Bezpečné přihlášení k SSH

pomocí hardwarových tokenů

Petr Krčmář

ROOT.CZ_ vpsFree.cz

6. března 2021

O mně

- linuxák od roku 1998
- správce serverů
- lektor a konzultant
- šéfredaktor Root.cz
- člen vpsFree.cz
- organizátor LinuxDays
- můj web je petrkrcmar.cz

ROOT.CZ_ vpsFree.cz

Prezentace už teď na webu

https://www.petrkrcmar.cz

ROOT.CZ_ vpsFree.cz

SSH v kostce

Dálkový přístup přes SSH

- šifrovaný bezpečný komunikační kanál (port 22)
- šifrované, autentizované, velmi bezpečné
- terminálový přístup, i přenos souborů
- další služby: scp, sftp, mosh, rsync…
- pro každé spojení se používá jiný symetrický klíč
  - algoritmus Diffie-Hellman pro bezpečný přenos
- obě strany vzájemně ověřují svou identitu
  - klient ověřuje otisk veřejného klíče serveru
  - server autentizuje uživatele heslem nebo klíčem

ROOT.CZ_ vpsFree.cz

Přihlašování heslem

- nejjednodušší varianta, ale zároveň nejméně bezpečná
- klient odešle své uživatelské jméno a heslo
- heslo je třeba pokaždé znovu zadávat
  - zároveň je nutné mít heslo unikátní a silné
- nedůsledné ověření identity serveru = vyzrazení hesla
- prostor pro hádání hesla libovolným útočníkem
  - každý správce to vidí v logu
- možnost vylepšení pomocí interaktivní autentizace klávesnicí
  - server může interaktivně komunikovat s klientem
  - lze přidat například další faktor (OTP)

ROOT.CZ_ vpsFree.cz

Přihlašování klíčem

- klient se může představit elektronickým podpisem
- má pár klíčů (veřejný a soukromý) a veřejný předá serveru
- soukromý slouží k přihlášení, veřejný k ověření
  - jeden veřejný může být na mnoha serverech
  - nelze jej použít k přihlášení
  - můžete si ho vystavit na webu

Generování klíčů u klienta

```
$ ssh-keygen -t rsa -b 4096 -C uzivatel@stanice
```

- klíče jsou v `~/.ssh/id_*`
- na serveru jdou do `~/.ssh/authorized_keys`

ROOT.CZ_ vpsFree.cz

Algoritmus přihlášení klíčem

- klient a server se dohodnou na symetrické šifře
- klient sdělí serveru, jaký veřejný klíč chce použít
- server odmítne nebo přijme použití daného klíče
- klient vytvoří autentizační zprávu za použití soukromého klíče
  - obsahuje uživatelské jméno
  - název služby
  - unikátní identifikátor relace
  - název algoritmu a samotný podpis
- server ověří použitelnost klíče a správnost podpisu
- server teď ví, že klient drží správný soukromý klíč a přihlásí ho

ROOT.CZ_ vpsFree.cz

Čtyři asymetrické šifry

- SSH zná čtyři různé algoritmy a tím i klíče
- DSA, RSA, ECDSA a Ed25519
  - DSA je opuštěný od OpenSSH 7.0 (2015)
- ECDSA a Ed25519 používají eliptické křivky
  - podpora ECDSA od OpenSSH 5.7 (2011)
  - podpora Ed25519 od OpenSSH 6.5 (2014)
  - výrazně menší klíče (256, 384, 512 bitů)
- klíče jsou uloženy na disku v souborech `~/.ssh/id_*`
- soukromé části chráněny šifrou AES-128 s blokovým režimem CTR (dříve CBC)
- pro dešifrování zadáváme frázi (možno uložit do SSH agenta)

ROOT.CZ_ vpsFree.cz

Zranitelné klíče

- klíč uložený v počítači je možné teoreticky ukrást
- šifrovaný leží v běžném souboru, dešifrovaný je v RAM
- malware je teoreticky schopen klíče ukrást
- v roce 2016 objeven útok na funkci roaming
  - zlomyslný server mohl vyčíst soukromý číst z klienta
  - zneužíval přitom chybu v nepoužívané funkci roaming
- v praxi ale většina adminů klíče v souborech používá
- je to pořád velmi bezpečné

Klíče v tokenu

Nedobytné klíče

- pokud si chceme být jisti, můžeme to dělat lépe
- můžeme klíče uložit na oddělený hardware (token)
- ten drží soukromý klíč a **nikdy ho nevydá**
- klíč je navždy v tokenu a ten jej na požádání používá
- velmi bezpečné, napadením počítače nezískáme nikdy tajemství
- token YubiKey umí vytvářet klíče pro PGP
  - ty je možné použít i pro přihlašování na SSH
  - vyžaduje ale gpg-agent a dražší klíč YubiKey
- novější OpenSSH ale umí použít levné U2F/FIDO2 tokeny

ROOT.CZ_ vpsFree.cz

Krátce o U2F/FIDO2

- standard původně vyvinutý v Google a YubiCo
- dnes je rozvíjen pod organizací FIDO Aliance
- umožňuje využít USB či NFC tokeny jako druhý faktor
- obvykle se používá na webu (WebAuthn)
- token obsahuje **unikátní tajemství**
- služba umí ověřit, že držíte právě ten jedinečný token
- komplexnější tokeny umí i U2F: YubiKey, Trezor, SoloKey…
- existují ale mnohem levnější klíče jen s U2F
  - o těch bude dále řeč

Registrace tokenu

- token se nejprve registruje do služby
- registrace = první představení službě
  - služba vygeneruje náhodnou výzvu (nonce) a uloží si ji
  - výzva se v tokenu zkombinuje se soukromým klíčem a doménou
    - použije se HMAC-SHA256
  - tím vzniká v dočasné paměti tokenu nový soukromý klíč
  - z něj je odvozen veřejný klíč a ten je poslán službě
  - služba teď zná náš veřejný klíč vytvořený na základě nonce
- každá služba dostane **jiný veřejný klíč**, pošle jinou nonce
- do tokenu se nic trvale neukládá = neomezený počet služeb

Přihlášení s tokenem

- přihlášení = použití známého tokenu
  - služba vygeneruje náhodnou přihlašovací výzvu (jednározovou)
  - společně s nonce ji pošle klientovi
  - token opět v paměti sestaví soukromý klíč pro službu
  - soukromým klíčem **podepíše** jednorázovou výzvu
  - server pomocí uloženého veřejného klíče zprávu ověří
  - server nyní ví, že klient drží správný token
  - klíč neopustil token a nedostal se ani do počítače
- akce se potvrzují tlačítkem (volitelně PIN)
- YubiKey BIO bude mít čtečku otisků

ROOT.CZ_ vpsFree.cz

U2F v OpenSSH

Podpora v OpenSSH

- v únoru 2020 vyšlo OpenSSH 8.2
- přineslo mimo jiné podporu přihlašování s U2F
- zavádí dva nové typy klíčů **ecdsa-sk** a **ed25519-sk** (security key)
- všechny tokeny umí ECDSA, podpora Ed25519 je jen v některých
- princip je stejný: registrace na serveru, přihlašování
- využíváme stejných výhod: klíč nikdy neopouští token
- server i klient musí mít OpenSSH 8.2 či vyšší
  - Debian 11, Ubuntu 20.04 LTS

ROOT.CZ_ vpsFree.cz

Registrace tokenu

- registrace vypadá stejně jako generování klíčů
- použijeme stejný postup, jen jiný typ klíče

Generování klíčů u klienta

```
$ ssh-keygen -t ecdsa-sk
```

- dostaneme dva soubory, jako obvykle
  - `id_ecdsa_sk` a `id_ecdsa_sk.pub`
- žádný z nich **neobsahuje soukromý klíč**
- místo něj je v souboru výzva (nonce) pro sestavení klíče v tokenu
- veřejný klíč je na svém místě, jako obvykle

ROOT.CZ_ vpsFree.cz

Použití tokenu

- veřejný klíč nahrajeme klasicky na server do `authorized_keys`
- že jde o SK klíče poznáme podle typu na začátku řádku

Příklad veřejného klíče

```
sk-ecdsa-sha2-nistp256@openssh.com AAAAInNrL…
```

- server nesmí mít klíč vypnuté (volba `PubkeyAcceptedKeyTypes`)
- přihlášení pak proběhne standardně
  - musíme potvrdit tlačítkem na tokenu

ROOT.CZ_ vpsFree.cz

Praktická ukázka

Bezpečné klíče

- počítač neobsahuje celé tajemství
- malware by mohl ukrást nonce, ale bez tokenu je k ničemu
- stejně tak obráceně: samotný token je nepoužitelný
- jen kombinace nonce a fyzického tokenu dá dohromady klíč
- nutnost potvrdit použití klíče snižuje riziko tichého zneužití

Záloha tokenu

- soukromý klíč nelze z tokenu nijak získat
- riziko: ztráta či poškození tokenu
- doporučuje se mít tokenů více = více veřejných klíčů
- můžeme ale kombinovat i s klasickým záložním klíčem
  - ten máme v trezoru (třeba vytištěný QR kód)
  - pokud token ztratíme, přejdeme na záložní klíč
  - po pořízení nového tokenu provedeme rotaci

ROOT.CZ_ vpsFree.cz

Děkuji za pozornost

Otázky?

Petr Krčmář
petr.krcmar@iinfo.cz

ROOT.CZ_ vpsFree.cz