SSH haladóknak

minden ami a sima jelszavas bejelentkezésen túl van, kulcsok, port forward szegény ember vpn-je

Zámbó Marcell <lilo@andrews.hu>

Andrews IT Engineering Kft.

Amit az sshról tudni érdemes...

- man ssh
- man ssh_config
- man sshd
- man sshd_config

- Köszönöm a figyelmet! :)

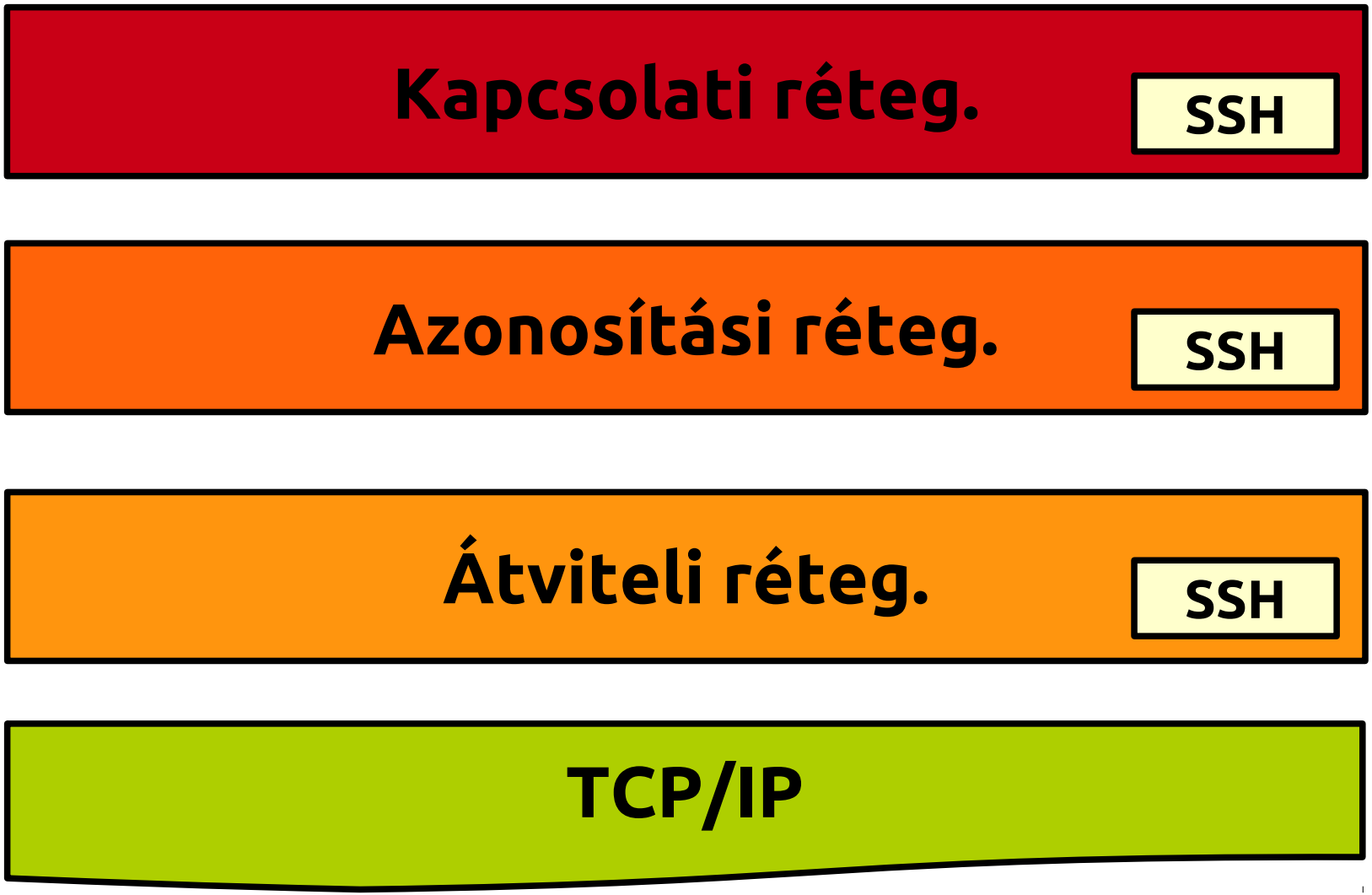
Egy kis történelem ...

- Kiket váltott le az SSH?
  - Telnet – 1969(!) RFC 15
  - Rlogin – 1983 RFC 1258
- Miért?
  - Lehallgathatóság volt a kiváltó ok.
    - 1995 – Helsinki egyetemen - Tatu Ylönen

OpenSSH bemutatása röviden

- Az OpenSSL könyvtárra épülő - titkosított, hitelesített adatátvitelt megvalósító - kliens-szerver alkalmazás csoport.
  - $ find **openssh-5**.8p1/ | egrep '\.(c|h)$' | xargs wc … **184.129** 611.431 5.008.990 összesen
  - $ find **openssl-1**.0.0e/ | egrep '\.(c|h)$' | xargs wc … **438.075** 1.564.441 13.710.386 összesen
  - find **netkit-telnet-ssl-0**.17... | egrep '\.(c|h)$' | xargs wc **11.122** 40.562 295.465 összesen
  - find gnutls26-2.10.5/ | egrep '\.(c|h)$' | xargs wc … **197.966** 647.472 5.462.680 összesen

Az SSH protokoll rétegei ...



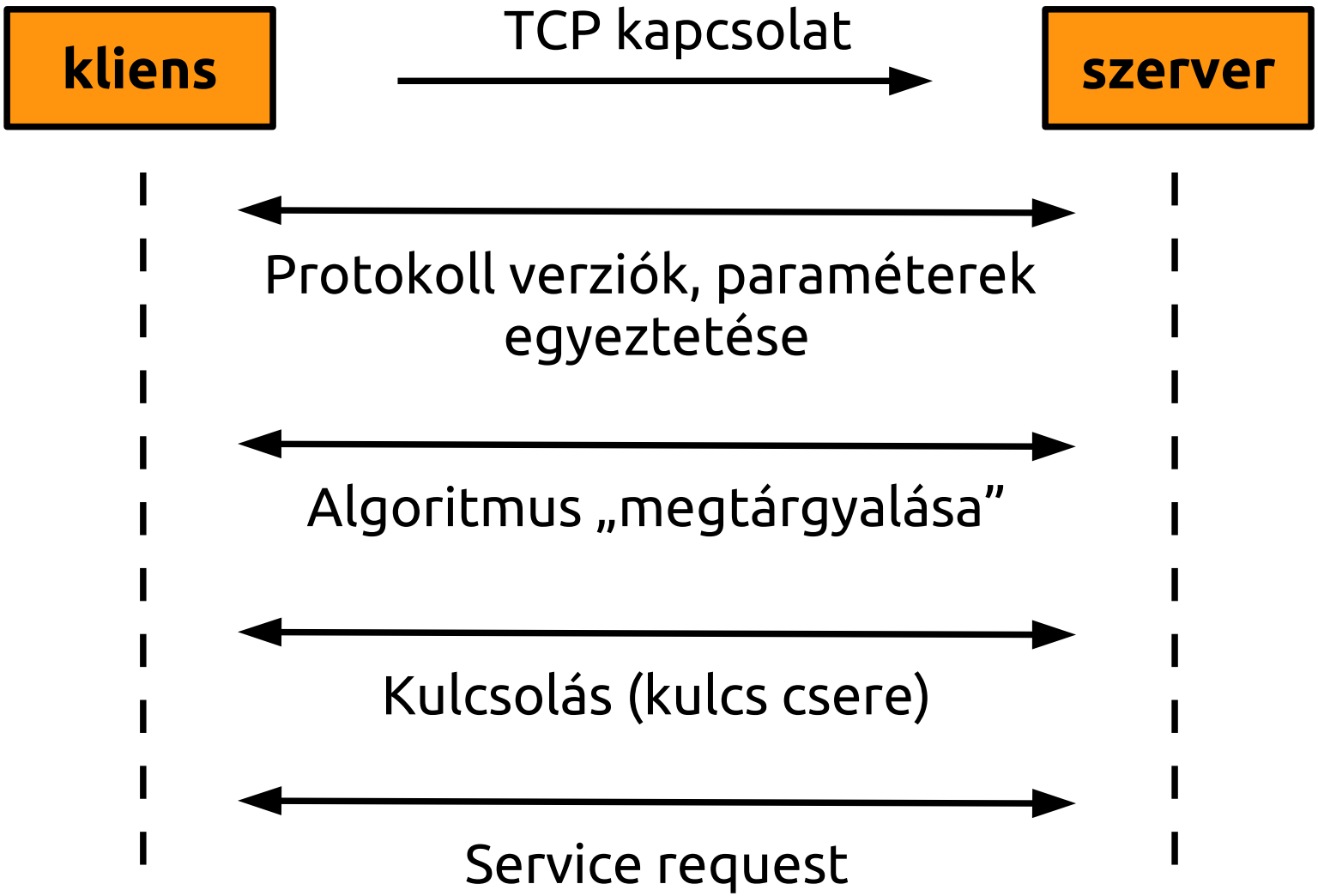
Az SSH protokoll rétegei … I.

- Kapcsolódási (connection) réteg, portforwardokért felel:
  - shell – SCP/SFTP,
  - direct-tcp klienstől a szerverhez,
  - forwarded-tcp szervertől a klienshez.
- **+1 független:** SSH ujjlenyomat ellenörző réget (SSHFP)
  - Using DNS to Securely Publish Secure Shell (SSH) Key Fingerprints
  - SSHFP resource record:
    pelda. SSHFP 2 1 123456789a.....

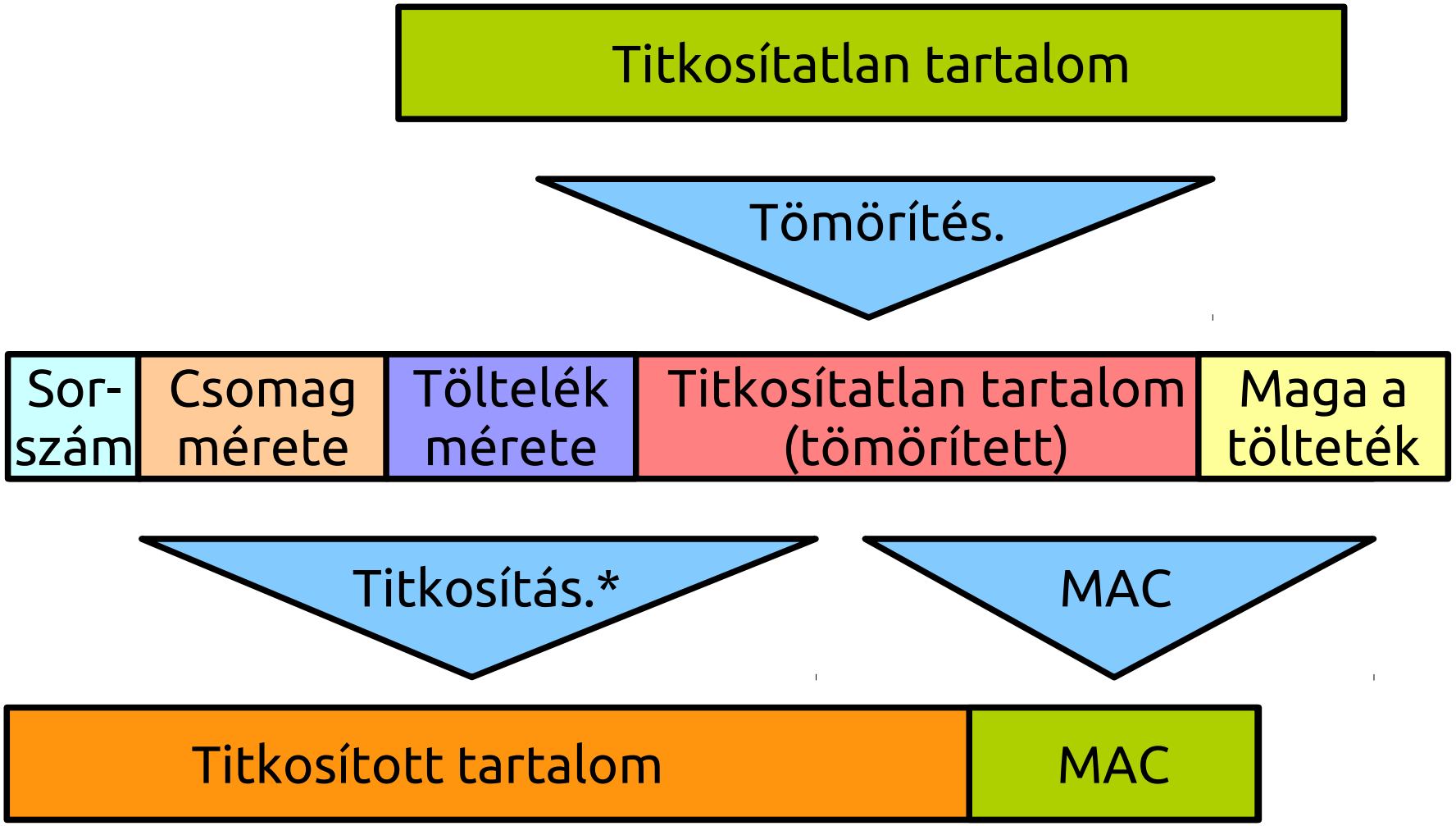
Az SSH protokoll rétegei … II.

- Felhasználói azonosítás réteg, feladata:
  - Jelszó, publikus kulcs (DSA/RSA/X.509), billentyűzet interaktivitás (CryptoCard, SecureID, OTP, S/Key) és GSSAPI (Kerberos5, NTLM) alapú felhasználó azonosítás.
    És minden egyéb amit PAM-on keresztül meg lehet valósítani
- Átviteli (transport) réteg, feladata:
  - Titkosítás (titkosítási kulcs kezelés – újra kulcsolás), integritás ellenőrzés és tömörítés.

SSH kapcsolat kiépítése ...



SSH csomagok szerkezete



* a sorszám nem kerül bele a titkosított tartalomba.

Hasznos parancssori opciók

- -N
  - Nem futtat semmilyen parancsot.
- -n
  - stdin >/dev/null, háttérben indított parancsokhoz.
- -f
  - Az ssh hattérben futtatásához szükséges.
- -C
  - Bekapcsolja a tömörítést.

Portfwd: távoli portot ide hozzám ...

- -L<ezen a helyi porton figyeljen>:<ennek a távoli az ip-nek>:<ez a portja>
- ssh -L1234:127.0.0.1:25 server
  - A saját gépem 127.0.0.1:1234 címén is elérhetem a „server” belső 25-ös portját
- -L localIP:localPort:remoteIP:remotePort
  - Megadhatom, hogy a saját gépemen, milyen ip-n figyeljen a forgalom

Portfwd: vidd el a portom messzire

- -R<a távoli gép ezen portján figyeljen>:<ennek a helyi ip-nek>:<ez a portja>
- ssh -R3128:proxy:3128 server
  - A server 3128-as portján elérhető lesz a proxy 3128-as portja
- -R localIP:localPort:remoteIP:remotePort
  - Akárcsak a -L-nél ...

Egy kis portfwd móka

- Mi történik ilyenkor? (DoS)
  - ssh server -L2223:127.0.0.1:2223 -R2223:127.0.0.1:2223 ?
- Vissza is akarunk ssh-zni …:
  - ssh server -g -R0.0.0.0:2022:0:22
    - A server-en a 0.0.0.0:2022-n elérhető lesz a kliens 22-es portja. (0.0.0.0, mi is ez?)
    - -g mit is csinál? (e nélkül csak 127.0.0.1-en figyel)
    - 0.0.0.0 helyett adhatunk más értelmes IP-t is ...

Dinamikus portfwd-ing (Socks[45])

- -D<helyi port>, pl ssh -D 4000 server
  - /etc/tsocks.conf
    server = 127.0.0.1
    server_type = 4
    server_port = 4000
  - # tsocks telnet valami 25
    - A telnet a socks-on és az ssh-n keresztül, a serverről fog kapcsolódni a valami 25-ös portjára
    - Tetszőleges ip-n, tetszőleges tcp port elérhető így …
    - Pl böngészőben beállítva hasznos tud lenni.

Lehetséges authokról ...

- Userek AD-ből, pam_winbindd:

  - /etc/nsswitch.conf:
    passwd: files winbind
    group: files winbind

  - /etc/pam.d/ssh:
    auth sufficient /lib/pam_winbind.so

- FIXME!

Escape

- Ami telnet-en a ^] volt, az itt szabadon beállítható, alapból a ~

Supported escape sequences:
~. - terminate connection (and any multiplexed sessions)
~B - send a BREAK to the remote system
~C - open a command line
~R - Request rekey (SSH protocol 2 only)
~^Z - suspend ssh
~# - list forwarded connections
~& - background ssh (when waiting for connections to terminate)
~? - this message
~~ - send the escape character by typing it twice
(Note that escapes are only recognized immediately after newline.)

SSH kliens parancssor ...

~C: / ssh> help

Commands:

-L[bind_address:]port:host:hostport
Request local forward

-R[bind_address:]port:host:hostport
Request remote forward

-D[bind_address:]port
Request dynamic forward

-KR[bind_address:]port
Cancel remote forward

Mire kell nagyon figyelni?

- Hibaüzenetek, változások

```
@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
@    WARNING: REMOTE HOST IDENTIFICATION HAS CHANGED!     @
@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
IT IS POSSIBLE THAT SOMEONE IS DOING SOMETHING NASTY!
Someone could be eavesdropping on you right now (man-in-the-middle attack)!
It is also possible that a host key has just been changed.
The fingerprint for the ECDSA key sent by the remote host is
1a:a7:5f:b7:d1:0c:63:ba:2c:71:e5:b4:af:6c:04:28.
Please contact your system administrator.
Add correct host key in /root/.ssh/known_hosts to get rid of this message.
Offending ECDSA key in /root/.ssh/known_hosts:5
  remove with: ssh-keygen -f "/root/.ssh/known_hosts" -R 0
ECDSA host key for 0 has changed and you have requested strict checking.
Host key verification failed.
```

ControlMaster, master mode connection sharing

- Master socket létrehozása
  - ssh -M -S ~/socket server -NnfC
- További kapcsolatok
  - ssh -S ~/socket server ...
- Példák, mire jó ez?
  - gyorsabb az új kapcsolat létrehozása
  - 1x authentikálunk csak (ccard, otp esetén, hm?)
  - csak 1 kapcsolat van a szerver-kliens között

Két hálózat teljes összekapcsolása

Szerveren:

```
sshd_config-ba:PermitTunnel yes
iface tun0 inet static
  address 10.254.254.1
sudo sysctl net.ipv4.conf.default.forwarding=1
sudo ifup tun0
```

Kliensen:

```
iface tun0 inet static
  address 10.254.254.2
  up route add -net 10.0.0.0 gw 10.254.254.1 tun0
sudo sysctl net.ipv4.conf.default.forwarding=1
ssh 5.6.7.8 -w 0:0 -Nnf
sudo ifup tun0
```

Hogyan védjük meg az SSHD-t? I.

- Saját konfigurációs állományából
  - ListenAddress, IPv[46]?, portok?
  - {Allow,Deny}{User,Group}s,
  - Allow{Agent,Tcp}Forwarding, X11Forwarding
  - Permit{EmptyPasswords,RootLogin} NO
  - Minden legyek kikapcsolva, amit nem használunk, ide értve a titkosító, hitelesítő algoritmusokat is! (Ciphers, MACs = AES, SHA)

Hogyan védjük meg az SSHD-t? II.

- Csomagszűrőből:
  - Fix IP-kről, akár GeoIP-vel …
  - limitekkel
- Akár újrafordítással
  - Ami nem kell, kivele (configure –with{,out} …)
- Auth hiányosságok pótlása
    - Pam modulok (pam_faildelay, pam_tally2)
    - Fail2ban jellegű megoldások

Ha marad(t volna) rá idő ...

- Szerver oldalon: sftp alrendszer
- Kliens oldalon:
  - ssh-copy-id,
  - ssh-agent,
  - ssh-argv0,
  - ssh-add
  - scp és sftp (lehet, hogy az 'rsync -e ssh' jobb? :)

Köszönöm a figyelmet! Kérdések?

- **„ Végállomás!**

  Kérjük kedves utasainkat megállás után szíveskedjenek elhagyni a vonatot és erre figyelmeztessék utastársaikat is! ”