Datasäkerhet

2D1522 Datorteknik och -kommunikation

2D2051 Databasteknik och datorkommunikation

http://www.nada.kth.se/kurser/kth/2D1522/

http://www.nada.kth.se/kurser/kth/2D2051/

Dagens föreläsning

- Syfte
  - Ge er kunskap nog att förstå de största riskerna vad gäller datasäkerhet
- Mål
  - Förstå kryptografins grunder
  - Kunna RSA-kryptografi
  - Förstå principen bakom brandväggar
  - Känna till de största problemen gällande drift av datorers/system
  - Känna till de viktigaste organisatoriska frågor som kan orsaka problem om de inte är uppmärksammade

Kryptografi

- Kryptografi har som mål att göra information oläslig för obehöriga.
- Termer/förkortningar som används senare
  - M = Meddelandet i klartext (läsligt för alla)
  - C = Det krypterade meddelandet
  - f(M) = Krypteringsfunktion dvs C = f(M)
  - f'(C) = Dekrypteringsfunktion dvs M=f'(C)
  - Alltså: M=f'(f(M))

Tidig historik

- 1900 f.kr. Egyptisk skrift med icke-standard-hieroglyfer. Kan sägas vara det första kryptot.
- 487 f.kr. Greker använder stav med viss diameter som ett läderband lindas runt. Sedan skrivs meddelandet ”rullen”, varpå bandet sedan sänds iväg. För att dekryptera krävs stav med samma diameter.
- 50 f.kr. Caesarkrypto. Enkelt substitiutionskrypto

Substitutionskrypton

- Caesarkrypto bygger på enkel substitution av bokstäver. Alfabetet förskjuts helt enkelt ett antal tecken.

NYCKEL

```
M: ABCDEFGHIJKLMNOPQ...
C: CDEFGHIJKLMNOPQRS...
```

EXEMPEL

```
M: HEJ HOPP
C: JGH JQRR
```


Problem med substitutionskrypton

- Om man vet principen bakom Caesarkryptot räcker det att testa 29 varianter för att knäcka ett meddelande.
- Den något mindre triviala varianten är att inte förskjuta alfabetet utan att ”slumpmässigt” välja vilken bokstav en krypterad bokstav ska få

NYCKEL

```
M: ABCDEFGHIJKLMNOPQ...
C: PILKMUJYGNBVRFDCE...
```

EXEMPEL

```
M: HEJ HOPP
C: YMN YDCC
```


Fler problem

- Detta är fortfarande väldigt enkelt att knäcka, eftersom varje bokstav har en exakt motsvarighet. I exemplet blev strängen ”PP” krypterad till ”CC”. Eftersom det nästan bara är konsonanter som kan vara två av samma i följd kan man antaga att det krypterade tecknet ”C” motsvarar en konsonant.
- När man har lite större textmassa kan man enkelt göra statistiskt baserade gissningar. Detta eftersom freknvensen av bokstäver varierar. Exempelvis är bokstaven ”E” mycket vanligare än ”Q”.

EXEMPEL

```
M: HEJ HOPP
C: YMN YDCC
```


Moderna krypton

- Moderna krypton brukar vara uppbyggda kring en algoritm och en nyckel.
- Algoritmen (f) är själva tillvägagångssättet, vilket antas vara känt av alla. Svårbytt.
- Nyckeln (K) är hemlig och används tillsammans med algoritmen för att kryptera ett meddelande. Lätt att byta.
- C=f(M,K)

Symmetriska krypton

- En kryptoalgoritm är symmmetrisk om nyckeln som används för att kryptera ett meddelande är samma som används för att dekryptera meddelandet

```
C=f(M,K)
M=f'(C,K)
  dvs
M=f'(f(M,K),K)
```

- Caesarkrypto är i princip ett symmetriskt krypto.

```
F = "förflytta alfabetet K steg till vänster"
f' = "förflytta alfabetet K steg till höger"
K = 2
```


XOR

- XOR är en logisk operation som ofta används i kryptosammanhang.

A AND B			A OR B			NOT A		A XOR B		
A	B	RESULTAT	A	B	RESULTAT	A	RESULTAT	A	B	RESULTAT
0	0	0	0	0	0	0	1	0	0	0
0	1	0	0	1	1	1	0	0	1	1
1	0	0	1	0	1			1	0	1
1	1	1	1	1	1			1	1	0

Blankettchiffer

- XOR har egenskapen att (A XOR B) XOR B = A
- Detta gör att XOR kan användas som enkel algoritm för ett symmetriskt krypto, så kallat blankettchiffer.

```
Sändare
M:10110111011
K:01101101001
-------------
C:11011010010

Mottagare
C:11011010010
K:01101101001
-------------
M:10110111011
```


Blankettchiffer (forts)

- Blankettchiffer har den trevliga egenskapen att det är oknäckbart så länge som två regler följs:
  - Nyckel K är helt slumpmässigt vald (datorgenererade slumptal är inte äkta slump utan s.k. pseudoslump)
  - Nyckeln används en och endast en gång.
- Informellt "bevis"
  - Eftersom nyckeln K är helt slumpmässig kommer även den krypterade strömmen av ettor och nollor vara helt slumpmässig, även om meddelandet M inte alls är slumpmässig.

```
Sändare
M:11111111111
K:01101101001
-------------
C:10010010110
```

  - Om nyckeln används flera gånger kan två krypterade meddelanen C1 och C2 jämföras, och mönster hittas som underlättar forcering.

Blankettchiffer (forts)

- Blankettchiffer, då det endast används med en viss nyckel till att kryptera ett meddelande, kallas engångskrypto och är oknäckbart.
- Problem finns dock:
  - C är lika långt som M.
  - Hur får man i praktiken en ”äkta” slumptalsfrekvens?
  - Hur överförs nyckeln?
  - Om man vet M och har C kan man räkna ut K och därefter kryptera godtycklig annan text (”Anfall nu”, resp ”Anfall ej”).

Strömkrypto mot blockkrypto

- Blankettchiffer är ett s.k. strömkrypto (stream cipher), vilket betyder att man får in en ”ström” med ettor och nollor som krypteras en och en när de kommer in.
- En annan variant är blockkrypton (block cipher) vilka buntar ihop en mängd ettor och nollor av meddelandet M och krypterar hela blocket. Exempel på vanlig blocklängd är 64 bitar.
- Exempel på blockkrypton: DES, IDEA, Blowfish

Ett problem med blockkrypto

- Ett möjligt problem med blockkrypto är om man har en given nyckel som krypterar alla block. Om då två olika block innehåller samma text, dvs M1=M2, så blir C1=C2. Om man har kunskap om texten eller språket kan det förenkla forcering.
  - T.ex. denna presentation innehåller strängen ”krypto” tämligen ofta. 8 tecken á 8 bitar ger 64 bitar, dvs ett block. Om presentationen krypterades skulle denna sekvens förekomma ofta, så någon kan göra en kvalificerad gissning om att klartexten för det kodade blocket är just ”krypto”.
  - Om xor-funktionen används har man då C och M och kan därmed räkna ut K varpå allt annat också kan dekrypteras.
- Därför finns oftast en återkopplingsslinga så att nyckeln ändras mellan block n och block n+1.

Asymmetriska krypton

- Symmetriska krypton använder alltså samma nyckel för kryptering som för dekryptering, dvs M=f’(f(M,K),K)
- Asymmetriska krypton använder däremot olika nycklar för kryptering och för dekryptering.
  - `C = f(M,K)`
  - `M = f'(C,K')`
- Detta kan uttnyttjas för s.k. publika nycklar. En organisation kan då publicera en nyckel, K, som kan användas om någon vill skicka meddelanden till organisationen. Dekrypteringsnyckeln K’ hålls däremot hemlig.
- Nu kan vem som helst skicka meddelanden till organisationen som ingen annan än organisationen kan läsa.
- OBS! Inte ens den som krypterar meddelandet kan återskapa ursprungsmeddelandet M ur det krypterade meddelandet C.
- Exempel på asymmetriskt krypto: RSA, DSS

Envägskrypton

- Ibland finns ingen anledning att återskapa ursprungsmeddelandet, det kan t.o.m. vara önskvärt att *inte* kunna göra det.
- Ett exempel är lösenordshantering.
  - Användaren väljer ett lösenord M som krypteras till C
  - C lagras i en användardatabas.
  - När användaren senare försöker logga in krypteras åter igen M och resultatet blir åter igen C.
  - Detta C jämförs med användardatabasen. Eftersom de matchar ska inloggning tillåtas.
- Fördel: Om någon kommer över lösenordsdatabasen känner denne någon inte till originallösenordet, dvs det går fortfarande inte att logga in på samma datorer som har en annan krypteringsnyckel K, även om de använder samma lösenordsalgoritm.
- Kan även användas för digitala signaturer, dvs för att verifiera att en sändare är den denne anger. Fungerar som publik nyckelhantering fast tvärt om.

Digitala signaturer

- Digitala signaturer använder envägskrypton. De används till att verifiera att en sändare är den denne anger. Fungerar som publik nyckelhantering fast tvärt om.
  1. (M + tid + avsändare) envägskrypteras till en ”message digest”
  2. Denna krypteras med en privat nyckel -> ”digital signatur”
  3. Denna + meddelandet + tid + avsändare sänds till mottagaren
  4. Mottagaren: Dekryptera den digitala signaturen med en publik nyckel. Mottagaren måste känna till den publika nyckeln en avsändare har.
  5. Mottagaren utför steg 1 och ser om om resultatet blir samma som i steg 4.

RSA

- RSA (Rivest, Shamir och Adleman) är en algoritm för publik nyckelhantering som är vanlig idag.
- Grunder:
  - Relativa primtal: Ett tal är ”relativt prima” ett annat tal ifall inget av talen har några gemensamma primtalsfaktorer. 8 och 9 är relativt prima (2*2*2 resp 3*3), 8 och 10 är inte relativt prima (2*2*2 resp 2*5)
  - Modulära funktioner: P mod Q = resten vid heltalsdivision.
    T.ex. 23 mod 4 = ((5*4) + 3) mod 4 = 3

RSA : Howto

- Välj två stora primtal, p och q. Räkna ut n = p * q
- Hitta ett tal e som är relativt prima (p-1)*(q-1)
- Hitta ett tal d som uppfyller d*e mod (p-1)*(q-1) = 1

- Kryptering: $C = M^e \bmod n$
- Dekryptering: $M = C^d \bmod n$

RSA : Exempel

- Välj två stora primtal, p och q. Räkna ut n = p * q
  - 33 = 3 * 11 -> p=3, q=11, n=33 (givetvis mycket större tal i verkligheten)
- Hitta ett tal e som är relativt prima (p-1)*(q-1)
  - 7 är relativt prima 2*10 -> e=7 (här är 7 ett primtal, men det är alltså inte nödvändigt)
- Hitta ett tal d som uppfyller d*e mod (p-1)*(q-1) = 1
  - 3*7 = 21 = 1*20 + 1 -> 3*7mod 20 = 1 -> d = 3
- Kryptering: $C = M^e \bmod n$
  - Om M=2 så: $C = 2^7 = 128 = 3*33 + 29$ -> $C = 2^7 \bmod 33 = 29$
- Dekryptering: $M = C^d \bmod n$
  - Om C=29 så $29^3 = 24389 = 739*33 + 2$ -> $M = 29^3 \bmod 33 = (739*33 + 2) \bmod 33 = 2$

Hur säkra är krypton

- För de flesta krypton (utom engångskrypton) kan man försöka göra en ”brute force-attack” dvs testa alla möjliga kombinationer av nycklar.
- Om nyckel på webben är t.ex. 40 bitar -> 2^{40} kombinationer dvs 1.099.511.627.776 kombinationer. Går att knäcka på någon dag med brute force på en modern dator.
- Med 128 bitar blir det 340.282.366.920.938.463.463.374.607.431.768.211.456 kombinationer, vilket inte kan knäckas med alla datorer i världen under universums livslängd.

Hur säkra är krypton (forts)

- Det finns dock många andra ”smartare” sätt än brute force.
  - Buggar i krypteringsprogram
  - Mäta strömkonsumtion av smarta kort
  - Bättre algoritmer
  - Kvantdatorer
  - dock finns kvantkrypton (system finns nu till salu)
  - samt framför allt att få/inga kända krypteringsalgoritmer har bevisats vara säkra mot effektivare metoder än brute force-attacker.

Nu till andra datasäkerhetsapekter

- Nu till ett mycket bredare perspektiv.
- Oftast är inte intrångsförsök de största problemen vad gäller data.
- I ett bredare perspektiv betraktar vi också mer organisatoriska problem såsom backupptagning etc.

Brandväggar

- Brandväggar används för att förhindra oönskad trafik in eller ut från ett nätverk/dator
- Baseras oftast på sändarens IP-adress eller domännamn
- Styr vilka portar som är åtkomliga för vem
- Regler kan vara av typen ”Tillåt alla inkommande anslutningar från datorer på nätverket X till tjänsten Y på datorn Z i det lokala nätverket”.
- Bra att ha även hemma, buggar i operativsystem upptäcks ständigt och (handen på hjärtat) hur ofta brukar man ladda ner säkerhetsuppdateringar mm.
- Ca 5-10 attacker/dag hemma för mig som har Chello
- Bra gratisprogram för PC är Zone Alarm, http://www.zonelabs.com/

Okrypterade lösenord

- Många system skickar/har skickat lösenord i klartext.
- Lätt att fånga upp med en ”sniffer”, speciellt om man sitter på det lokala nätverket.
- Om samma lösenord används på många platser eller för många tjänster kan det leda till allvarliga säkerhetshål.
  - Exempel på protokoll som är osäkra respektive deras säkra motsvarighet.

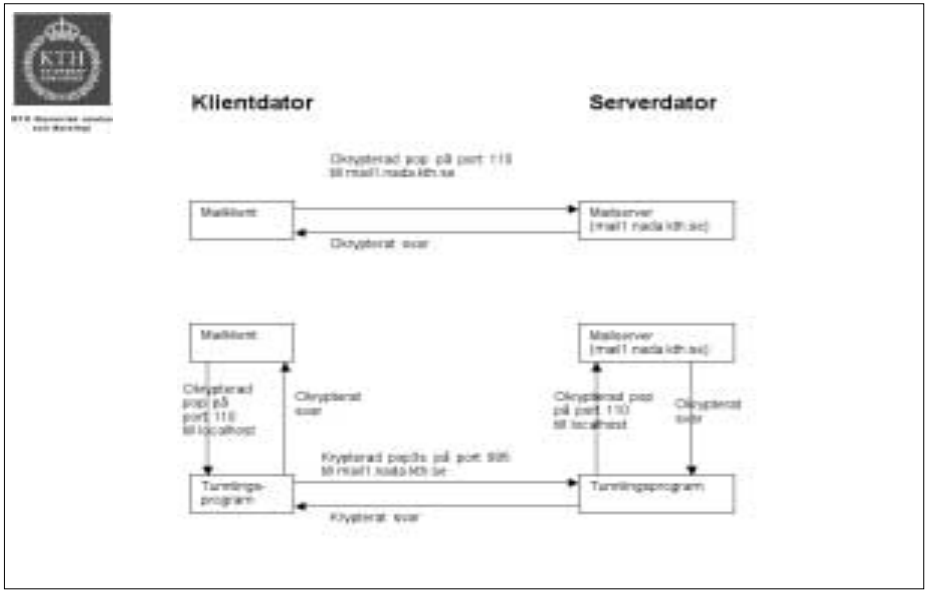
telnet (port 23)	ssh (port 22)
pop3 (port 110)	pop3s (port 995)
smtp (port 25)	ssmtp (port 465)

Val av lösenord

- Ord som står i ordlistor är DÅLIGA. Speciellt dåligt är namn på hund, bil och barn. Tar några sekunder att knäcka med standardknäckarprogram.
- Se till att ha olika lösenord, så att förslusten av ett lösenord inte innebär att alla tjänster du använder blir vidöppna.
- Ta gärna en mening man kommer ihåg, tag begynnelsebokstäverna i varje ord.
  - Exempel: ”Nu vill jag gå hem och sova”
    Nvjghos
- För att öka säkerheten ytterligare bör man blanda gemener och versaler samt använda några siffror.

Tunnling av osäker kommunikation

- Om man har en osäker (okrypterad) förbindelse mellan två datorer kan man välja att ”tunnla” det protokoll som används via en säker förbindelse, t.ex. med SSL (Secure Socket Layer).
- Det går ofta bra att att använda gamla server- och klientprogram utan någon som helst ändring, annat än att man lägger till tunnlingsprogramvaran.
- Därefter går inte kommunikationen direkt mellan klientprogram och serverprogram, utan all kommunikation går via tunnlingsprogramvaran.



Attacker

- Vanliga säkerhetshål
  - Sårbara egenutvecklade webbscript
  - Sårbara webbservrar, t.ex. webmailtjänster
  - Dåliga lösenord
  - Radionätverk
  - Terminalservers
- Väl inne
  - Ladda dit egna verktyg för att kunna komma ”djupare in” i nätet.
  - Hämta viktig information (t.ex. lösnordsfilen via att skicka ett email)
  - Scanna nätverket efter lösenord mm
  - Hacka sig vidare in på företaget/datorn eller vidare till andra företag

Nu till ”det icketekniska”. Vad vill man skydda sig mot?

- Skydda mot intrång/attacker
- Säkerställa att data inte förloras genom attacker/slarv/stöld/brand
- Säkerställa support av viktiga program
- Säkerställa drift
- Säkerställa kompetens när folk slutar
- Organisatoriska frågor i allmänhet

Programvaror

- Underhållsavtal
  - Se till att ha ordentliga underhållsavtal för viktiga system.
- Utveckling och inköp
  - Bli inte beroende av programvara som kan bli ”osupportad”. Se till att skaffa underhållsavtal.
- Ändringshantering.
  - Ha rutiner för att uppgradera program organiserat
  - (En genomsnittlig systemadministratör får i genomsnitt 1362 st programvaruuppdateringar per år)

Drift

- Fysiskt skydd
  - Brand, inbrott
- Distribution av data
  - Kryptera viktig information. Använd eventuellt andra distributionskanaler än elektroniska för extra känslig information.
- Tillträde
  - Vem har tillträde vart och när
- Säkerhetsskåp
  - Lagra viktig information såsom avtal, backupper, rootlösenord osv i ett säkerhetsskåp.
  - Olika klassningar finns som exempelvis klarar brand olika lång tid.

Drift (forts)

- Stationära datorer
  - I möjligaste mån begränsa fysisk åtkomst
  - Inga datorer utan lösenord (som t.ex. MacOS 9 eller Windows före Windows 2000)
  - Eventuellt kryptera hårddisken
- Bärbara datorer
  - Om möjligt kryptera hårddisk samt använd lösenord eftersom det är lättare att bärbara datorer kommer bort.
- Behörigheter och systemövervakning
  - Logga misslyckade inloggningsförsök (samt kolla loggen)
  - Logga andra misstänkta aktiviteter (t.ex. inloggningar på udda tider)
  - Individer bör ha behörighet endast till de tjänster och dokument de behöver.

Drift (forts 2)

- Virusskydd
  - Se till att ha ett virusskydd, eventuellt även på mailservern som går igenom alla inkommande bilagor.
  - Se till att hålla virusskyddet uppdaterat.
- Backupp
  - Se till att ha backuprutiner
  - Se till att backuperna går att återställa
  - Ha rutiner för hur backuperna förvaras, då det är mycket illa om en backup blir stulen.

Drift (forts 3)

- Internet
  - Stäng av onödiga tjänster som inte används
  - Använd brandväggar
- Epost
  - Kryptera känsliga mail
  - Använd digitala signaturer för att verifiera avsändare och äkthet på viktiga mail
  - Kräv eventuellt mottagningsbevis
- Stora uppdateringar
  - Om möjligt testa större ändringar i en testmiljö först.

Trådlöst

- Ofta går modemuppkopplingar och trådlösa lan förbi brandväggar och liknande, och är därför stora säkerhetsrisker
- Trådlösa nätverk (WLAN etc) kan (enligt en indelning) delas in i fyra kategorier
  - Ingen säkerhet
    - Vem som helst kan logga in
  - Användaren måste känna till nätverkets namn
    - Det finns dock kort som ”sniffar” på alla möjliga frekvenser och då enkelt kan komma in
  - Nätverksnamn + lösenord
    - Lätt att med samma teknik som ovan hitta nätverksnamnet. Sedan lyssnar man efter inloggningar och kan då snappa upp lösenord mm.
  - Endast förregistrerade MAC-adresser tillåts + ovanstående
    - Det går att snappa upp vilka MAC-adresser som tillåts komma in, och sedan kan man ”fejka” en av dessa

Organisatoriska frågor

- IT-säkerhetspolicy
  - Ha en IT-säkerhetspolicy
  - Se till att personalen känner till den
  - Se till att ha ansvariga för IT-säkerheten (inkl brandskydd, backupp mm)
- Lagar
  - Se till vilken lagstiftning som gäller, t.ex. gällande PUL och upphovsrätt
- Risk- och sårbarhetsanalys
  - Gör en risk- och sårbarhetsanalys.
  - Dataföreningen i Sverige har en standardmetod och ett standardverktyg ”SBA Scenario”

Organisatoriska frågor (forts)

- Dokumentförstörare
  - Känsliga dokument ska inte kastas i papperskorg/papperså tervinning
- Tillämpningsägare och systemägare
  - Ha personer ansvariga för alla tillämpningar och alla system.
- Användarsupport
  - Dokumentera uppkomna problem samt dess åtgärder
  - Dokumentera rutiner
  - Detta leder till mindre mängd dubbelarbete samt minskar beroendet av nyckelpersoner
- Kontinuitetsplan i händelse av avbrott/katastrof
  - Elavbrott, brand, stöld...