Week-09 SSH + Crypto - 1

Læringsmål

Usecase kryptering:

- sikker udveksling af data mellem to servere

Explain conceptually all the following terms, and how/why they are needed for SSH and TLS/SSL

- **Symmetric Encryption**
  Symmetrisk kryptering er en type kryptering, hvor en nøgle kan bruges til at kryptere meddelelser til den modsatte part, og også til at dekryptere de meddelelser, der er modtaget fra den anden deltager. Dette betyder, at enhver, der har nøglen, kan kryptere og dekryptere meddelelser til enhver anden, der har nøglen.

  Et problem med dette er at begge skal have den samme nøgle, og hvordan får man leveret en nøgle sikkert til den anden.

  Er hurtig og effektiv. Een nøgle.

  Symmetriske nøgler bruges af SSH til at kryptere hele forbindelsen.

  Offentlige / private asymmetriske nøglepar, der kan oprettes, bruges kun til godkendelse, ikke til kryptering af forbindelsen

  Der er forskellige symmetriske chiffersystemer, herunder AES, Blowfish, 3DES, CAST128 og Arcfour

- **Asymmetric Encryption**
  Offentlige / private asymmetriske nøglepar, bruges kun til godkendelse af en SSH klient

  Er ikke så hurtig og effektiv til symmetrisk. To nøgler

  Asymmetrisk kryptering adskiller sig fra symmetrisk kryptering, idet der kræves to tilknyttede nøgler for at sende data i en enkelt retning. En af disse nøgler kaldes den private nøgle, mens den anden kaldes den offentlige nøgle.

  Den offentlige nøgle kan deles frit med enhver part. Den er knyttet til dens parrede nøgle, men den private nøgle kan ikke udledes fra den offentlige nøgle. Det matematiske forhold mellem den offentlige nøgle og den private nøgle tillader den offentlige nøgle at kryptere meddelelser, der kun kan dekrypteres af den private nøgle. Dette er en envejsevne, hvilket betyder, at den offentlige nøgle ikke har nogen evne til at dekryptere de meddelelser, den skriver, og heller ikke kan den dekryptere noget, den private nøgle måtte sende den.

  Den private nøgle skal holdes helt hemmelig og bør aldrig deles med en anden part. Dette er et nøglekrav for, at det offentlige nøgelparadigme kan fungere.
  Gemmes i `~/.ssh/authorized_keys` filen i brugerens hjemmefolder på serveren. Serveren kan bruge den offentlige nøgle i denne fil til at kryptere en udfordringsmeddelelse til klienten. Hvis klienten kan bevise, at den var i stand til at dekryptere denne meddelelse, har den vist, at den ejer den tilknyttede private nøgle. Serveren kan derefter indstille miljøet til klienten.

- **Hashing**
  Kryptografiske hashfunktioner er metoder til at skabe en kortfattet “signatur” eller resume af et sæt information. Deres vigtigste kendetegn er, at de aldrig er beregnet til at blive vendt, de er næsten umulige at påvirke forudsigeligt, og de er praktisk talt unikke.

  Fingeraftryk af data. Man kan kun gå den ene vej. Det er IKKE kryptering. Hver hashing skulle gerne være unik

Hashes bruges hovedsageligt til dataintegritetsformål og til at verificere ægtheden af kommunikation.

HMAC, eller **h**ash-based **m**essage **a**uthentication **c**odes

Det anbefales generelt at kryptere data først og derefter beregne MAC.

Explain what it takes to safely log in to an SSH server, without having to provide a password

- klient ssh'er ind på server.
- server sender sin public key og der bliver spurgt om klienten vil acceptere denne. Man skal skrive yes fuldt ud for at komme videre. Her kunne man ringe til admin for at høre om public key er rigtig.
- klienten laver så et key-pair. Public key placeres i `~/.ssh/authorized_keys` filen i brugerens hjemmefolder på serveren.

Explain the term SSH-tunnel, and provide a practical example for its use
Et virtuelt rør, som beskytter data mod at udefrakommende kan se ind

Explain conceptually the purpose of Symmetrical Encryption, Asymmetrical Encryption and hashing for an SSH-connection

En SSH-session er etableret i to separate trin.

1. (symmetrisk) Den første er at blive enige om og etablere kryptering for at beskytte fremtidig kommunikation.
2. (assymetrisk) Den anden fase er at autentificere brugeren og opdage, om der skal gives adgang til serveren.**

   Derefter overføres data symmetrisk enkrypteret og hashed så der kan checkes når det når frem om der er pillet ved data.

SSL/TSL - SSH:

- **SSL er til sikring af internetforbindelser mellem websteder og deres besøgende**
- **SSH er til at køre kommandoer via fjernadgang.**

Explain the steps you have to go through to set up a server with MySQL, as secure as possible →

- How can we limit the client IP's that can connect
  - **Når man laver en bruger kan man vælge hvilke IP'er denne bruger kan connecte fra (og at denne skal bruge SSL)**
  - **man kan sætte serverens firewall til kun at kun en hvis IP kan forbinde til port 3306 f.eks.**
- If set up to allow only localhost and a firewall that deny 3306, can we still connect "safely" from a remote server
  Man kan SSH tunnel sig ind bag firewall'en på serveren og forbinde til databasen derfra
- how to set up an SSL connection that anyone can use
  lav en bruger der skal benytte SSL.
  `mysql -u root -p -h 127.0.0.1`

  ```
  mysql> CREATE USER 'everywhere'@'%' identified by 'test' REQUIRE SSL;
  ```

  mysql> GRANT ALL ON example.* TO 'everywhere'@'%';
  mysql> FLUSH PRIVILEGES;

  Sæt MYSQL op til at kunne forbinde til alle, men at de skal bruge en sikker forbindelse, SSL:

  ```
  **`sudo nano /etc/mysql/my.cnf`**
  [mysqld]
  require_secure_transport = ON
  ```

```
bind-address = 0.0.0.0
```

- Demonstrate a client application (Java or whatever you prefer) running on a separate server that access the Database using SSL
  - java
    persistence.xml
    java class
- how to set up an SSL connection that requires clients to identify themselves with a certificate.
  Kunne ikke få nedenstående til at virke, men det var det eneste jeg kunne finde om certifiakter

  `sudo nano /etc/mysql/my.cnf`
  [client]
  ssl-ca=/var/lib/mysql/ca.pem
  ssl-cert=/var/lib/mysql/server-cert.pem
  ssl-key=/var/lib/mysql/server-key.pem

You decide the order in which you want to present these topics since you probably won't have time to cover all the individual "pins"

Week-13 Crypto - 2

Explain conceptually all the following terms, and how/why they are needed for SSH and TLS/SSL

- Symmetric Encryption
- Asymmetric Encryption
- Hashing and MAC (Message Authentication Code)
- Cipher Suites
- Diffie–Hellman key exchange
- Digital Signatures
- Certificates
- Certificate Authorities and Certificate Trust Hierarchies

Feel free to introduce many of the terms using a saved Wireshark capture of a TLS-handshake

You decide the order in which you want to present these two topics since you probably won't have time to cover all the individual "pins"