



VZDIALENÁ SPRÁVA POČÍTAČA S OS LINUX

PRIPOJENIE NA SSH SERVER



Teoretická časť

SSH (Secure SHell) protokol slúži na vzdialenú správu systému v textovom režime bezpečným spôsobom. Existujú dve verzie tohto protokolu; v CentOS 7 už nie je podpora pre verziu 1.

SSH server sa voči klientovi autentizuje pomocou asymetrickej kryptografie, t. j. vlastníctvom dvojice šifrovacích kľúčov, verejného a privátneho. Ak sa klient prihlasuje na *SSH* server prvýkrát, zobrazí sa mu varovanie o neznámej identite *SSH* servera obsahujúce odtlačok jeho verejného kľúča. Po akceptovaní tohto kľúča by sa už pri ďalšom pripojení toto varovanie zobrazovať nemalo. Iný typ varovnej správy sa objaví v prípade, keď boli na serveri pregenerované kľúče alebo v prípade, keď sa server autentizuje iným typom kľúča ako doteraz. V CentOS 6 boli pre *SSH* verzie 2 dostupné kľúče generované algoritmami *RSA* a *DSA*, v CentOS 7 bolo *DSA* nahradené algoritmami *ECDSA* a *ED25519*.

Používatelia často nerozumejú základom nadväzovania bezpečného spojenia a ani samotnému priebehu komunikácie. Asymetrická kryptografia sa používa iba v procese autentizácie a nie na šifrovanie dátovej prevádzky; šifrovanie komunikácie prebieha niektorým zo symetrických šifrovacích algoritmov. Kedysi sa na odvodenie potrebného symetrického šifrovacieho kľúča využívala asymetrická kryptografia; bolo to tak napr. v protokole *SSH* verzie 1. V súčasnosti sa na odvodenie symetrického šifrovacieho kľúča používa najčastejšie nejaká forma algoritmu *Diffie-Hellman*.

Bezpečná komunikácia nezahŕňa len šifrovanie. Mala by dokázať zabezpečiť dôvernosť, integritu a autenticitu pôvodu dát. Na to všetko sú určené rôzne algoritmy; pri nadväzovaní SSH spojenia si klient aj server navzájom vymenia množinu algoritmov, ktoré podporujú a dohodnú sa, aké z nich použijú.

V OS Linux sa najčastejšie používa implementácia protokolu SSH s názvom *OpenSSH*. Pri inštalácii CentOS 7 z inštalačného DVD sa nainštaluje klientská aj serverová časť *OpenSSH*, SSH server je predvolene spustený a pre pripojenie k nemu je nastavená aj výnimka vo firewall. Najdôležitejšie súbory potrebné pre fungovanie protokolu SSH sa nachádzajú v */etc/ssh*. Tento priečinok obsahuje súbory:

<i>ssh_config</i>	– konfiguračný súbor SSH klienta
<i>sshd_config</i>	– konfiguračný súbor SSH servera
<i>moduli</i>	– obsahuje dáta potrebné pre vytvorenie symetrického šifrovacieho kľúča algoritmom <i>Diffie – Hellman</i>
<i>ssh_host_rsa_key</i>	– privátny RSA kľúč
<i>ssh_host_rsa_key.pub</i>	– verejný RSA kľúč
<i>ssh_host_ecdsa_key</i>	– privátny ECDSA kľúč
<i>ssh_host_ecdsa_key.pub</i>	– verejný ECDSA kľúč
<i>ssh_host_ed25519_key</i>	– privátny ED25519 kľúč
<i>ssh_host_ed25519_key.pub</i>	– verejný ED25519 kľúč



Pomôcky

Tri virtuálne stroje vytvorené vo *VMware vSphere*; dva s CentOS 7 a jeden s OS Windows (na verzii OS Windows nezáleží, pri písaní tohto materiálu bol použitý Windows 10). Stroje s CentOS 7 majú kvôli vzájomnému rozlíšeniu názvy *SERVER* a *CLIENT*. Úloha je určená pre jedného žiaka.



Úlohy

Použitie tohto učebného materiálu je určené výhradne pre Duálne vzdelávanie realizované SPŠ elektrotechnickou Košice v spolupráci s Deutsche Telekom IT Solutions Slovakia.

Autor: J. Ploščica
Verzia 3

Predmet: ZIL, 1. ročník
Strana 2 z 14

1. Spustíte všetky virtuálne stroje, do strojov *SERVER* a *CLIENT* sa prihlásite ako *root*, do počítača s Windows 10 ako používateľ s administrátorskými právami.
2. Na strojoch s CentOS 7 zistíte, aké IP adresy dostali tieto počítače prostredníctvom *DHCP*, overte, že súhlasia s IP adresami, ktoré máte na obidvoch strojoch zapísané v súbore */etc/hosts*.
3. Zistíte akú IP adresu získal od *DHCP* servera stroj s Windows 10 a pridajte príslušný zápis do súboru */etc/hosts* na obidvoch počítačoch s CentOS 7; mal by mať tvar (IP adresu nahraďte zistenou IP adresou):
10.200.0.144 win10
4. Na stroji s Windows 10 zeditujte ako administrátor súbor *hosts*, ktorý sa nachádza v *C:\Windows\System32\drivers\etc* a dopíšte doňho záznamy na počítače *client*, *server*, *win10*.
5. Na stroji s Windows 10 si stiahnite inštalátor programu *PuTTY* a nainštalujte ho.
6. Počítače s OS Windows predvolene neodpovedajú na *ICMP* správy typu *echo-request*, skontrolujte preto, či je na počítači s Windows 10 vypnutý firewall.
7. Pingom overte, že všetky tri stroje vedia spolu komunikovať prostredníctvom IP adres aj hostiteľských mien.
8. Na obidvoch počítačoch s CentOS 7 si príkazom **yum install wireshark-gnome** nainštalujte sieťový analyzátor *Wireshark*.
9. Na obidvoch počítačoch s CentOS 7 vytvorte nového používateľa s prihlasovacím menom *jahoda*. Vytvorte mu aj heslo, nech toto heslo na *serveri* a *clientovi* nie je rovnaké. Na *clientovi* v termináli príkazom **su - jahoda** zmeňte svoju identitu na používateľa *jahoda*, príkazom **ls -la** si prezrite obsah domovského priečinka tohto používateľa.
10. Príkazom **systemctl status sshd.service** si na *serveri* overte, že je služba *sshd* spustená.
11. Príkazom **iptables -L** si na *serveri* zobrazte nastavenie firewallu. Predvolene je firewall zapnutý, je v ňom ale zahrnuté pravidlo, ktoré zabezpečí, aby server prijímal prichádzajúce spojenia protokolom *SSH*. Pravidlo má nasledujúci tvar:

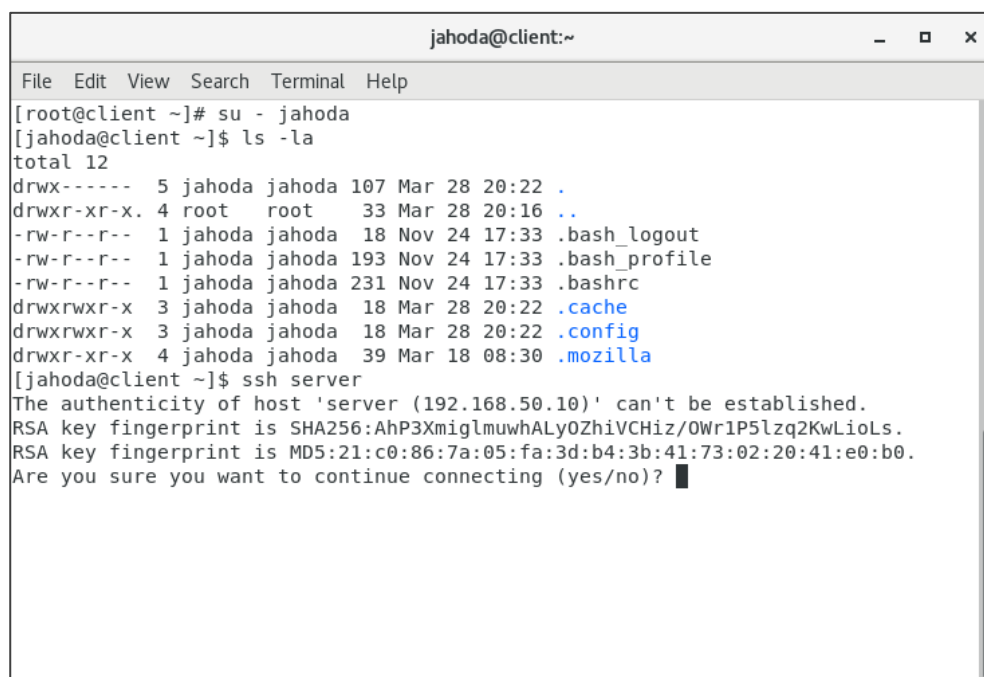
```
ACCEPT tcp -- anywhere anywhere tcp dpt:ssh ctstate
NEW,UNTRACKED
```

12. Na *serveri* sa presuňte do priečinka `/etc/ssh`, zeditujte v ňom súbor `sshd_config` a odkomentujte riadok s `RSA` kľúčom, zvyšné riadky s kľúčmi ponechajte zakomentované. Príslušná časť má po úprave vyzeráť nasledovne:

```
HostKey /etc/ssh/ssh_host_rsa_key
#HostKey /etc/ssh/ssh_host_ecdsa_key
#HostKey /etc/ssh/ssh_host_ed25519_key
```

Cieľom je, aby *server* na svoju autentizáciu používal namiesto `ECDSA`, resp. `ED25519` kľúčov `RSA` kľúče. Reštartujte `SSH` server príkazom **systemctl restart ssh.service**.

13. Z *clienta* z účtu *jahoda* nadviažte spojenie s `SSH` serverom na počítači *server* pomocou príkazu **ssh server** (namiesto hostiteľského mena *server* môžete použiť aj jeho IP adresu). Keďže sa na tento počítač ako používateľ *jahoda* prihlasujete z *clienta* prvýkrát, zjaví sa varovné hlásenie ako na Obrázku 1.

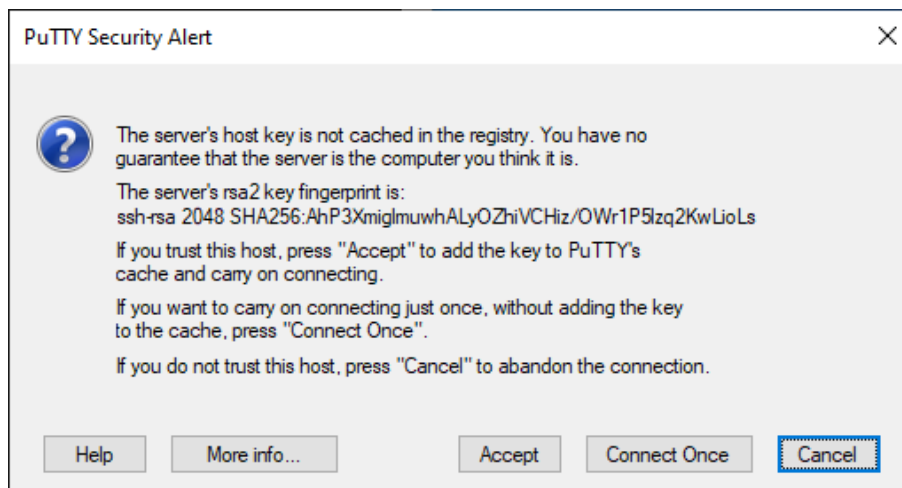


```
jahoda@client:~
File Edit View Search Terminal Help
[root@client ~]# su - jahoda
[jahoda@client ~]$ ls -la
total 12
drwx----- 5 jahoda jahoda 107 Mar 28 20:22 .
drwxr-xr-x. 4 root root 33 Mar 28 20:16 ..
-rw-r--r-- 1 jahoda jahoda 18 Nov 24 17:33 .bash_logout
-rw-r--r-- 1 jahoda jahoda 193 Nov 24 17:33 .bash_profile
-rw-r--r-- 1 jahoda jahoda 231 Nov 24 17:33 .bashrc
drwxrwxr-x 3 jahoda jahoda 18 Mar 28 20:22 .cache
drwxrwxr-x 3 jahoda jahoda 18 Mar 28 20:22 .config
drwxr-xr-x 4 jahoda jahoda 39 Mar 18 08:30 .mozilla
[jahoda@client ~]$ ssh server
The authenticity of host 'server (192.168.50.10)' can't be established.
RSA key fingerprint is SHA256:Ahp3XmiglmuwHAlY0ZhiVCHiz/OWr1P5lzq2KwLioLs.
RSA key fingerprint is MD5:21:c0:86:7a:05:fa:3d:b4:3b:41:73:02:20:41:e0:b0.
Are you sure you want to continue connecting (yes/no)?
```

Obrázok 1 Varovné okno pri prvom nadväzovaní spojenia s `SSH` serverom linuxovým `SSH` klientom.

14. Z počítača *client* sa zatiaľ neprihlasujte, nadviažte spojenie so *serverom* aj zo stroja s Windows 10; ako klientský softvér použite program *PuTTY*; do poľa *Host Name or IP Address* zadajte jeho hostiteľské meno *server* alebo jeho IP adresu. Keďže sa na tento počítač prihlasujete programom *PuTTY* prvýkrát, zjaví

sa varovné okno ako na Obrázku 2. Zistíte aká možnosť je v okne programu *PuTTY* navyše oproti možnostiam *yes/no*, ktoré boli na počítači *client*. Porozmýšľajte, prečo nie je vhodné túto voľbu použiť.



Obrázok 2 Varovné okno pri prvom nadväzovaní spojenia s SSH serverom v programe *PuTTY*.

V *PuTTY* sú oproti linuxovému SSH klientovi na výber 3 možnosti:

- Kliknutie na tlačidlo *Accept* má rovnakú funkciu ako možnosť *yes* v SSH klientovi v CentOS 7; táto voľba spôsobí, že sa verejný kľúč SSH servera uloží na klientský počítač a používateľ pokračuje v nadväzovaní spojenia.
- Kliknutie na tlačidlo *Cancel* má rovnakú funkciu ako možnosť *no* v SSH klientovi v CentOS 7; táto voľba spôsobí, že sa spojenie ukončí.
- Kliknutie na tlačidlo *Connect Once* spôsobí, že sa verejný kľúč SSH servera na klientský počítač neuloží a používateľ môže napriek tomu pokračovať v nadväzovaní spojenia. Pri používaní tejto možnosti sa pri každom pripojení znovu zobrazuje varovné okno; používateľ si ho postupne prestane všimnúť a môže sa stať ľahkým terčom útoku.

Porovnajte odtlačky verejného kľúča servera na jednotlivých klientských počítačoch; ak boli generované rovnakým algoritmom, mali by byť rovnaké. V reálnom živote by ste sa mali presvedčiť, že sa omylom nepripájate na cudzí server. Odtlačok verejného kľúča vzdialeného počítača by ste mali mať niekde zapísaný, alebo pred pripojením pre istotu kontaktovať správcu vzdialeného počítača. Na serveri vygenerujte odtlačok jeho verejného kľúča príkazom **ssh-**

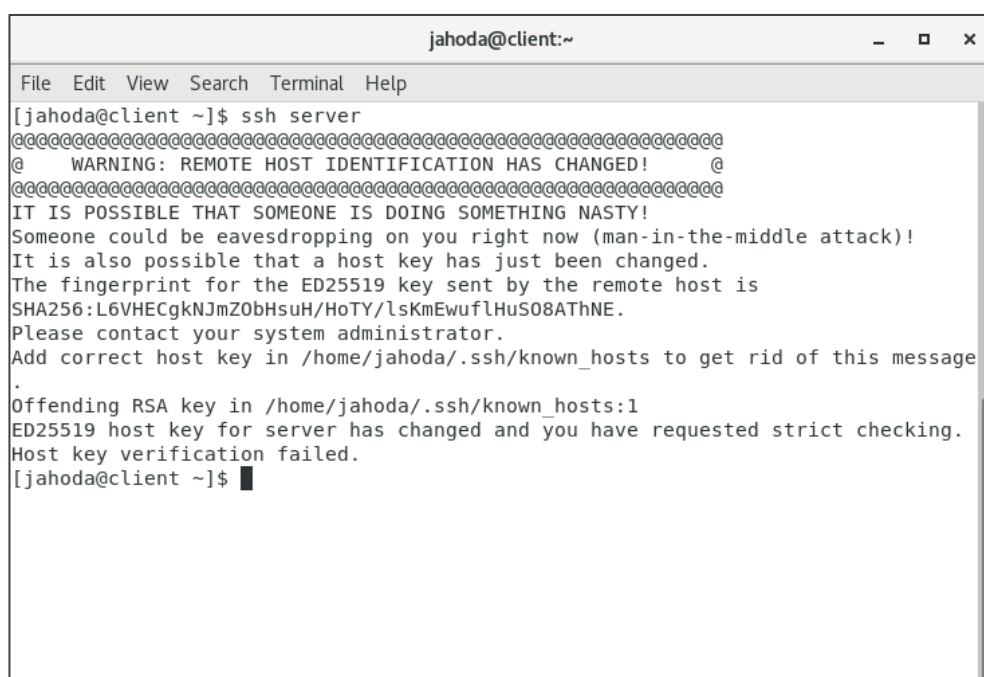
keygen -lf /etc/ssh/ssh_host_rsa_key.pub a výsledok porovnajte s odtlačkom vytvoreným algoritmom *SHA256*, ktorý je zobrazený v termináli *clienta* a v okne programu *PuTTY*.

15. V termináli *clienta* zadajte **yes**, aby ste akceptovali verejný kľúč *servera* a mohli sa pripojiť. Ak vám medzitým vypršal čas na pripojenie, pripojte sa znova a zadajte heslo používateľa *jahoda*, ktoré má nastavené na *serveri*.
16. Príkazom **exit** v termináli *clienta* sa odhláste zo *servera*. Príkazom **ls -la** si opäť zobrazte obsah domovského priečinka používateľa *jahoda*. Pribudol v ňom priečinok *.ssh*, ktorý obsahuje súbor *known_hosts*. Zobrazte si obsah tohto súboru.
17. Prihláste sa rovnakým spôsobom z *clienta* na *server* znova. Varovanie sa už neobjaví, lebo verejný kľúč *servera* už máte uložený v súbore *known_hosts*. Potom sa príkazom **exit** v termináli *clienta* odhláste zo *servera*.
18. Na stroji s Windows 10 v okne programu *PuTTY* kliknite na tlačidlo *Accept*, aby ste prijali verejný kľúč *servera* a mohli sa pripojiť. Ak vám medzitým vypršal čas na pripojenie, pripojte sa znova a zadajte prihlasovacie údaje používateľa *jahoda* na *serveri*.
19. Zavrite okno programu *PuTTY*, potom ho opäť spustíte a prihláste sa na *server* znova. Varovanie sa už neobjaví, lebo verejný kľúč *servera* už máte uložený. Potom okno programu *PuTTY* zavrite.
20. Verejný kľúč *SSH* *servera* sa v OS Windows ukladá do registrov. Spustíte na stroji s Windows 10 program *regedit*, presuňte sa v ňom do *HKEY_CURRENT_USER\Software\SimonTatham\PuTTY\SshHostKeys* a prezrite si informácie o uloženom kľúči.
21. Žiadna varovné hlásenie by sa už pri opakovanom pripojení nemalo objaviť. Ak by sa predsa len objavilo, môže to znamenať pokus o útok alebo to, že sa používateľ omylom prihlasuje na iný *SSH* server. Upravujte postupne nastavenia *SSH* *servera* tak, aby ste sa oboznámili so situáciami, ktoré môžu pri pripájaní nastať. Postupujte podľa nasledujúcich pokynov:
 - a) Na *serveri* zeditujte súbor *sshd_config* a tentokrát z riadkov s kľúčmi ponechajte odkomentovaný iba riadok:

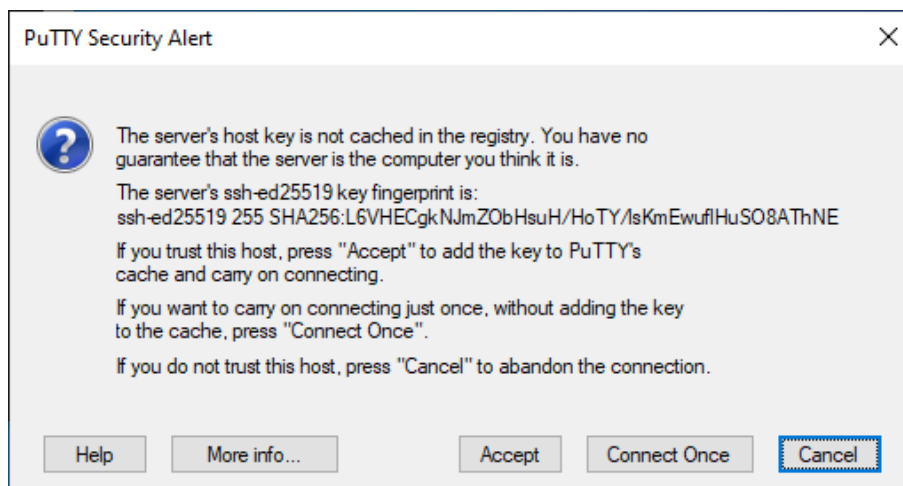
HostKey /etc/ssh/ssh_host_ed25519_key

Cieľom je, aby *server* na svoju autentizáciu používal namiesto *RSA* kľúčov *ED25519* kľúče. Reštartujte *SSH* server príkazom **systemctl restart sshd.service**.

- b) Z počítača *client* ako používateľ *jahoda* nadviažte *SSH* spojenie so *serverom*; nepripájajte sa, sledujte iba varovné hlásenie.
- c) *SSH* spojenie so *serverom* nadviažte aj zo stroja s Windows 10. Ani tu sa nepripájajte, sledujte iba varovné hlásenie.
- d) Rozdiely vo varovných hláseniach sú zobrazené na Obrázkoch 3 a 4. Pozorne si ich preštudujte.

A screenshot of a terminal window titled 'jahoda@client:~'. The terminal shows the command '[jahoda@client ~]\$ ssh server' being executed. The output is a warning message from the SSH client, enclosed in a box of asterisks. The message states: '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 ED25519 key sent by the remote host is SHA256:L6VHECgkNJmZ0bHsuH/HoTY/lSkmEwufLHuS08AThNE. Please contact your system administrator. Add correct host key in /home/jahoda/.ssh/known_hosts to get rid of this message.' Below the warning, it says 'Offending RSA key in /home/jahoda/.ssh/known_hosts:1' and 'ED25519 host key for server has changed and you have requested strict checking. Host key verification failed.' The prompt returns to '[jahoda@client ~]\$'.

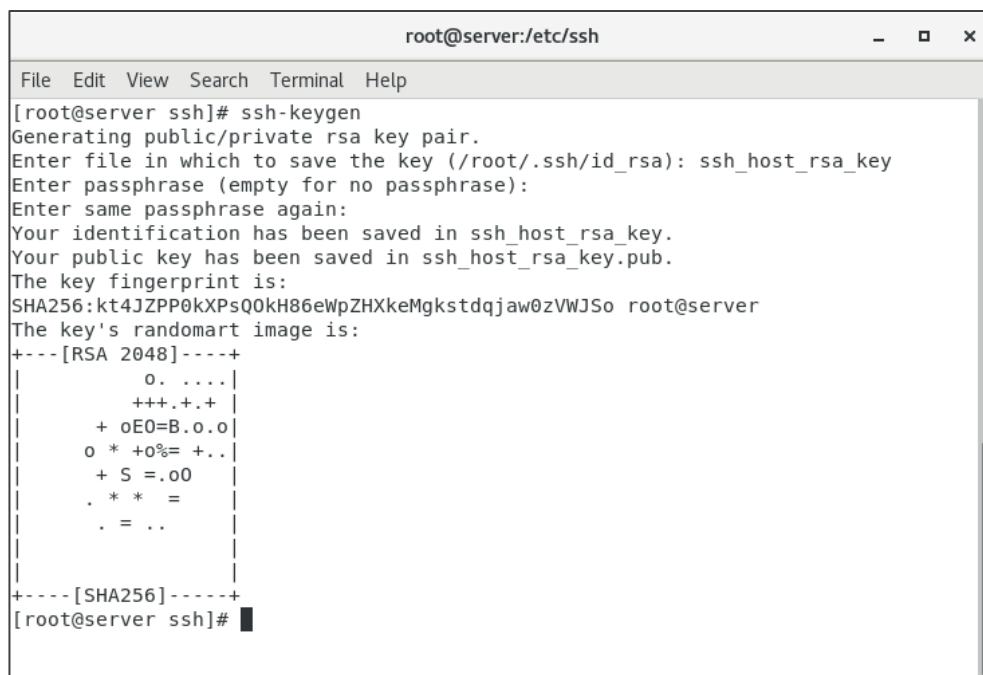
Obrázok 3 Varovné okno v *SSH* klientovi v CentOS 7 v prípade, že klient už akceptoval verejný *RSA* kľúč servera a server sa mu pokúša autentizovať pomocou *ED25519* kľúča.



Obrázok 4 Varovné okno v *PuTTY* v prípade, že klient už akceptoval verejný RSA kľúč servera a server sa mu pokúša autentizovať ED25519 kľúčom.

SSH klient v CentOS 7 používateľa upozornil, že pre daný SSH server už akceptoval iný typ kľúča a spojenie so serverom automaticky ukončil. Ak sa chce používateľ pripojiť, musí manuálne vymazať predchádzajúci verejný kľúč servera a pri ďalšom pripojení akceptovať nový. Varovné okno v *PuTTY* je rovnakého typu ako pri prvom pripojení na server. Klikline v ňom na tlačidlo *Cancel*.

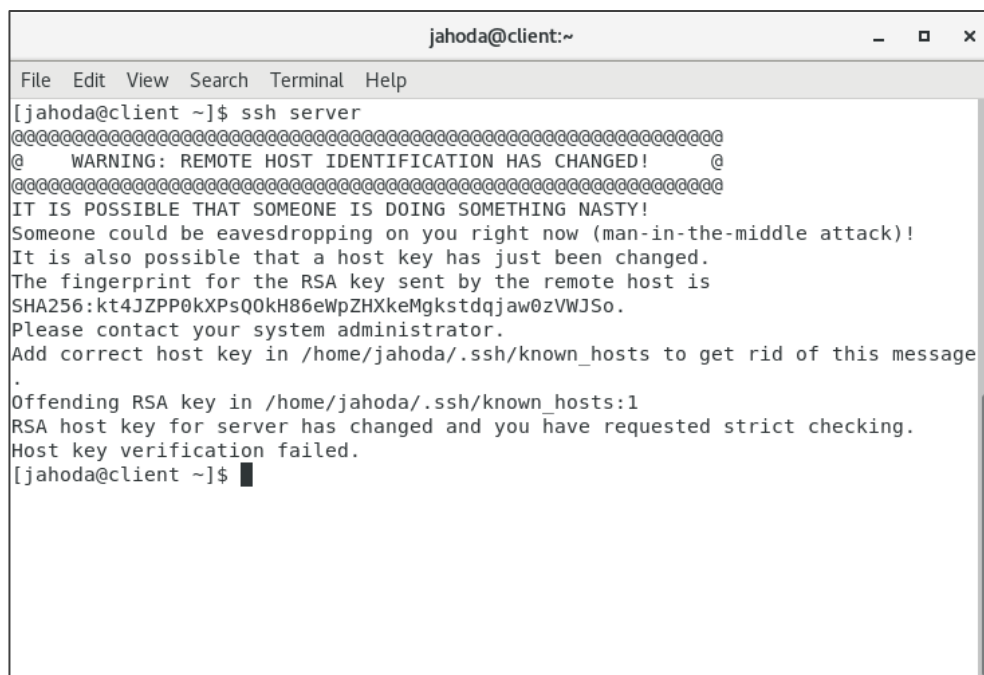
- e) Na serveri znova zeditujte súbor *sshd_config* a odkomentujte resp. zakomentujte v ňom príslušné riadky tak, aby sa server opäť autentizoval iba dvojicou RSA kľúčov. Príkazom **systemctl restart sshd.service** reštartuje SSH server. Pôvodné RSA kľúče premenujte na *ssh_host_rsa_key.backup* a *ssh_host_rsa_key.pub.backup*. Potom príkazom **ssh-keygen** vygenerujte nové RSA kľúče, k privátnemu kľúču nevytvárajte žiadnu *passphrase* (výzvu na zadanie *passphrase* odenterujte). Vytvoreným kľúčom na výzvu zmeňte mená na *ssh_host_rsa_key* a *ssh_host_rsa_key.pub*. Postup je zobrazený na Obrázku 5.



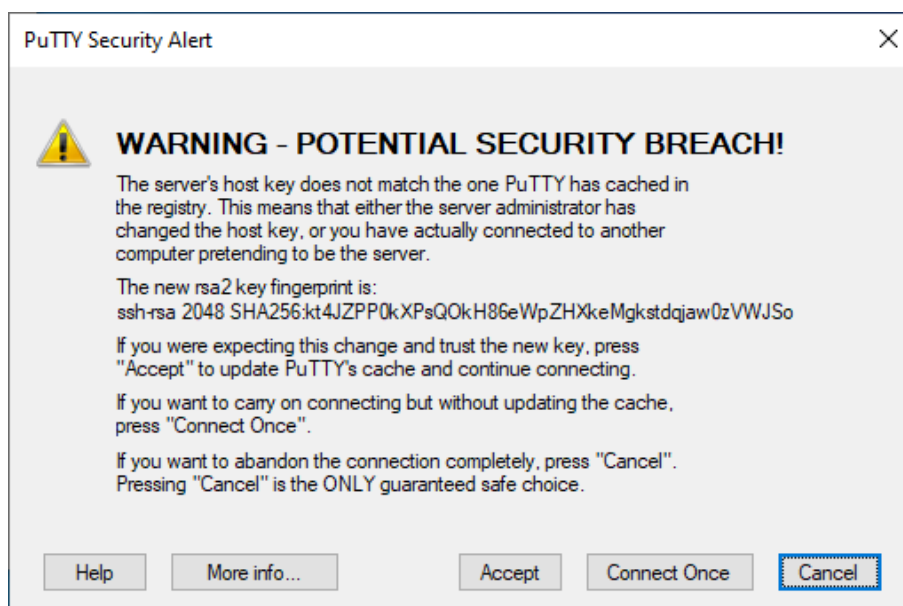
```
root@server:/etc/ssh
File Edit View Search Terminal Help
[root@server ssh]# ssh-keygen
Generating public/private rsa key pair.
Enter file in which to save the key (/root/.ssh/id_rsa): ssh_host_rsa_key
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in ssh_host_rsa_key.
Your public key has been saved in ssh_host_rsa_key.pub.
The key fingerprint is:
SHA256:kt4JZPP0kXP0kH86eWpZHXkeMgkstdqjaw0zVWJS0 root@server
The key's randomart image is:
+---[RSA 2048]---+
|                 |
|      o . . . .  |
|      + + + . + + |
|    + oE0=B.o.o |
|  o * +o%= +. . |
|    + S =.o0    |
|      . * * =   |
|      . = . .   |
|                 |
+-----[SHA256]-----+
[root@server ssh]#
```

Obrázok 5 Generovanie novej dvojice *RSA* kľúčov na *serveri*.

- f) Z počítača *client* ako používateľ *jahoda* nadviažete *SSH* spojenie so *serverom*; nepripájajte sa, sledujte iba varovné hlásenie.
- g) *SSH* spojenie so *serverom* nadviažete aj z počítača s Windows 10. Nepripájajte sa, sledujte iba varovné hlásenie.
- h) Rozdiely vo varovných hláseniach sú zobrazené na Obrázkoch 6 a 7. Pozorne si ich preštudujte.



Obrázok 5 Varovné okno v SSH klientovi v CentOS 7 v prípade, že klient už akceptoval verejný RSA kľúč servera a kľúč bol pregenerovaný.



Obrázok 6 Varovné okno v PuTTY v prípade, že klient už akceptoval verejný RSA kľúč servera a kľúč bol pregenerovaný.

Varovné hlásenia v SSH klientovi v CentOS 7 a v PuTTY majú približne rovnaký charakter. SSH klient v CentOS 7 používateľa upozornil, že pre daný SSH server už akceptoval iný RSA kľúč a spojenie so serverom automaticky ukončil. Ak sa chce používateľ pripojiť, musí manuálne

vymazať predchádzajúci verejný kľúč servera a pri ďalšom pripojení akceptovať nový. *PuTTY* umožňuje kliknutím na tlačidlo *Accept* akceptovať nový verejný kľúč servera a nahradiť ním starý. Používateľ, ktorý nevenuje pozornosť varovným hláškam sa tak ľahko môže stať terčom útoku. V okne programu *PuTTY* kliknite na tlačidlo *Cancel*.

- i) Na *serveri* zmažte novovytvorené *RSA* kľúče a pôvodným odstráňte z názvu reťazec *.backup*, ktorý ste tam predtým pridali. Potom zeditujte súbor *sshd_config* a tentokrát z riadkov s kľúčmi ponechajte odkomentovaný iba riadok:

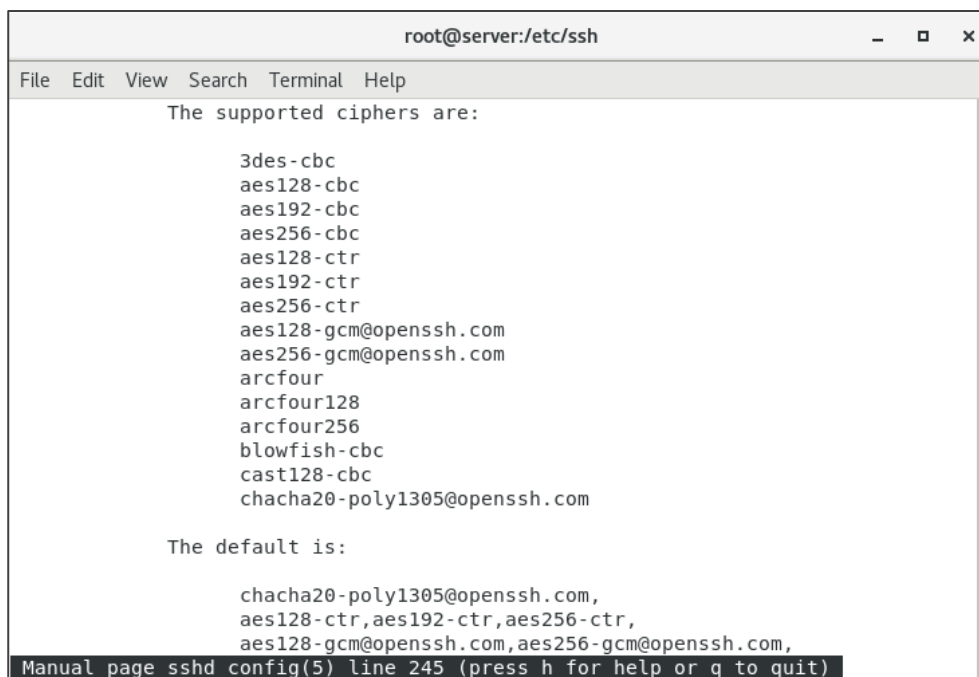
HostKey /etc/ssh/ssh_host_ecdsa_key

Príkazom **sudo systemctl restart ssh.service** reštartuje *SSH* server.

- j) Z počítača *client* ako používateľ *jahoda* nadviažte *SSH* spojenie so *serverom*. Keďže máte pre *server* uložený iný typ kľúča, v termináli sa objaví príslušné varovanie a spojenie sa ukončí.
- k) Na počítači *client* zmažte obsah súboru */home/jahoda/.ssh/known_hosts*, prípadne zmažte celý súbor. Potom ako používateľ *jahoda* znovu nadviažte *SSH* spojenie so *serverom*, akceptujte nový *ECDSA* kľúč a prihláste sa. Po úspešnom prihlásení sa od *SSH* servera odpojte.
- l) *SSH* spojenie so *serverom* nadviažte aj zo stroja s Windows 10. Keďže tento typ kľúča nemáte v *PuTTY* uložený, zjaví sa rovnaké varovanie, ako keby ste sa na *server* pripájali prvý krát. Kliknite na tlačidlo *Accept*, prijmete tým nový *ECDSA* verejný kľúč servera. Zadaťte prihlasovacie údaje používateľa *jahoda* na *serveri* a prihláste sa. Po úspešnom prihlásení zavrite okno programu *PuTTY*.
- m) Na stroji s Windows 10 spustíte program *regedit*, overte si, že v *HKEY_CURRENT_USER\Software\SimonTatham\PuTTY\SshHostKeys* máte pre počítač *server* uložený aj *RSA* aj *ECDSA* verejný kľúč.

22. Stroj s Windows 10 už môžete vypnúť.

23. Súbor */etc/ssh/sshd_config* je dobre okomentovaný; napriek tomu si na *serveri* zobrazte podrobnejšie informácie o možnostiach konfigurácie *SSH* servera príkazom **man sshd_config**. Na Obrázku 7 je časť výpisu získaného týmto príkazom obsahujúca šifrovacie algoritmy, ktoré *SSH* server podporuje.



```
root@server:/etc/ssh
File Edit View Search Terminal Help

The supported ciphers are:

3des-cbc
aes128-cbc
aes192-cbc
aes256-cbc
aes128-ctr
aes192-ctr
aes256-ctr
aes128-gcm@openssh.com
aes256-gcm@openssh.com
arcfour
arcfour128
arcfour256
blowfish-cbc
cast128-cbc
chacha20-poly1305@openssh.com

The default is:

chacha20-poly1305@openssh.com,
aes128-ctr,aes192-ctr,aes256-ctr,
aes128-gcm@openssh.com,aes256-gcm@openssh.com,
Manual page sshd config(5) line 245 (press h for help or q to quit)
```

Obrázok 7 Prehľad šifrovacích algoritmov, ktoré podporuje *OpenSSH* server.

24. Na počítači *client* si podobne zobrazte možnosti konfigurácie *SSH* klienta príkazom **man ssh_config**; podpora šifrovacích algoritmov by mala byť rovnaká.
25. Pred ďalším pripojením na *server* z *clienta* si v niektorom z týchto strojov z menu *Applications – Internet – Wireshark Network Analyzer* spustíte program *Wireshark* a nechajte ho načúvať na sieťovej karte *ens192*; potom sa ako používateľ *jahoda* prihláste prostredníctvom *SSH* z *clienta* na *server* a použite nejaký príkaz, napr. **pwd**.
26. Vo *Wiresharku* označte v hornej časti riadok s protokolom *SSH2*, ktorý má v stĺpci *Info* popis *Client: Key Exchange Init*, v strednej časti rozkliknite niekoľkokrát ponuku *SSH protocol* a prezrite si zoznam bezpečnostných algoritmov, ktoré ponúkol počítač *client* počítaču *server*. Potom kliknite na riadok, ktorý má v stĺpci *Info* popis *Client: Encrypted packet* a overte, že ako šifrovací algoritmus sa použil prvý z predvolených šifrovacích algoritmov, v tomto prípade to bol algoritmus *chacha20-poly1305@openssh.com*. Potom sa príkazom **exit** v termináli *clienta* odhláste zo *servera*.
27. Zastavte *Wireshark* a spustíte ho znovu, potom sa z *clienta* prihláste na *server* ako používateľ *jahoda* pomocou príkazu **ssh -c aes128-ctr server**. Cieľom tohto

- príkazu je, aby sa na šifrovanie dátovej prevádzky používal v poradí druhý z predvolených šifrovacích algoritmov.
28. Vo *Wiresharku* si overte, že ako šifrovací algoritmus sa naozaj použil *aes128-ctr*. Potom sa príkazom **exit** v termináli *clienta* odhláste zo *servera*.
 29. Ako *root* vytvorte na *clientovi* ďalšieho používateľa s prihlasovacím menom *malina* a v termináli zmeňte identitu na tohto používateľa.
 30. Prihláste sa z *clienta* z účtu *malina* na *server* ako *jahoda* príkazom **ssh jahoda@server**. Keďže sa z *clienta* ako používateľ *malina* prihlasujete na *server* prvý krát, opäť sa vám zobrazí varovné hlásenie. Po pripojení sa príkazom **exit** v termináli *clienta* odhláste zo *servera* a prezrite si obsah súboru */home/malina/.ssh/known_hosts*.
 31. Na *clientovi* sa z účtu *malina* pripojte na *server* ako *jahoda*, tento krát použite príkaz **ssh -l jahoda server**. Po úspešnom prihlásení sa zo *servera* odhláste a v termináli zmeňte svoju identitu na používateľa *jahoda*.
 32. Overte si, že predvolene je povolené prihlasovať sa na *SSH* server ako *root*. Prihláste sa z *clienta* z účtu *jahoda* na *server* prihlasovacím menom *root* príkazom **ssh root@server** a zadajte heslo *roota*, ktoré používa na *serveri*. Potom sa príkazom **exit** v termináli *clienta* odhláste zo *servera*.
 33. Na *serveri* zeditujte súbor */etc/ssh/sshd_config*. Odkomentujte riadok *#PermitRootLogin yes*, zmeňte ho na *PermitRootLogin no* a reštartujte službu *sshd*.
 34. Pokúste sa z *clienta* prihlásiť na *server* prihlasovacím menom *root*. Aj keď zadávate správne heslo, pripojiť sa vám nepodarí.
 35. Na niektorých systémoch je nastavené, aby sa *root* nemohol prihlasovať pomocou *SSH* priamo, najskôr sa musí prihlásiť ako bežný používateľ a až potom zmeniť svoju identitu na *roota*. (Prípadne má tento používateľ špecifikované svoje práva na *serveri* v súbore */etc/sudoers* a môže niektoré príkazy spúšťať pomocou *sudo*). Pripojte sa z *clienta* na *server* ako používateľ *jahoda*. Po pripojení zmeňte na *serveri* svoju identitu na superužívateľa *root*. Potom príkazom **exit** zmeňte v termináli svoju identitu naspäť na používateľa *jahoda* a ďalším príkazom **exit** sa odhláste zo *servera*.

36. Na *serveri* vytvorte používateľa *cernica*. Potom zeditujte súbor */etc/shadow* a z nastavenia jeho účtu zmažte znaky *!* z druhého poľa, aby sa tento používateľ mohol prihlásiť bez hesla.
37. Z *clinta* sa z účtu *jahoda* prihláste na *server* do účtu *jahoda*, po prihlásení zmeňte svoju identitu na *cernica*; na zmenu identity nie je potrebné zadať heslo. Príkazom **exit** dvakrát po sebe sa odpojte od *servera*.
38. Z *clinta* sa z účtu *jahoda* pokúste prihlásiť na *server* priamo do účtu *cernica*. Prihlásenie sa nepodarí; aj keď používateľ *cernica* nemá na *serveri* nastavené heslo, systém ho požaduje zadať.
39. Na *serveri* zeditujte súbor */etc/ssh/sshd_config*. Odkomentujte riadok *#PermitEmptyPasswords no*, zmeňte ho na *PermitEmptyPasswords yes* a reštartujte službu *sshd*.
40. Z *clinta* sa z účtu *jahoda* pokúste prihlásiť na *server* priamo do účtu *cernica*. Prihlásenie sa podarí bez potreby zadávania hesla.
41. Súbor */etc/ssh/sshd_config* na *serveri* vráťte do pôvodného stavu.
42. Na *serveri* odstráňte používateľské účty *jahoda* a *cernica*, na *clintovi* účty *jahoda* a *malina*.