Week-13 Crypto - 2

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

* **Symmetric Encryption**

En algoritme hvor en enkelt nøgle bruges til at kryptere og dekryptere information. Entiteterne som kommunikerer med hinanden skal begge kende til nøglen, hvilket er denne type krypterings ulempe.  
Algoritmen konverterer data’en til en form, som ikke kan “forstås” med mindre man har nøglen til at dekryptere det. Når modtageren får beskeden kører algoritmen bagud så beskeden returneres i sin oprindelige læsbare form.  
Der er to typer af symmetrisk krypterings algoritmer:  
Block algorithms og Stream algorithms.

* **Asymmetric Encryption**

En algoritme hvor der i modsætning til symmetrisk kryptering er to nøgler i stedet for en enkelt. Den ene nøgle bruges til krypteringen og den anden til dekrypteringen, derfor er disse nøgler et par, da de kun virker over for deres modstående part.  
Dog er det muligt for nøgle A at kryptere en besked til nøgle B men også at dekryptere beskeder fra B.  
Ofte kaldes denne type kryptering også for Public Key Cryptography, fordi der laves en public key og en private key. I mange tilfælde deles public key ud til bestemte personer, som så har deres egen private key.

* **Hashing and        MAC (Message Authentication Code)**

Hashing er en metode/funktion, som mapper en arbitrær størrelse til en bit-streng(alfanumerisk) af en absolut størrelse. Det er en en-vejs funktion, så den kan ikke “invertes”.   
MAC en metode/funktion, som har til formål at sikre dataintegriteten og autenticiteten af beskeden, som er sendt afsted. Den finder ud af om beskeden kommer fra den rigtige sender/server og at denne ikke er blevet ændret.

* **Digital Signatures**

Et scheme der verificerer autenticiteten af beskeder eller dokumenter. Signaturer er et standard element i krypterings protokoller og bruges ofte i software distribuering, finansielle transaktioner, kontrakter osv.  
En sådan digital signatur bruges også her i Danmark, hvor vi benytter os af NemID.

* **Certificates**

Et digitalt certifikat fungerer meget som et fysisk certificat. Det er information/data med en tilknyttet public key, som hjælper andre med at verificere, at nøglen er gyldig.   
Certifikatet består af tre elementer, en public key, information om klienten (såsom navn, bruger ID, kodeord) samt en eller flere digitale signaturer.  
Det er i bund og grund en public key med en eller flere former for ID tilknyttet med et stempel af fra en troværdig tredjepart.

* Certificate Authorities and Certificate Trust Hierarchies
* Cipher Suites
* Diffie–Hellman key exchange

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

Explain, and preferably demonstrate,  the consequence of using a self-issued certificate