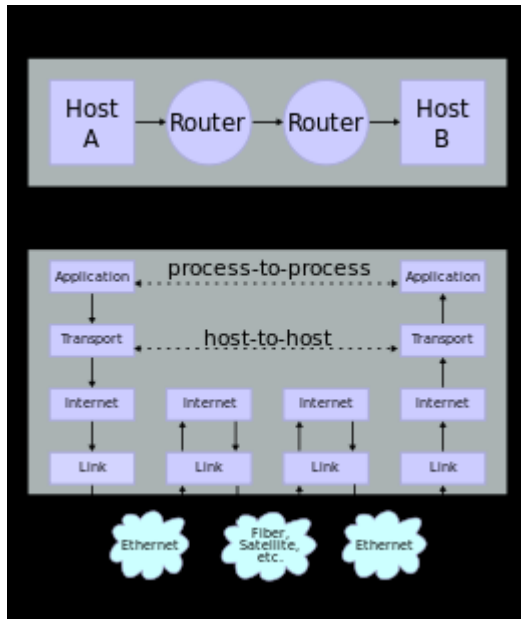
Mõned tähtsad protokollid

- Electronic commerce layer, PayPal, Ecash, 3D secure
- S-http, PGP, PGM, S/MIME
- Transport Layer security (ssh, ssl, tls)
- Transmission Control Protocol (TCP), User Datagramm Protocol (UDP)
- IPsec (Internet Security Protocol)

TCP/IP

- TCP-IP protokoll ei võimalda turvalisust. Edastatavad sõnumid on kõigile nähtavad, krüpteerimist ei kasutata
- Ei ole võimalik veenduda, et saatja on just see kelleks ta end nimetab
- Ei ole võimalik veenduda, et saadetud info on tulnud kohale just sellisel kujul nagu see oli saadetud
- Ei ole võimalik kaitsta andmeid pealtkuulamise eest

TCP/IP ja muud kihid



Rakenduskiht (application layer). Sisaldab kasutajaga seotud toiminguid. Üle võrgu suhtlemiseks: POP, IMAP, SMTP, FTP, HTTP

Transpordikiht (transport layer). Juhib programmide omavahelist suhtlemist võrgus: UDP, TCP

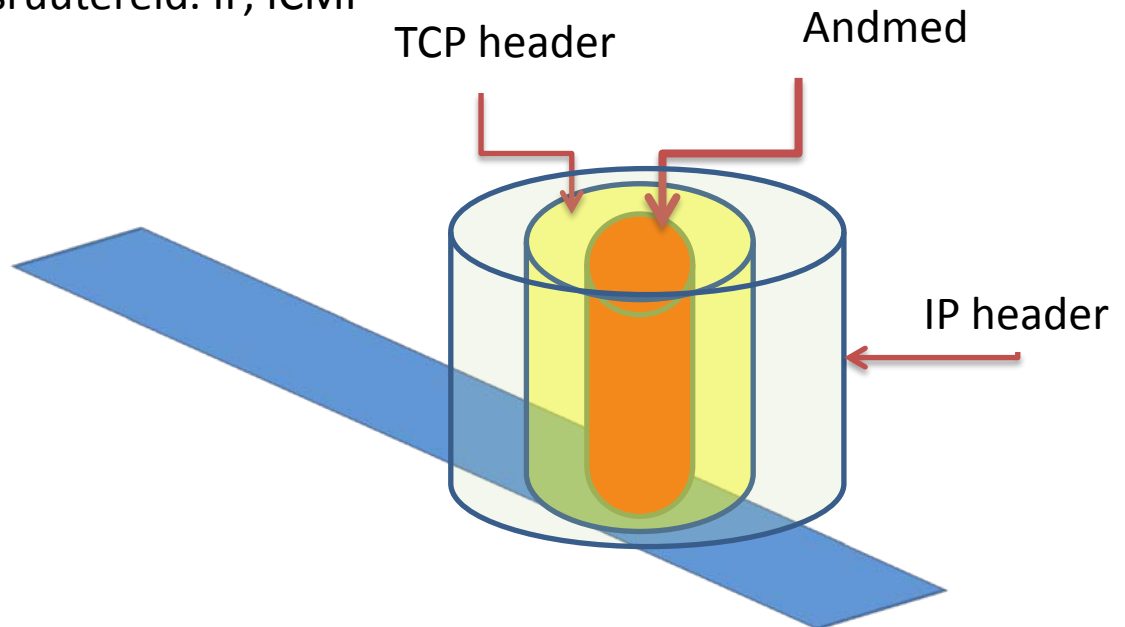
Võrgukiht (network layer)- võimaldab andmeedastust masinate vahel, mis asuvad erinevates alamvõrkudes. Kasutatakse maršruutereid. IP, ICMP

Võrgupööruskiht (link layer)- toimub füüsiline adresseerimine ja füüsiliste parameetrite määramine.

Füüsiline kiht (physical layer)- füüsiline andmeedastus

IP päis vastutab selle eest, et oleks teada kuhu on andmed edastatud võrgus. Näiteks punktist A punkti B

TCP päis – kuhu peaksid olema andmed edastatud süsteemis B

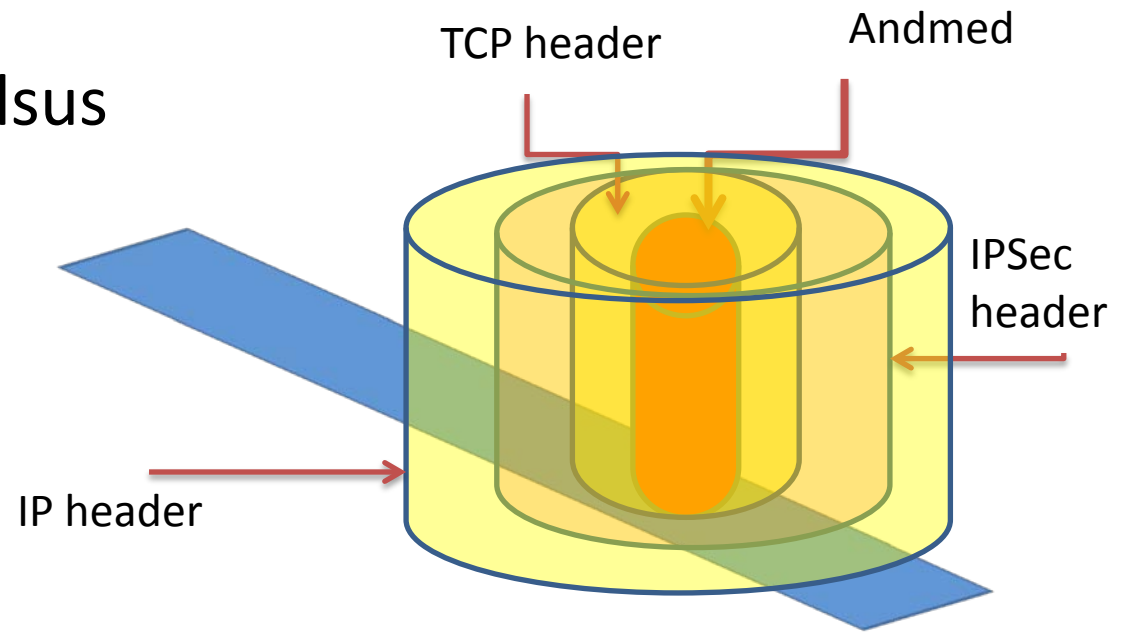


Protokoll IPSec

- Selleks, et alustada turvalist ühendust IPSec protokolliga, peavad osapooled vahetama võtmeid, näiteks IKE (Internet Key Exchange) protokolliga
- IPSec on kolmanda kihi (network -võrgukiht) protokoll ning on kasutatav TCP ja UDP protokollidega
- IPSec protokollis on võimalik kasutada kahte režiimi: transpordikihi ja tunneli tasemel
- IPSec protokoll vajab kaht andmebaasi:
 - Security association database: Turvaindeksid. Sisaldab informatsiooni, mis on vajalik ühe paketi kapseldamiseks: krüptoalgoritme, võtmete suurust, paketi numbreid jne
 - Security policy database: Turvapoliitikad. Andmed selle kohta millal tohib ja millal mitte võtta vastu mittekaitstud pakette

Protokoll IPSec server 2003 näitel

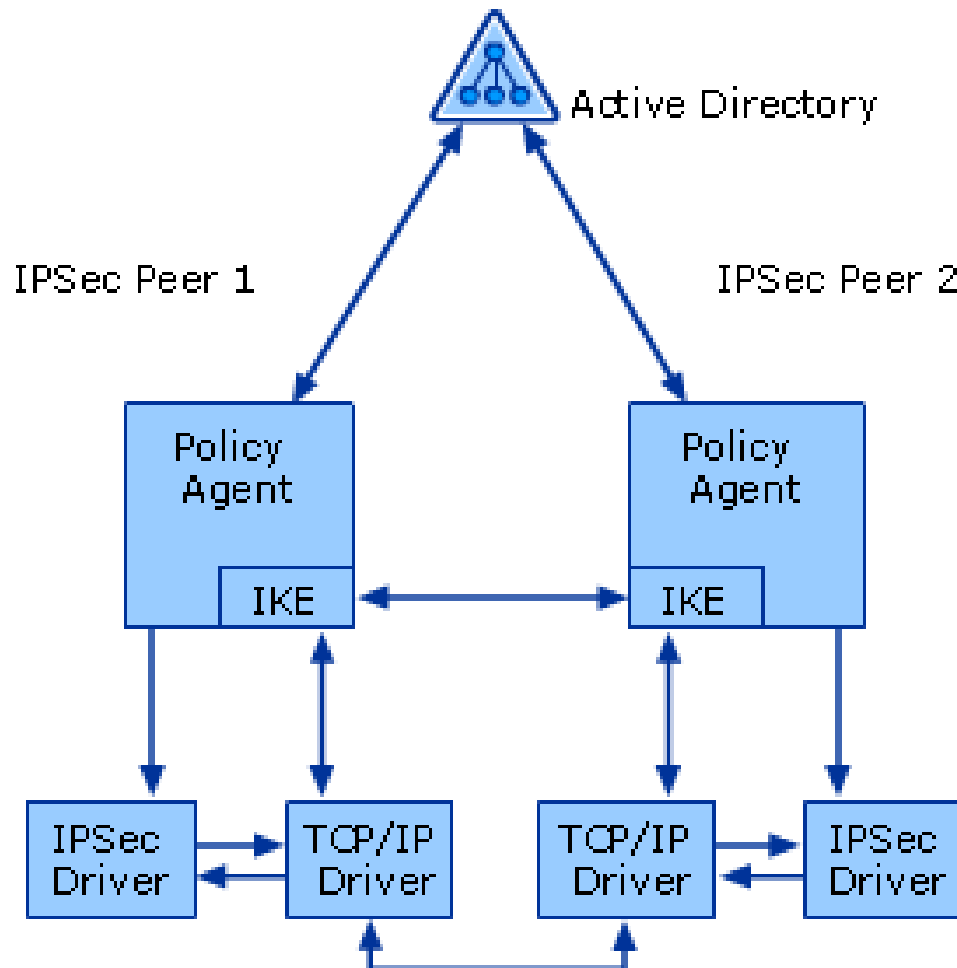
- Tagab järgmisi asju:
 - Autentimine
 - Terviklus
 - Konfidentsiaalsus



Kommunikatsiooni võimalus:

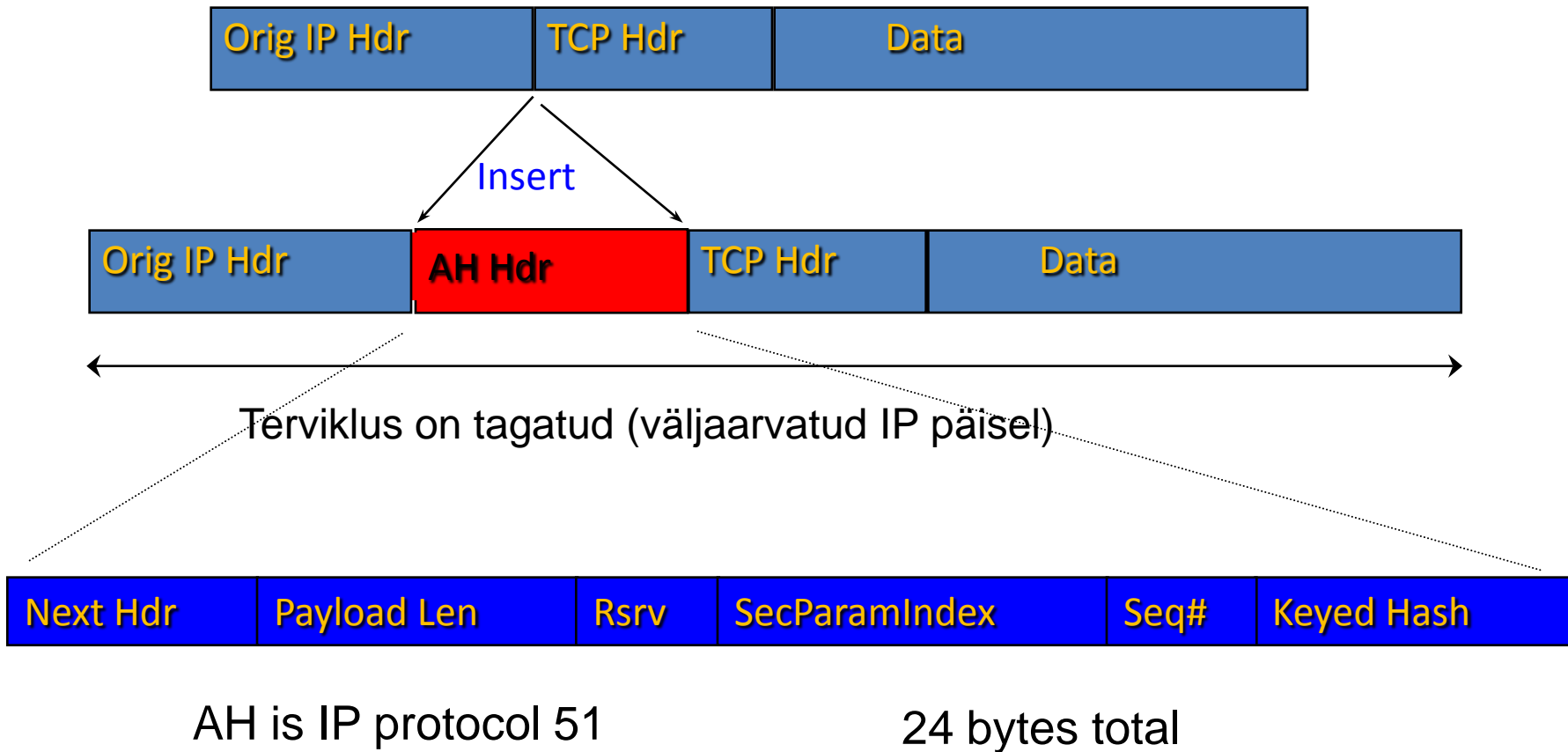
- Klient-Server
- Server-Server
- Üks võrk- teine võrk (network-network)

Protokoll IPSec server 2003 näitel



IPSec Autentimise päis: Authentication Header (AH) transpordikihi tasemel

© 2000 Microsoft Corporation



Klient-Server autentimine

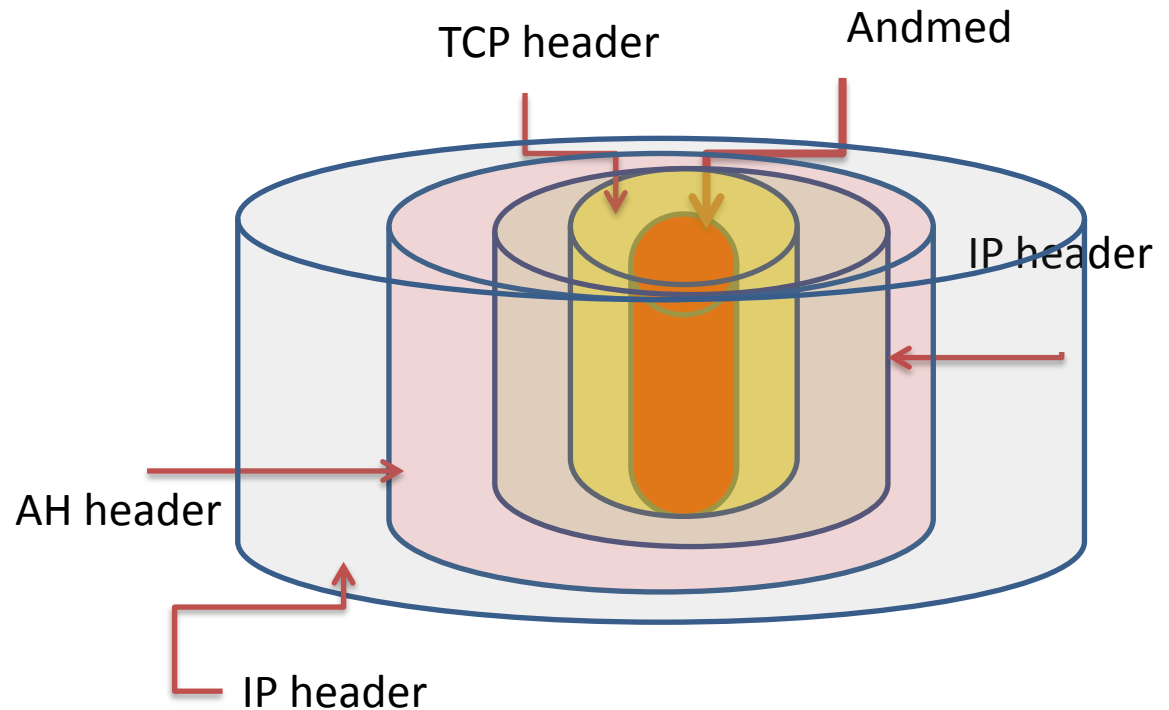
© 2000 Microsoft Corporation

IPSec Autentimise päis: Authentication Header (AH) transpordikihi tasemel

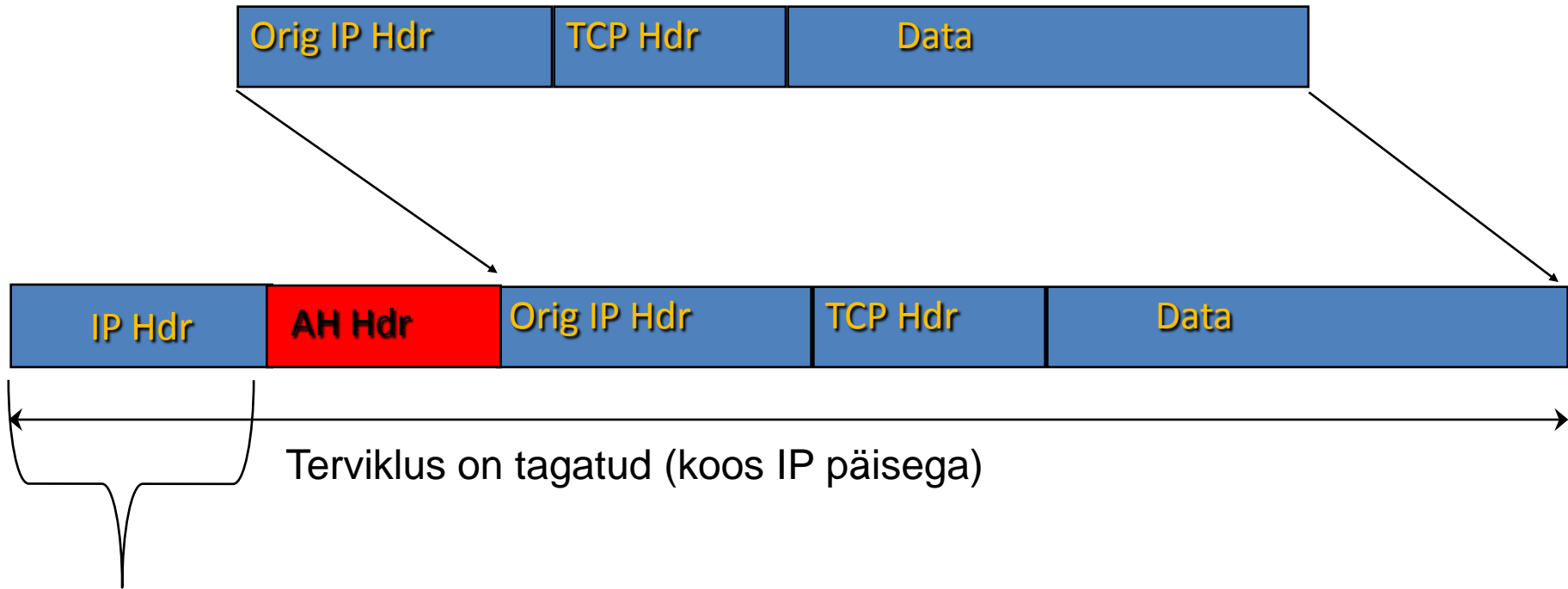
- AH eesmärk. Kaitse rünnete eest, mis on suunatud tervikluse vastu, sh võrguaadressi võltsimisele.
- Next Hdr (8 bitti) – protokoll nimetus, mis järgneb AH-le
- Payload Len (8 bitti) – AH suurus 32 bitistes sõnades miinus 2, peab olema jagatav 8 Baidiga
- Rsrv (8 bitti) - reserveeritud, täidetakse nullidega
- SecParamIndex (32 bitti) - turvalisuse indeks SPI1...255
<http://www.omnisecc.com/security/ipsec/ipsec-security-parameter-index-spi.php>
- Seq# (32 bitti) – paketi järjekorra number, kaitseb selle eest, et autentimisega seotud parameetreid ei saaks kasutada mitmekordselt
- Keyed Hash-kontrollsumma arvutamine
- AH järgi on võimalik kasutada ESP (Encapsulating Security Payload), mis on sarnane AH-ga, aga võimaldab sisu edastamist krüpteeritud kujul

Protokoll IPSec pakett tunneli jaoks server 2003 näitel

- IP päist on kasutatud kaks korda



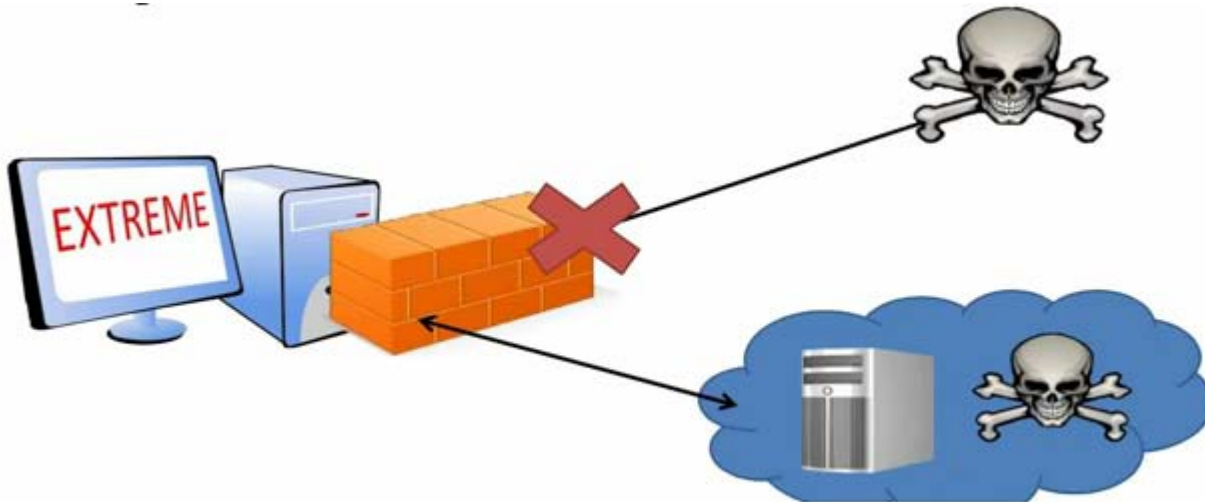
IPSec Autentimise päis: Authentication Header (AH) tunneli tasemel



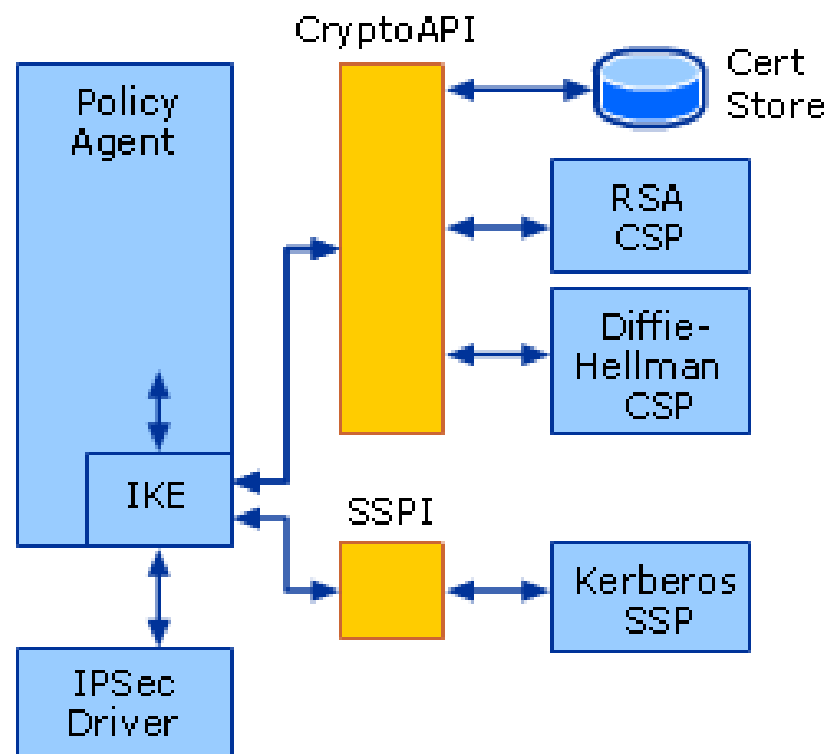
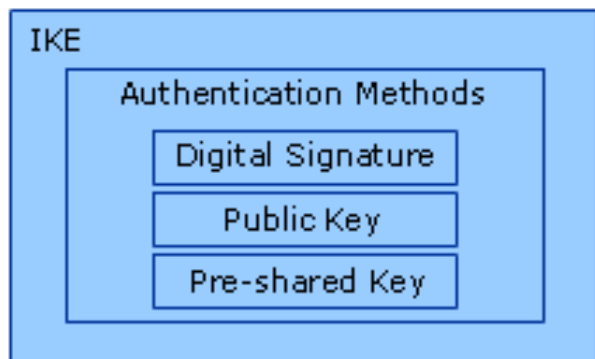
Uus IP päis koos IP aadressiga

Windows server 2008 edasiarendused

- Integreeritud windows tulemüüri
- Toetab IPv6
- Integreeritud Nap-iga (Network Access Protection), mis kontrollib, kas võrk on turvaline
- Rohkem krüpteerimisalgoritme



Paar sõna IKE kohta



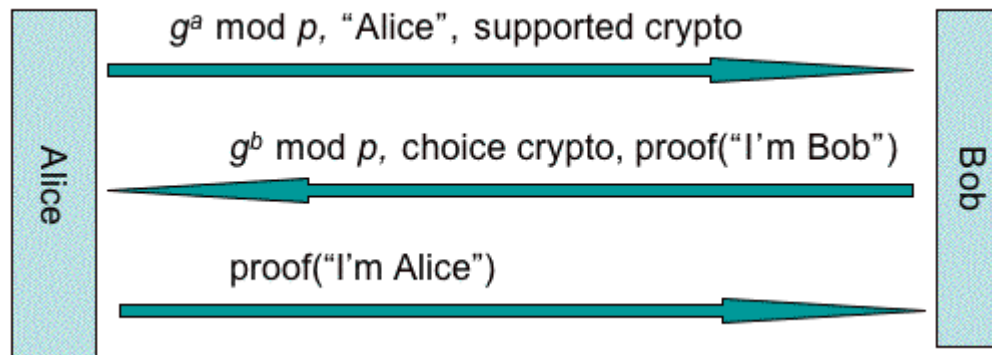
IKE

- Edastab SA (Security association) parameetreid, moodustab ja kustutab SA
- Kooskõlastab protokollide kasutust
- Genereerib võtmeid
- Koosneb järgmistest etappidest:
 - Faas 1. SA moodustamine IKE jaoks, kaks režiimi: tavaline ja agressiivne
 - Faas 2. SA moodustamine IPSec jaoks. Töötab ainult kiirrežiimis
 - IKE SA hooldus (Maintenance)
 - Kokkulepped millist Diffie-Hellman gruppi kasutada

IKE

- Esimese sammuna lepitakse kokku, millist autentimismeetodit hakatakse kasutama:
 - Eelvahetatud võtmed (preshared keys)
 - Kerberos
 - Digiallkiri koos DSA-ga
 - Digiallkiri koos RSA-ga
 - Autentimine, mis kasutab krüpteeritud juhuarve (*nonce*).

IKE. Agressiivne režiim



In aggressive mode, Alice chooses some Elgamal context (p, g) . Bob may not support it, and reject the connection. If that happens, Alice should try and connect to Bob using main mode.

Aggressive mode provides mutual authentication, and a shared secret $g^{ab} \bmod p$, which can be used to derive keys for the symmetric crypto protocols.

Diffie-Hellman'i rühmad

$$A = g^a \pmod{p}$$

1. MODP rühm, mis kasutab 768-bitist moodulit
2. MODP rühm, mis kasutab 1024-bitist moodulit
3. ECP rühm, mis kasutab 155-bitist moodulit
4. EC2N rühm, mis kasutab 185-bitist moodulit
5. MODP rühm, mis kasutab 1680-bitist moodulit

Rühmad, mis on indekseeritud EC2N kasutavad elliptiliste kõverate algoritmi

Tavaline režiim, autentimiseks Kerberos

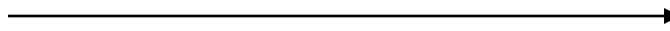


Initiator

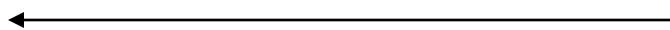


Responder

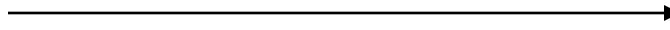
Header, SA Proposals



Header, Selected SA Proposal



Header, D-H Key Exchange, Nonce_i,
Kerberos Token_i



Header, D-H Key Exchange, Nonce_r,
Kerberos Token_r

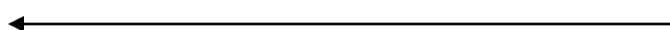


Encrypted

Header, Id_i, Hash_i



Header, Id_r, Hash_r



Tavaline režiim, autentimiseks sertifikaat

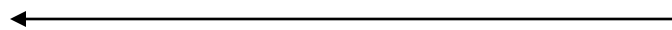


Initiator



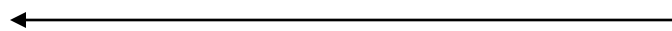
Responder

Header, SA Proposals →



← Header, Selected SA Proposal

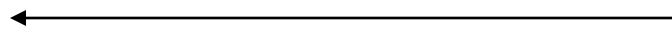
Header, D-H Key Exchange, Nonce_i →



← Header, D-H Key Exchange,
Nonce_r, Certificate Request

Encrypted

Header, Id_i, Certificate_i, Signature_i,
Certificate Request →



← Header, Id_r, Certificate_r,
Signature_r

Tavaline režiim, autentimiseks eelvahetatud võtmed (Pre-shared Key)

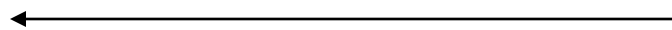


Initiator



Responder

Header, SA Proposals →



← Header, Selected SA Proposal

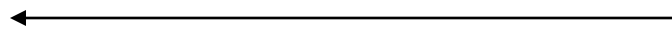
Header, D-H Key Exchange, Nonce_i →



← Header, D-H Key Exchange, Nonce_r

Encrypted

Header, Id_i, Hash_i →



← Header, Id_r, Hash_r

Kiirrežiim (Quick Mode Negotiation)



Initiator



Responder

Encrypted

Header, IPSec Proposed SA →

← Header, IPSec Selected SA

Header, Hash →

← Header, Connected Notification

IKE poliitikad

- Kui on SA mehhanismid kokkulepitud, on vaja leppida kokku poliitikad
- Poliitika on antud juhul midagi sellist nagu: kõike tuleb autentida ja võimalusel krüpteerida ning kui võimalik, siis kokku pakkida
- Iga operatsiooni jaoks on võimalik valida mitu algoritmi
- Kokkulepe poliitikas töötab nii, et initsiaator pakub välja algoritme ja vastaja eemaldab nimekirjas kõike, mis talle ei sobi