ISA - Laboratorní cvičení č. 2

Zabezpečený přenos dat

Vysoké učení technické v Brně

`https://github.com/nesfit/ISA/tree/master/encrypted_transfers_online`

Cíl laboratorního cvičení:

- Základní seznámení se synchronizací času a protokolem NTP.
- Naučit se práci s nástrojem SSH a správu klíčů.
- Naučit se základy protokolu TLS.
- Seznámit se s Certificate Transparency Log.

Pokyny

- Pro práci v cvičení budeme používat virtuální stroje v programu VirtualBox[1].
- Cvičení předpokládá funkční interní síť s nakonfigurovanými IPv4 adresami mezi dvěma virtuálními stroji z prvního cvičení (Sekce 3.).
- Odpovědi pište do odpovědního archu `protokol.md` který odevzdáte do WIS-u. Dostupný je na adrese `https://github.com/nesfit/ISA/blob/master/encrypted_transfers_online/protokol.md`.
- Do WIS-u budete také odevzdávat všechny zachycené `pcap` soubory.
- Nastartujte oba stroje a přihlaste se do OS GNU/Linux, user/password `user`/`user4lab`.
- V případě potřeby se přepněte na uživatele `root` příkazem `su` (switch user), heslo `root4lab`.
- V případě potřeby si otevřete další terminál v novém okně.
- Pro editaci konfiguračních souborů použijte libovolný editor (např. nano, vim, gedit).

1 NTP

Vašim úkolem je zajistit synchronizaci hodin počítače. V rámci bezpečnosti je důležité provozovat počítač se správným časem kvůli časovým razítkům označujícím platnost a neplatnost klíčů, certifikátů, podpisů apod.

1. Zkontrolujte obsah konfiguračního souboru `/etc/ntp.conf`. V souboru by měli být použity pooly serveru CentOS (`*.centos.pool.ntp.org`).

[1] `https://nes.fit.vutbr.cz/isa/ISA2020.ova`

2. Zapněte démona pro NTP (`systemctl start ntpd.service`). Ověřte pomocí `systemctl status ntpd.service`, že služba běží; pokud neběží, zobrazte chyby pomocí příkazu `journalctl -u ntpd`, nalezené chyby opravte a službu `ntpd` restartujte.

3. Sledujte postup synchronizace příkazem `watch ntpq -p`. Nemusíte čekat na dokončení synchronizace, nechte okno spuštěné a pokračujte dalšími úkoly.

2 Vzdálený terminál – SSH, Secure Shell

Namísto řetězce `<login>` používejte své studentské přihlašovací jméno.

1. Otevřete si dvě okna: příkazovou řádku pro uživatele `user` a další příkazovou řádku pro uživatele `root` příkazem `su` (switch user). Pomocí příkazu `whoami` vypište jméno aktivního uživatele, ověřte, že je očekávané.

   V obou otevřených terminálech dočasně vypněte podporu pro SSH agenta:

   ```
   [user@localhost]$ unset SSH_AUTH_SOCK
   [root@localhost]# unset SSH_AUTH_SOCK
   ```

   Pokud budete otevírat v průběhu řešení úkolu nové terminály, nezapomeňte v nich také vypnout podporu agenta SSH.

2. **Bezpečné připojení na vzdálený počítač bez autentizačních klíčů.**

   (a) Spusťte program wireshark a zachytávejte komunikaci na portu 22 (rozhraní enp0s8).

   (b) Přihlaste se příkazem `ssh user@<IP adresa druhého stroje>` a zadejte heslo **user4lab**.

   (c) Na serveru zadejte libovolný příkaz, který znáte (např. zobrazte obsah manuálové stránky ssh příkazem `man ssh`, vypište obsah adresáře příkazem `ls`, obsah souborů příkazem `ls`, aktuálního uživatele příkazem `whoami` apod.).

   (d) Příkazem `exit` nebo stiskem `Ctrl-D` spojení ukončete.

   (e) V programu Wireshark zobrazte obsah komunikace (follow TCP stream) a zjistěte informace o spojení — verze programu `ssh` na serveru i klientovi, podporované šifry, autentizační mechanismus HMAC, mechanismus pro výměnu klíčů. Ověřte, že pomocí Wiresharku nevidíte příkazy zadané v předchozích bodech a jejich výstup. Můžete na základě zachycené komunikace říct něco o jejím obsahu? Wireshark nechejte dále zachytávat komunikaci.

3. **Vytvoření veřejného a privátního klíče.**

   (a) Jako uživatel `user` vygenerujte příkazem `$ ssh-keygen -C <login>@user` implicitní klíč pro uživatele `user`. Neměňte jeho název a zvolte heslo o délce alespoň osmi znaků, například **fitvutisa**.

   (b) Jako uživatel `root` vygenerujte příkazem `# ssh-keygen -N "" -C <login>@root` implicitní klíč pro uživatele `root` bez hesla.

   (c) Ověřte obsah a přístupová práva u nově vzniklých souborů (`ls -l ~/.ssh`). Jak se liší práva mezi souborem s privátním a veřejným klíčem?

4. **Distribuce klíčů**

(a) Oba veřejné klíče zkopírujte na druhý počítač do souboru `.ssh/authorized_keys` pro klíč uživatele `user` např.:
`cat ~user/.ssh/*.pub | ssh user@<IP> "cat >> .ssh/authorized_keys"`
pro klíč uživatele `root` např.:
`cat /root/.ssh/*.pub | ssh root@<IP> "cat >> .ssh/authorized_keys"`.

Pokud složka `.ssh` na vzdáleném stroji neexistuje, příkaz selže. V takovém případě se přihlaste pomocí hesla nebo přepněte do druhého stroje a složku vytvořte. Můžete také rovnou vytvořit soubor `.ssh/authorized_keys` (např. příkazem `touch`). **Je důležité aby složka měla přístupová práva `700` a soubor `600`, jinak nebude možné zkopírované klíče použít.**

Jaká hesla bylo nutné zadat?

(b) Zkuste se znovu přihlásit na druhý počítač. Jaké heslo bylo nyní nutné zadat? Zkuste zadat (opakovaně) špatné heslo a pozorujte, co se stalo. Při experimentech můžete také využít tzv. verbose režim ssh (`ssh -v`).

5. **Omezení použití klíčů**

Nyní bude naším cílem omezit použití klíče uživatele `root`, který není chráněn heslem tak, aby pomocí něj bylo možné na vzdáleném serveru vykonat pouze konkrétní příkaz.

(a) Přihlaste se jako uživatel `root` na druhý počítač, kam jste nakopírovali své veřejné klíče a upravte soubor s autorizovanými veřejnými klíči tak, že na začátek řádku s klíčem uživatele `root` (řádek poznáte tak, že končí řetězcem `<login>@root`) napíšete
`command="ntpq -p"`
(následovaný jednou mezerou a původním obsahem řádku).

(b) Odhlaste se ze vzdáleného počítače a znovu se na něj přihlaste z účtu `root` jako `root`. Aplikovalo se omezené využití klíče?

6. **Pohodlné opakované použití klíče zabezpečeného heslem.**

(a) Ukončete terminál uživatele `user` a vytvořte nový. Pomocí příkazu `env` ověřte, že je nastavená proměnná `SSH_AUTH_SOCK`.

(b) Přihlaste se na druhý počítač. Museli jste zadávat znovu heslo?

(c) Zachycenou komunikaci uložte do souboru `cv2-ssh.pcap` který budete odevzdávat.

3 Zabezpečení transportní vrstvy – TLS, Transport Layer Security

Pro tento úkol doinstalujte balíček `telnet` příkazem `sudo yum install telnet`.

1. **Nezabezpečený přenos dat**

(a) Spusťte program Wireshark, zachytávejte komunikaci na portu 80 na rozhraní enp0s3.

(b) Pomocí programu `telnet` se připojte k fakultnímu webovému serveru:
`telnet www.fit.vut.cz 80`.

(c) V programu Wireshark pozorujte navázání spojení TCP pomocí trojcestného handshaku.

(d) Zašlete serveru požadavek protokolem HTTP, např.: `GET / HTTP/1.0`, dotaz ukončete prázdným řádkem. V terminálu pozorujte odpověď.

(e) Zobrazte si v programu Wireshark komunikaci pomocí HTTP. Je možné přečíst obsah komunikace?

(f) Zachycenou komunikaci uložte do souboru `cv2-http.pcap` který budete odevzdávat.

2. **Přenos dat zabezpečený TLS**

(a) Spusťte program Wireshark, zachytávejte komunikaci na portu 443 na rozhraní enp0s3.

(b) Pomocí programu `openssl` se připojte k fakultnímu webovému serveru: `openssl s_client -quiet -connect www.fit.vut.cz:443`. Všimněte si řádku *Verify return code*, co říká o certifikátu?

(c) V programu Wireshark pozorujte navázání spojení TCP a TLS. Jaké informace lze vyčíst z handshaku TLS? Je možné zjistit jméno serveru?

(d) Zašlete serveru požadavek protokolem HTTP, např.: `GET / HTTP/1.0`, dotaz ukončete prázdným řádkem. V terminálu pozorujte odpověď.

(e) Zobrazte si v programu Wireshark komunikaci pomocí HTTP s využitím TLS. Je možné přečíst obsah komunikace?

(f) Zachycenou komunikaci uložte do souboru `cv2-tls.pcap` který budete odevzdávat.

3. **Zabezpečený přenos dat TLS v prohlížeči**

(a) Spusťte program Wireshark, zachytávejte komunikaci na portu 443.

(b) Spusťte prohlížeč Firefox a otevřete v něm stránku `wis.fit.vutbr.cz`.

(c) Kliknutím na informace o certifikátu (vlevo od URL) zobrazte informace o použitém certifikátu.

(d) V menu *Preferences* » *Privacy & Security* v sekci *Certificates* zobrazte důvěryhodné certifikáty. Prohlédněte si seznam a odhadněte jejich počet.

(e) V programu Wireshark pozorujte navázání spojení TCP a TLS. Jaké informace lze vyčíst z handshaku TLS? Je možné zjistit jméno serveru?

(f) Zachycenou komunikaci uložte do souboru `cv2-https.pcap` který budete odevzdávat.

4. **Certificate Transparency**

(a) Příkazem `host -t CAA fit.vut.cz` si zobrazte certifikační autority oprávněné vydávat certifikáty pro doménu `fit.vut.cz`.

(b) V prohlížeči se připojte na stránku `https://certstream.calidog.io/`, klikněte na tlačítko *Open Fire Hose* a pozorujte nově vydávané certifikáty. K čemu si myslíte, že můžou být tato data dobrá? Myslíte si, že je možné data zneužít?

(c) V prohlížeči se připojte na stránku `https://crt.sh`.

(d) Nalezněte certifikáty vydané pro servery v rámci domény *fit.vut.cz*, použijte znak `%` jako zástupný znak.

(e) Najděte na stránce ikonu směřují k souboru Atom odkazující na soubor obsahující vystavené certifikáty. **Zamyslete se k čemu byste tento soubor využili**.

Odevzdávané soubory

Zkontrolujte, zda máte všechny soubory které se budou odevzdávat:

- `protokol.md`
- `cv2-ssh.pcap`
- `cv2-http.pcap`
- `cv2-tls.pcap`
- `cv2-https.pcap`

4 Ukončení cvičení

Do WIS-u odevzdejte vyplněný `protokol.md` a všechny zachycené `pcap` soubory. Vypňete virtuální stroje a obnovte snapshot ze začátku.