Sikkerhet i Blåtann

Hovedoppgave
ved
sivilingeniørutdanning i
informasjons- og kommunikasjonsteknologi

av
Stian Hartviksen

Grimstad, mai 2000

Sammendrag

Denne diplomoppgaven tar for seg sikkerhetsaspekter til trådløs kommunikasjon ved bruk av Blåtann. Sikkerhet består av tre hoveddeler; autentisering, autorisering og kryptering.

Autentisering er det å bekrefte ektheten til personer, organisasjoner, maskinvare, filer, meldinger og dokumenter. Blåtann støtter denne formen for sikkerhet på enhetsnivå, men ikke på tjenestenivå. Autentisering kan i tillegg implementeres i applikasjoner.

Autorisering er å sjekke om man har tilgang til en tjeneste. Autorisering inkluderer alltid autentisering. Blåtann støtter kun autorisering for alle tjenester, ikke hver enkelt. For mer spesifikk tilgangssjekking til hver tjeneste må det implementeres en sikkerhetsmodul.

Kryptering sørger for å gjøre informasjonen uforståelig for andre enn dem som har de riktige nøklene. Sikkerheten til ulike algoritmer vurderes ofte ut fra tiden det vil ta å dekryptere en tekst kryptert med algoritmen. Blåtann støtter kryptering på forbindelse og applikasjonsnivå.

Oppgaven tar for seg hvordan disse tre delene er støttet, og gir en generell innføring i hva datasikkerhet er.

Etter en vurdering av de forskjellige sikkerhetsaspektene ble det valgt å fordype autentiseringsrutinene til Blåtann. Autentiseringsrutinene støtter ikke forskjellig tilgang til forskjellige tjenester i samme forbindelse, da denne kun går på enheter og ikke tjenester. På bakgrunn av dette er det lagt frem en mulig løsning for å bedre tjenestekvaliteten ved autentisering. Denne løsningen baserer seg på å innføre en sikkerhetsorganisator som de forskjellige lagene i Blåtannstakken skal sjekke informasjon mot.

Forord

Diplomoppgaven ”Sikkerhet i Blåtann” tar for seg sikkerhetsaspekter til Blåtann på protokollnivå. Eventuelle sikkerhetsmekanismer i applikasjonslaget til Blåtannprotokollen er utenfor omfanget til denne oppgaven.

Diplomoppgaven er skrevet som ett ledd i sivilingeniørutdanningen ved Høgskolen i Agder i samarbeid med Ericsson AS i Grimstad. Arbeidet har pågått i tidsrommet fra januar til og med mai 2000. Jonny Ervik har vært veileder fra Ericsson AS og Magne Arild Haglund har vært veileder fra Høgskolen.

Takk til Rune Knutsen på Ericsson AS, Vladimir Oleshchuk foreleser ved HiA, veiledere og Blåtannforumet ved HiA.

Grimstad, 29.mai 2000

Stian Hartviksen

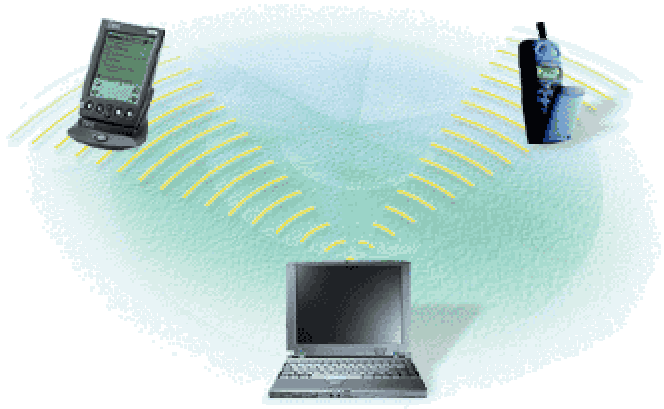
Innholdsfortegnelse :

Figuroversikt

Tabelloversikt

1 Innledning

Trådløs kommunikasjon er blitt ett attraktivt marked, ikke bare for telefoni men også for andre aktører. Blåtann er en ny trådløs kommunikasjonsstandard som kom i mai 1998 og bruker radioforbindelse til å overføre data over korte avstander. Dette gjør at man kan få en enkel kommunikasjon mellom mobiltelefoner, bærbare PCer, PDA (Håndholdte PCer) og annet periferi utstyr.



Figur 1 - Eksempel på bruk av trådløs kommunikasjon

Blåtann SIG (Special Interest Group) er ett resultat av ett samarbeid mellom de fem grunnleggerne som utviklet konseptet til en standard. Denne standarden er en konkurrent med 802.11 [9] som er en spesifikasjon for trådløst datanettverk. De fem grunnleggerne er Ericsson, Toshiba, IBM, Nokia og Intel. Blåtann SIG består av 1882 medlemmer [10] per 25.mai 2000.

Det er forventet at Blåtannprodukter som handsfree [13], kabelerstatning til PC [14] og lokalnettverkløsninger [15] vil være i handelen fra høsten 2000.

1.1 Oppgavedefinisjon og spesialisering :

Oppgaven gikk ut på å studere i hvilke grad Blåtann behandler dataene med hensyn på sikkerhet. Hvordan enhetene autentiseres med hverandre, og hvordan sikkerhetsalgoritmene fungerer samt å se på muligheter som skal til for å gjøre Blåtann til en sikker kommunikasjonskanal, med hensyn på riktig overføring, at ikke dataene kan avlyttes og at ikke dataene kan forandres. Ett av de overstående tema skulle fordypes.

Etter en vurdering av teorien ble det valgt å fordype hvordan enhetene autentiseres med hverandre. Svakheter i autentiseringsrutinene ble funnet og prøvd forbedret. Ett utkast til løsning ble utviklet, og ved å sammenligne denne løsningen med Thomas Muller sitt whitepaper om sikkerhet [4] ble drøftingen gjennomført og konklusjonen skrevet.

1.2 Metode

Oppgaven har i hovedsak vært ett litteraturstudie. Januar ble brukt til å finne ut generelt om teknologien og definering av oppgaveteksten. februar, mars og april gikk med til å søke etter informasjon om sikkerhet generelt og definere hva sikkerhet er. Fra mars begynte skriveprosessen og informasjon om sikkerhetsaspektet ble funnet. April og mai gikk med til å sortere informasjon og se på en ny sikkerhetsmodul som ikke er en del av standarden pr 29. Mai 2000.

1.3 Møter

Det ble opprettet ett Blåtannforum på HiA som hadde møter hver tirsdag. På disse møtene ble det tatt opp fremdrift fra de forskjellige aktørene, problemstillinger ble drøftet, og veiledning ble gitt. Medlemmene var ansatte fra Ericsson (Jonny Ervik og Tone Strømseng), foreleser fra HiA (Magne Arild Haglund), sivilingeniørstudenter (Tom Farbrot, Per Rune Grønhovd og Stian Hartviksen) og telekommunikasjons studenter (Kjell Rune Øyrås, Anders Pedersen og Endre Laugerud). Studentene fra HiA hadde forskjellig oppgaver innen for området Blåtann, og brukte hverandres kompetanse til å utføre sine respektive oppgaver.

2 Teoribakgrunn for oppgaven

Datasikkerhet som denne oppgaven tar for seg består i hovedsak av tre deler. Disse er autentisering, autorisering og kryptering [11]. I dette kapitlet blir det gitt en rask innføring i hva sikkerhet er, hvorfor man må ha sikkerhet og litt om kryptering [12]. I tillegg til sikkerhet, inneholder kapittelet en kort innføring i Blåtann.

2.1 Hva er sikkerhet ?

Bruken av datamaskiner, og dermed sårbarheten til dataene som behandles, har endret seg dramatisk de siste 30 årene. På 1970 tallet var det vanlig å benytte sentrale datamaskiner. Datasikkerheten var først og fremst truet av egne ansatte, fysiske innbrudd og systemfeil. Truslene mot dataene kunne begrenses ved å låse inn maskinene i skjermede rom med streng adgangskontroll. Trusselbildet var med andre ord ganske oversiktlig.

I dag er de fleste knyttet opp mot nettverk av ulike typer (LAN, WANs), mange også mot lands- eller verdensomfattende nettverk (UNINETT, Internet). Dette gir et helt annet, og adskillig mer uoversiktlig trusselbilde. Truslene kommer ikke lenger bare fra egne ansatte, innbruddstyver o.l., men potensielt fra hvem som helst som på en eller annen måte kan koble seg opp mot, eller avlytte nettverkene en er tilkoblet.

Det er med andre ord ikke lenger nok å låse inn datamaskiner for å hindre uvedkommende tilgang til data. En må i steden sørge for at dataene sikres i alle ledd i behandlingsprosessen; fra innsamling av rådata til distribusjon av ferdigbehandlede data. Totalsikkerhet er blitt et sentralt begrep.

2.2 Hvorfor er datasikkerhet viktig?

Det er mange grunner til at en ønsker å beskytte informasjonen som behandles på datamaskiner. Noen er åpenbare, mens andre er mer diffuse og dermed vanskeligere å forholde seg til.
Noen av de mest åpenbare grunnene til å ville beskytte informasjon er:

- **Personvern** - hindre at sensitiv personinformasjon blir misbrukt. Fødselsnumre, karakterutskrifter og legejournaler er eksempler på informasjon som faller inn under denne kategorien. Også post/adresselister kan inneholde sensitiv personinformasjon.
- **Forhindre industrispionasje** - forhindre at informasjon/data av verdi for en bedrift eller organisasjon kommer på avveie. Mange bedrifter har svært høye utviklingskostnader. Formler, rådata og beskrivelser av egenutviklet maskin og programvare er eksempler på informasjon som kan være viktig for en bedrift. Informasjon som kommer på avveie kan for eksempel gi konkurrenter konkurransefortrinn ved at de unngår utviklingskostnader. Men den kan også brukes til utpressing, svindel, i tradisjonell spionasje og sikkert misbrukes på mange andre måter. Informasjonen, dataene eller programmene som må sikres trenger heller ikke å ha verdi for andre enn bedriften som eier dem for å være av interesse for tjuver. Det finnes tilfeller der lagringsmedier (harddisker, disketter o.l.) har blitt stjålet og brukt til utpressing.

- **Unngå hacking** - stoppe uvedkommende i å få tilgang til maskiner eller andre ressurser slik at data eller informasjon blir blottstilt. Grovt sett har en to typer hackere, de "vennlige" som bryter seg inn for å se om de klarer det, og de destruktive som ofte bryter seg inn for å ødelegge. Det er ønskelig å nekte begge typer tilgang til sensitive data.
- **Verne informasjon som er belagt taushetsplikt** - Forhindre at informasjon f.eks. av viktighet for "rikets sikkerhet" faller i hendene på reelle eller imaginere fiender.

Noen mindre åpenbare grunner til å ville beskytte informasjon er ønsket om å verne om ytringsfriheten og ønsket om å beskytte retten til å ha et privatliv.

2.3 Autentisering

Autentisering er det å bekrefte ektheten til alt fra personer, organisasjoner og maskinvare til filer, meldinger og dokumenter. Autensitet er sentralt i all kommunikasjon og datastøttet samarbeid. Det danner også grunnlaget for mange andre tjenester som er mulig via krypteringsteknologi. Dataintegritet og adgangskontroll er eksempler på krypteringstjenester som er avhengige av sikker autentisering, digitale signaturer og ulike sertifikater eksempler på hvordan autentisering kan implementeres.

Å skaffe seg uautorisert tilgang, brudd på konfidensialitet, er enklere enn man skulle tro. Datautstyr "lekker" informasjon via f.eks. elektromagnetisk stråling. Ved hjelp av relativt rimelig utstyr (et modifisert TV-apparat og retningsantenne) kan man forholdsvis lett kopiere skjermbildet fra en PC som sitter bak en tykk vegg et stykke unna.

2.4 Autorisering

Autorisering er å sjekke om enhet x skal ha tilgang til tjeneste y. Dette er konseptet hvor tillit eksisterer. Enheter som har tillit får tilgang til å aksessere tjenester, mens de enhetene som er ukjente eller ikke har tillit må autentisere seg for å få tilgang. Autorisering inkluderer alltid autentisering.

2.5 Kryptering

Kryptering er konfidensialitet. Krypteringsteknologi sørger for at fortrolig informasjon ikke kommer på avveie.

Å kryptere en tekst vil si å gjøre innholdet uforståelig(uleselig) for andre. Dekryptering er den motsatte prosessen. Ofte brukes betegnelsene chiffrering og dechiffrering om henholdsvis kryptering og dekryptering. Klarteksten er teksten før kryptering, chifferteksten er resultatet av krypteringen.

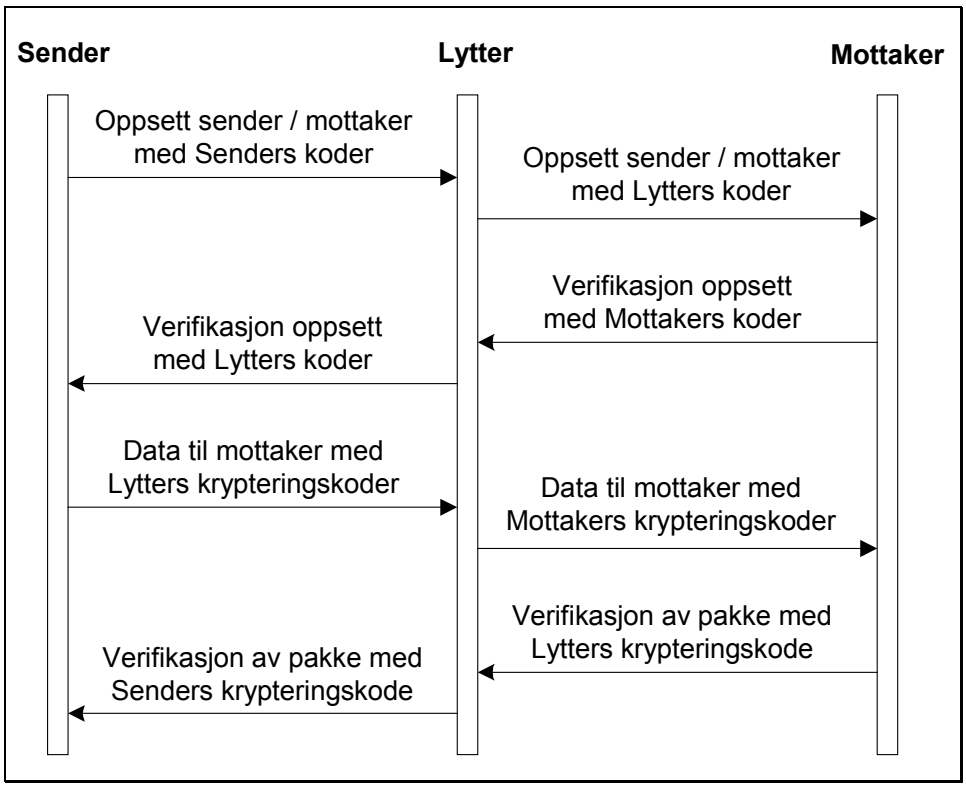
Sikkerheten til ulike algoritmer vurderes ofte ut fra tiden det vil ta å dekryptere en tekst kryptert med algoritmen.

Sikkerheten til nøkkelbaserte algoritmer er i tillegg avhengig av nøklene, spesielt av nøkkellengden. En sikker algoritme (som RSA algoritmen) trenger ikke være sikker dersom en benytter en for kort nøkkel. Også databeskaffenheten påvirker sikkerheten,

bruk av for eksempel faste felter og header i klarteksten forenkler dekrypteringen betraktelig. Ofte ser en eksempler på at leverandører av programvare reklamerer med at de har muligheter for kryptering. Dette kan lett gi en falsk følelse av trygghet da krypteringen ofte er svært enkel (som f.eks. i noen Windows applikasjoner som benytter XOR på trivielle måter!), og der det reklameres med velkjente algoritmer er ofte nøklene for korte. En bør også unngå faste felter i beskjeder etc, helst bør klarteksten komprimeres før kryptering for å minske redundansen (jo mer redundans i klarteksten, jo lettere er det å dekryptere chifferteksten.).

Maskinvareimplementasjoner er i utgangspunktet raskere og sikrere enn programvareimplementasjoner. Symmetriske algoritmer er som regel raskere enn offentlige nøkkelalgoritmer, DES er for eksempel raskere enn RSA. Jo lengre nøkler som benyttes, jo lenger tid tar krypteringen. Det er derfor viktig å foreta en avveining mellom hvor kraftig kryptering en trenger, hvor mye tid en har til rådighet og hvilken datakapasitet man har. En bør også ta hensyn til hvor lenge en kryptert tekst skal kunne holdes hemmelig. Jo flere år en vil holde noe hemmelig, jo lengre nøkler/sikrere algoritmer bør en velge. Etter å ha foretatt grunnleggende valg av algoritmer, generert nøkler etc, bør en benytte applikasjoner som skjuler den underliggende krypteringsteknologien mest mulig for brukerne.

En svært vanlig måte å misbruke nøkler på er å "snike seg inn" mellom sender og mottaker i en dialog ved å bytte ut de offentlige nøklene til en eller begge parter med ens egne. En kan da tjuvlytte ved å motta det som senderen sender (kryptert med ens egen falske nøkkel), dekryptere meldingen og kryptere den igjen med mottagers offentlige nøkkel og vice versa (samtidig som en beholder en kopi av klarteksten selv).



Figur 2 - En uatorisert tredjepart bytter ut nøklene til sender og mottaker og lytter på data.

2.6 Krypteringsalgoritmer

En *krypteringsalgoritme* er en matematisk funksjon som brukes til kryptering/dekryptering.
De er matematiske funksjoner som kan benyttes til kryptering og dekryptering av data. Sikkerheten til slike algoritmer avhenger av flere ting, men den beste måten å finne ut om en algoritme er sikker, er å se hvor lenge den har motstått angrep. Jo lengre tid en algoritme motstår angrep, jo sikrere er den. En bør med andre ord unngå å benytte nye algoritmer, spesielt egenkonstruerte. Utviklingen av krypteringsalgoritmer er en vitenskap og krever års erfaring for å lykkes.

Det finnes mange krypteringsalgoritmer på markedet, og under er en oversikt over de to mest brukte.

RSA - oppkalt etter de som laget den; Rivest, Adleman og Shamir. Den mest populære offentlige nøkkelalgoritmen på markedet. Algoritmen er svært enkel å forstå, og nøkkellengden kan varieres. Jo lenger nøkkel, jo sikrere og tregere er den. Sikkerheten til algoritmen er basert på at det er vanskelig å faktorisere store tall, spesielt tall som er produkt av primtall. [7]

DES - Data Encryption Standard, ble utviklet av IBM (med hjelp av NSA) og vedtatt som offentlig standard i USA i 1976. DES er en svært god symmetrisk krypteringsalgoritme som opererer på 64 bits blokker med klartekst. Den ansees fortsatt som sikker nok for sikring av mange typer sensitiv informasjon snart 25 år etter at den ble utviklet! Blant annet har Datatilsynet vurdert algoritmen som sikker nok til å beskytte sensitiv personinformasjon. Nøkkellengden er 56 bit (egentlig 64, men hver 8. bit er en sjekkbit), og algoritmen er svært rask, ofte opptil 1000 ganger raskere enn for eksempel RSA. Det finnes også billige maskinvareimplementasjoner av algoritmen, noe som øker hastigheten ytterligere. Pr 29.05.2000 er det nok med 64 bit krypteringsnøkkel. Ved å bruke dobbel eller trippel DES vil man få en sikrere kryptering [7].

2.7 Hva er Blåtann ?

Blåtann er en høyhastighets trådløs kommunikasjons teknologi som er designet til å forbinde telefoner, PCer, PDAer og annet mobilt utstyr. I motsetning til Infrarød overføring trenger ikke Blåtann fri sikt til de forbundne enhetene.

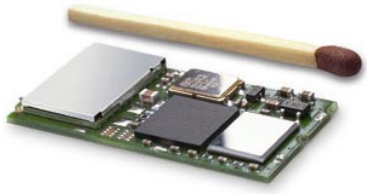
Teknologien er fysisk liten og har lavt strømforbruk. Dagens prototyper er på 0,9cm2 med en mye mindre krets under utvikling. Forventet pris per enhet er 200kr i 2000, og 30kr innen 2002. Ønsket utvikling er at Blåtannenhetene er integrert som standardutstyr i de fleste elektriske innretninger i nærmeste fremtid.

BlåtannSIG ble grunnlagt av Ericsson, IBM, Nokia, Intel og Toshiba. [10]

2.8 Hvordan fungerer Blåtann ?

Hver enkelt Blåtannenhet er utstyrt med en sender og mottaker på størrelse med en mikrobrikke. Enheten bruker det tidligere ubrukte 2,45 GHz båndet som er globalt tilgjengelig med enkelte variasjoner i båndbredden i enkelte land [1;19]. I tillegg til en

datakanal er opp til tre lydkanaler tilgjengelige. Hver enhet har en unik 48-bit adresse fra IEEE 802 standarden [9].



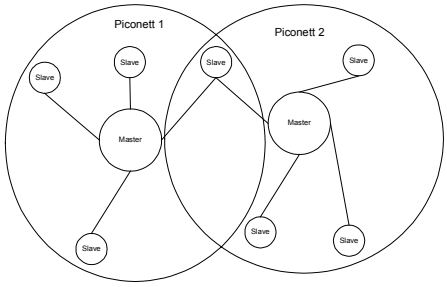
Figur 3 - Fysisk størrelse på Ericsson AS sin Blåtannmodul

Maksimal rekkevidde for Blåtann er 10 meter, og data kan utveksles med 1Mbit pr sekund i versjon en og opp til 100meter med 2Mbit i versjon to. Båndbredden er felles for alle enhetene i samme Piconett, så hvis to enheter bruker 300Kbit, er det kun 700Kbit igjen til de resterende enhetene. Ved frekvenskollisjoner mellom flere enheter fungerer overføringen på samme måte som ved ethernet. Hvis du skal si noe til en som står på andre siden av lokalet, roper du til vedkommende. Blir du ikke hørt, roper du en gang til osv. Den som roper høyest får frem sitt budskap.

Siden Blåtann bruker frekvenshopping kan enhetene kommunisere med hverandre selv i områder med mye elektromagnetisk interferens. Siden det i enkelte land er begrensning på 2,4GHz området er det designet to versjoner av frekvenshoppingen. En versjon med 79 hopp og en med 24 hopp. Frekvenshoppingen skjer vanligvis 1600 ganger hvert sekund (Det vil si en tidsluke på 625µs), men ved søking etter andre enheter skjer frekvenshoppingen 3200 ganger i sekundet.

Forbindelser kan enten gjøres en til en eller en til mange. Blåtannettverk kalles Piconett og består alltid av en master og opp til 7 aktive (og 255++ passive) slaver. Det er ikke nødvendig med en basestasjon.

Ved større nettverk enn 8 enheter har man Scatternett som er nettverk av opp til 10 Piconett. En slave i ett Piconett kan være master i ett annet. Ved å bruke Scatternett vil båndbredden i hvert av Piconettene være 333Kbit per sekund



Figur 4 - Scatternett bestående av to Piconett

Blåtann har ett forbruk på 10 pikowatt ved normalt bruk. Ved ett batteri på 600mAh vil man ved beredskap ha 3 måneders kapasitet, ved dataoverføring 100 timers kapasitet og taleoverføring 60 timers kapasitet.

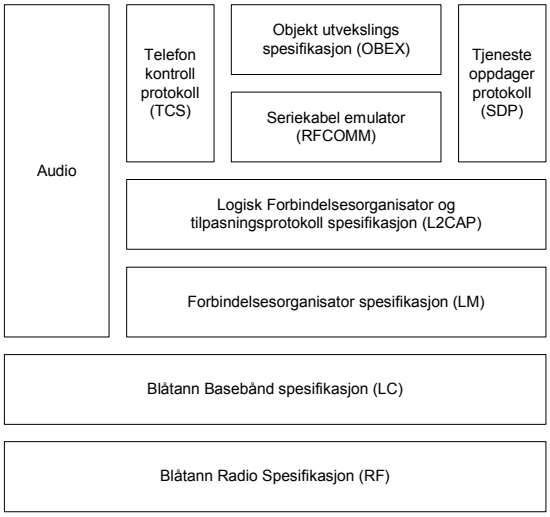
Spesifikasjon	*Verdi*
Frekvens	2,4GHz
Uteffekt	1mW ved 0dB, 100mW ved 20dB
Rekkevidde	10m ved 0dB (100m ved 20dB)
Modulasjon	GFSK (Gaussian Frequency Shift Key)
Modulasjon index	0.32 ±1%
Bit Rate	1Mbps ± 1ppm
Modulasjonsdata	PRBS9

Tabell 1 - Informasjon om Blåtann

Det finnes innebygde sikkerhetsalgoritmer for autentisering og kryptering i Blåtann, mer om dette i kapittel 3.

2.9 Protollstakken til Blåtann

Blåtann er delt opp i flere lag fra radiolaget til applikasjonslaget. Under er en oversikt med forklaring til de forskjellige lagene.



Radiospesifikasjonen (RF) [1:17] tar for seg kravene til radiolinken; Hvilke frekvenser som skal brukes i forskjellige deler av verden, krav til effektforbruk og rekkevidde, modulasjon, følsomhet osv.

Basebånd spesifikasjonen (LC) [1:35] tar for seg forbindelses kontrollen som for eksempel pakketype, pakkesignatur, kanalkontroll, feilretting, sende og mottakingsrutiner, frekvenshopping, sikkerhetsaspekter, adressering og synkronisering.

Figur 5 - Protokollstakken til Blåtann

Forbindelsesorganisatoren (LM) [1:187] tar seg av oppsetting av forbindelsene, sikkerhet og kontroll av disse. Informasjon blir lagt til dataene ved utsending, og filtrert bort ved mottaking

Logisk Forbindelsesorganisator og tilpasningsprotokoll (L2CAP) [1:247] støtter høyere lags protokoll multipleksing, pakke segmentering og sammensetning samt transport av tjenestekvalitets informasjon. Protokollen behandler dataene fra forbindelsesorganisatoren og sender dem videre til en av mulitpleksingsprotokollene over, og omvendt.

Tjeneste oppdager protokollen (SDP) [1:323] skal finne tjenester som er tilgjengelig for Blåtannenheter og holde denne listen oppdatert til enhver tid selv om enheten beveger seg eller andre enheter kommer til eller faller fra.

Serieport emulatoren (RFCOMM) [1:385] er en enkel protokoll for å emulere RS-232 standarden over Blåtann. Protokollen støtter opp til 60 samtidige forbindelser mellom to enheter.

Telefon kontroll protokollen (TCS) [1:429] tar seg av kontroll av tale, oppsett og nedkobling av samtaler, gruppekontroll for flere enheter, feilkontroll av data, format på data og hvordan kode informasjonen.

Objektsutvekslings spesifikasjonen (OBEX) [1:411] er en høyerelags spesifikasjon som ble laget for overføring over IR. De lavere IR lagene er byttet ut med Blåtanns kommunikasjonslag. Eksempler på applikasjoner som bruker denne protokollen er : Synkronisering og filutveksling

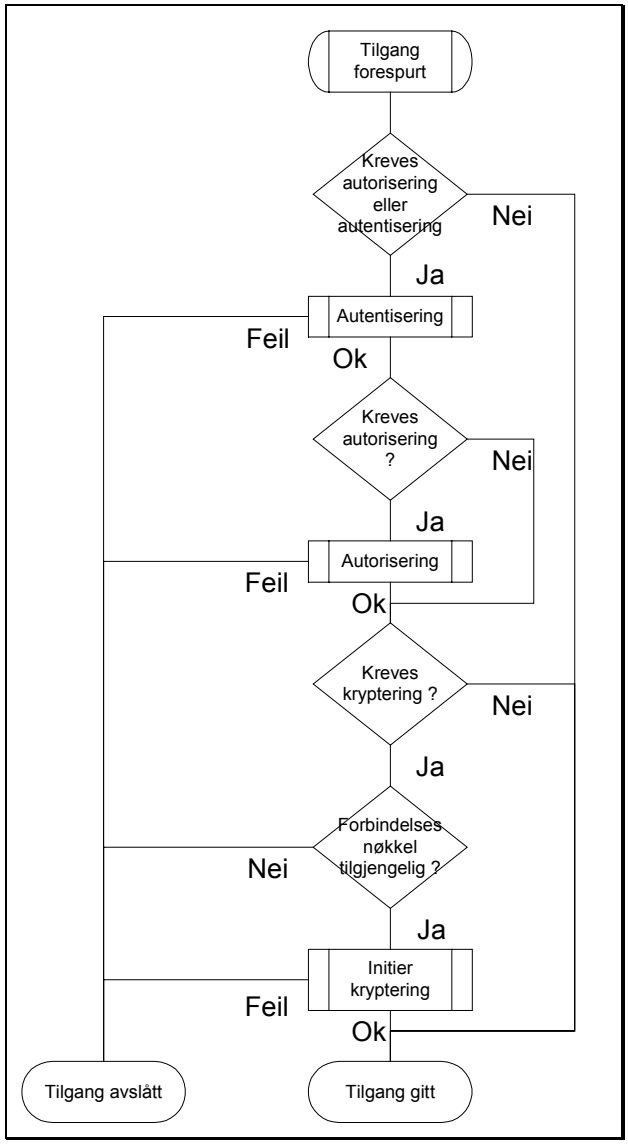
I tillegg finnes det en lydmodul [1:987]. Denne inneholder informasjon om lydnivå, telefonnettverkskrav og frekvensmaske.

3 Sikkerhet i Blåtann

Siden radiosignaler enkelt kan avlyttes er det viktig at Blåtannenheter har innebygget sikkerhet for å motvirke avlytting eller falsifisering av meldinger (Spoofing). Dette er implementert på forbindelseslaget.

3.1 Sikkerhetsmekanismene i Blåtann

Blåtannspesifikasjonen[1] inkluderer sikkerhetsmekanismer på forbindelseslaget. Dette laget støtter autentisering og kryptering. Disse egenskapene baserer seg på en hemmelig forbindelsesnøkkel som er delt mellom enhetene. For å generere denne nøkkelen blir det brukt en paringsprosedyre første gang de to enhetene kommuniserer med hverandre. Autentisering motvirker falsifisering av meldinger og uønsket tilgang til kritiske data og funksjoner, mens krypteringen sikrer mot avlytting og opprettholder en privat forbindelse mellom enhetene. Se figur 6 nedenfor. I tillegg til disse forbindelseslag og applikasjons funksjonene vil frekvenshopping og avstandsbegrensningen også hjelpe til med å motvirke avlytting.



Figur 6 - Informasjonsflyt ved tilgangskontroll

Siden forskjellige applikasjoner har forskjellige krav til sikkerhet, er det innført tre sikkerhetsnivåer på forbindelseslaget. Blåtannspesifikasjonen definerer disse tre sikkerhetsnivåene i figur 7 nedenfor som :

3.1.1 Nivå 1 : Ikke sikker

Sikkerhetsnivå én skal brukes til enheter som ikke innehar noen kritiske applikasjoner. Den utelukker sikkerheten på forbindelseslaget, og er egnet til å aksessere for eksempel databasser som ikke inneholder sensitiv informasjon. Ett eksempel på dette kan være utveksling av elektroniske visittkort.

3.1.2 Nivå 2 : Tjenestenivå sikkerhet

Dette sikkerhetsnivået tillater forskjellig tilgang for applikasjoner med forskjellige krav. Det er mulig å definere sikkerhetsnivåer for enheter og tjenester. Det er to mulige måter å kategorisere enhetene :

1. En enhet med tillit *(trusted)* har en fast oppsatt forbindelse og har ubegrenset tilgang til alle tjenestene.
2. En enhet uten tillit har ikke fast oppsatt forbindelse (men kanskje en temporær), eller har en fast oppsatt forbindelse men er ikke betrodd. Denne enheten har ikke tilgang til alle tjenestene.

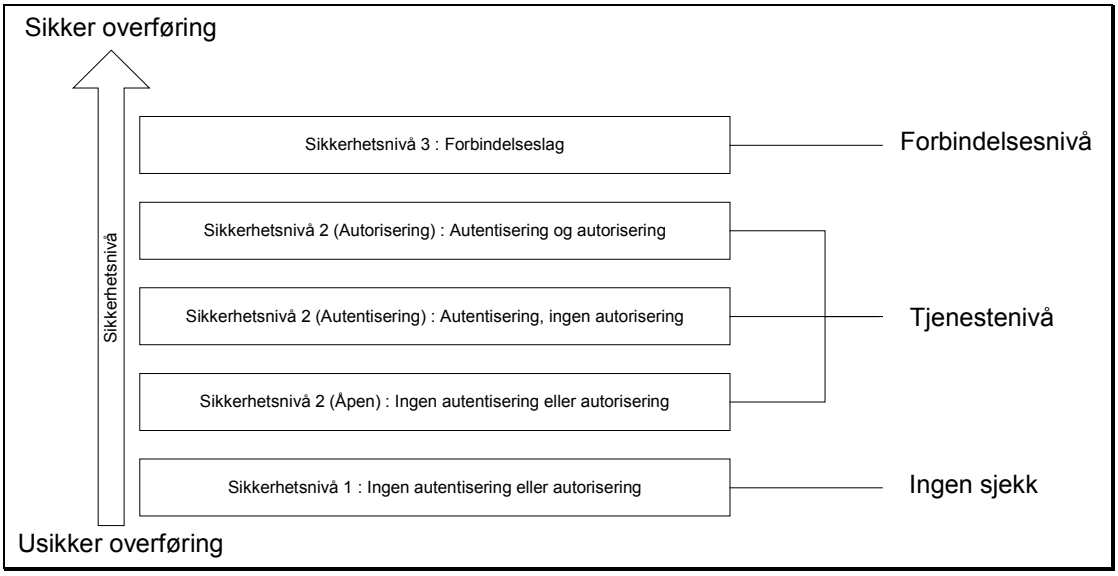
En mulig spissfindighet er å sette tillitsnivået til en enhet spesifikt for tjenester eller grupper av tjenester. For tjenester er autorisering (Tilgang eller ikke til en tjeneste), autentisering (Identifisering av hvem man kommuniserer med) og kryptering satt uavhengig av andre tjenester. Tre sikkerhetsnivåer styrer hvilken tilgang tjenesten skal ha :

1. Tjenester som krever autorisering og autentisering. Automatisk tilgang blir kun innrømmet til betrodde enheter, mens andre enheter må ha manuell autorisering.
2. Tjenester som trenger kun autentisering
3. Tjenester som er åpne for alle enheter.

Normalt blir sikkerhetsnivået satt til kravene til en applikasjon. Denne standard policyen vil bli brukt hvis ikke andre innstillinger blir funnet i en sikkerhetsdatabase som er i sammenheng med tjenesten. For eksempel en intern sikkerhetsinformasjons database. En mulig måte å gjøre dette på er å innføre en sikkerhetsorganisator som tar seg av tilgangen på bakgrunn av tillit til enheten og sikkerhetsnivået til tjenesten som begge blir hentet fra en intern database. Mer om dette finnes i kapittel 4. Blåtann er ikke lagd for å erstatte eksisterende nettverk sikkerhetsprosedyrer, men derimot bruke de eksisterende. For ekstremt høy sikkerhets applikasjoner som for eksempel elektronisk handel må man legge til sikkerhet i applikasjonslaget. Mer informasjon om dette kan finnes i Blåtann profilseksjonen [2] hvor denne tilnærmingen allerede har vært brukt.

3.1.3 Nivå 3 : Forbindelseslag sikkerhet

I dette sikkerhetsnivået vil forbindelsesorganisatoren påtvinge sikkerhet til samme nivå for alle applikasjonene i begynnelsen på oppkoblingen. Selv om dette sikkerhetsnivået er mindre fleksibelt enn sikkerhetsnivå to, er det lettere å implementere.



Figur 7 - Oversikt over sikkerhetsnivåene i Blåtannspesifikasjonen

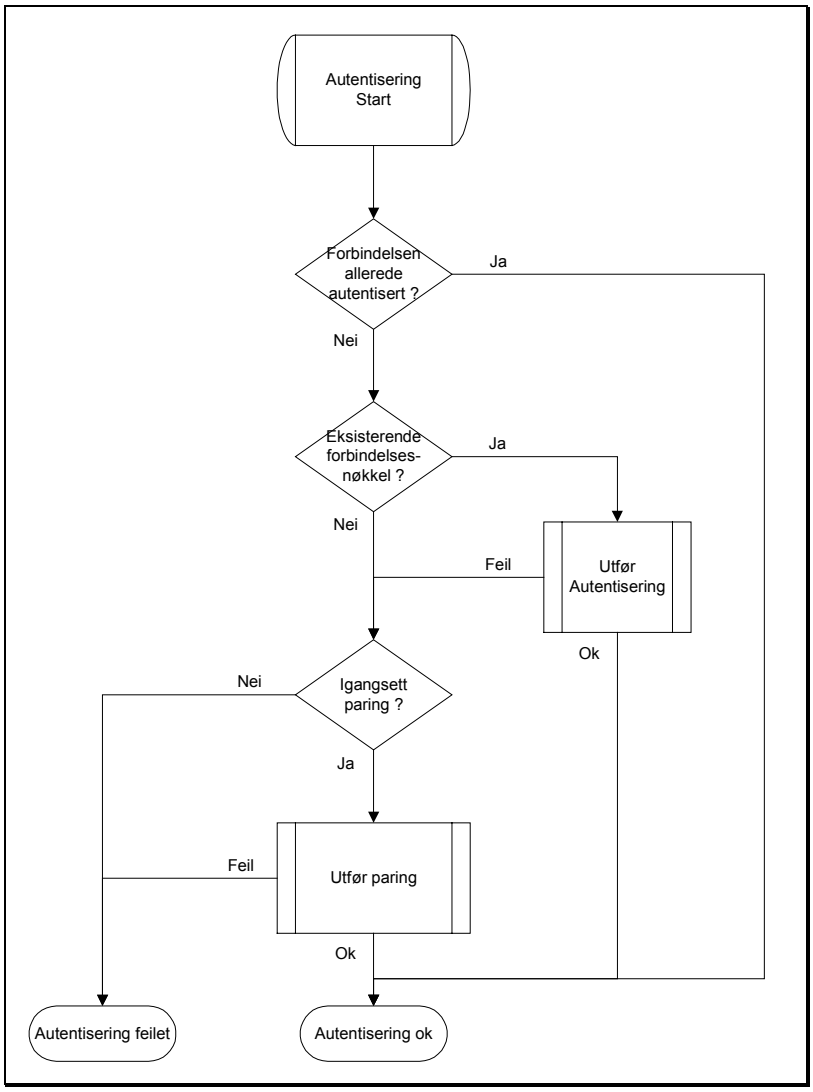
Alle forbindelseslag sikkerhetsfunksjoner er basert på konseptet av en forbindelsesnøkkel. Dette er en hemmelig nøkkel bestående av ett 128 bit tilfeldig nummer som er lagret individuelt for hvert enkelt par av enheter. Hver gang disse enhetene kommuniserer via Blåtann vil forbindelsesnøkkelen bli brukt for autentisering og kryptering uten påvirkning av Piconet topologi. Den mest sikre forbindelsesnøkkelen er en kombinasjonsnøkkel med data fra både sender og mottaker.

For enheter med liten lagringskapasitet kan man bruke en enhetsnøkkel som kan brukes for flere enheter. I tillegg må man ha en temporær nøkkel for en eventuell kringkasting til flere enheter. Denne kan naturlig nok ikke brukes til autentisering, men motvirker avlytting fra enheter utenfor Piconettet. (Men ikke fra enheter som deler denne temporære nøkkelen.)

3.2 Autentisering

Autentiseringprosedyren trenger ikke data fra brukeren. Autentiseringsnøkkelen har en fast størrelse på 128 bits, og vil være mer statisk enn krypteringsnøkkelen. Når en forbindelse er satt opp, er det applikasjonen på Blåtann enhetene som bestemmer når (om i det hele tatt) man skal forandre denne nøkkelen.

Blåtannenhetene autentiseres slik figur 8 viser



Figur 8 - Skjematisk fremstilling av autentiseringsprosedyren

Ved autentisering sjekkes det først om enhetene allerede har vært autentisert mot hverandre. Hvis de har det, vil forbindelsesnøkkelen fra forrige gang bli brukt om igjen, og hvis ikke vil det bli opprettet en ny. Denne prosedyren er kalt paring (Algoritme E2.2). Den mest vanlige måten å utføre paring på forutsetter at brukeren har tilgang til begge Blåtannenhetene til samme tid. Ved førstegangs kommunikasjon krever parings prosedyren at man taster inn en sikkerhetskode på maksimalt 16 byte eller 128 bit på de to enhetene. Hvis dette gjøres manuelt, er ofte koden kortere.

Selv om Blåtann sikkerhetskoden ofte er referert til ”PIN” (Personal Identity Number), er dette ikke en kode som brukeren trenger å huske da denne som oftest kun blir brukt en gang. Når en forbindelsesnøkkel blir slettet av en eller annen grunn og den initielle paring prosedyren må gjentas kan hvilken som helst kode tastes inn. Hvis man ønsker ett lavere sikkerhetsnivå kan man ha en fast kode i enheter som ikke har mulighet for brukeren til å taste inn en kode.

Algoritme E2.2

$$E_{22} : \{0,1\}^{8L'} x \{0,1\}^{128} x \{1,2,...,16\} \rightarrow \{0,1\}^{128}$$

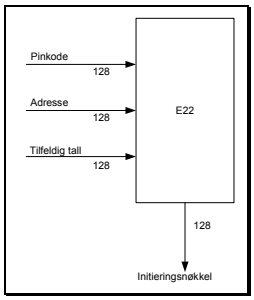
$$(\text{Pinkode}, \text{Tilfeldig tall}, \text{Antall oktetter}) \rightarrow \textit{Initieringsnøkkel}(X, Y)$$

Hvor

$$\begin{cases} X = \bigcup_{i=0}^{15} PIN[i(\bmod L)], \\ Y = \text{Tilfedig tall}[0....14] \cup (\text{Tilfeldig tall}[15] \oplus L) \end{cases}$$

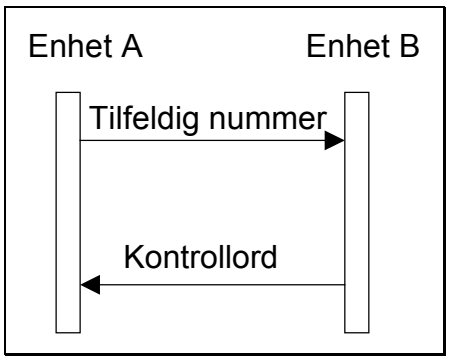
Paringprosedyren innebærer :

a. Generering av ett felles tilfeldig tall (Initierings Nøkkel)fra koden som brukeren taster inn. Dette blir kun brukt en gang før det blir slettet.
b. Autentisering som sjekker at Blåtann sikkerhetskoden er identisk i de to enhetene.
c. Generering av ett felles 128 bit tilfeldig tall (Forbindelsesnøkkel) som blir lagret temporært eller delvis permanent i de to parete enhetene.



Figur 9 - Generering av initialiseringsnøkkelen (paring)

Etter at initieringsnøkkelen er verifisert av begge parter blir den slettet, og autentiseringsprosedyren kan fortsette. For hemmelig symmetriske nøkler blir verifiseringen som figur 10 nedenfor :



Figur 10 - Verifisering av initieringsnøkkel for hemmelig symmertiske nøkler

Enhet A sender ett tilfeldig nummer til enhet B, og koder dette tilfeldige tallet med adressen til enhet B og initieringsnøkkelen. Enhet B gjør det samme og sender den krypterte meldingen (kodeordet) tilbake til enhet A. Denne sjekker sitt eget kodeord med kodeordet fra enhet B. Er disse kodene like går autentiseringsrutinen videre.

Neste steg er å generere en enhetsnøkkel. Denne blir kun laget første gang Blåtann enheten utfører en operasjon, og ikke for hver initialisering. Den blir nesten aldri forandret. Nøkkelen genereres av algoritmen E2.1.

Algoritme E2.1

$$E_{21} : \{0,1\}^{128} x \{0,1\}^{48} \rightarrow \{0,1\}^{128}$$

$$(\text{Tilfeldig tall}, \text{Adressen}) \rightarrow \textit{Initieringsnøkkel}(X, Y)$$

Hvor

$$\begin{cases} X = \text{Tilfedig tall}[0....14] \cup (\text{Tilfeldig tall}[15] \oplus 6) \\ Y = \bigcup_{i=0}^{15} \textit{Adressen}[i(\text{mod}\,6)], \end{cases}$$

Hvis enhetsnøkkelen forandres må tidligere autentiserte enheter autentiseres på nytt igjen. For enheter med enkle krav til autentiseringssikkerhet, blir kun enhetens egen nøkkel lagret (Dette gjør at enheten ikke trenger mye hukommelse og er ofte brukt i utstyr som skal være aksesserbart fra store grupper av brukere). Den andre enheten vil da lagre denne nøkkelen i sin hukommelse (i tillegg til sin egen enhetsnøkkel), og denne nøkkelen blir brukt i overføringen. Med enheter som krever mer kompleks autentiseringssikkerhet må man lagre en kombinasjonsnøkkel som består av informasjon fra forbindelsesnøkkelen til begge enhetene.

Hvis det er ønskelig å bruke en kombinasjons nøkkel, blir denne generert i initialiserings prosedyren. Denne består av kombinasjoner mellom tall generert i de to enhetene. Først genereres ett tilfeldig tall i hver enhet. Så brukes E2.1 algoritmen på dette tallet og adressen til enheten. (Dette gjøres på begge enhetene) Det tilfeldig valgte tallet i hver enhet blir xoret med forbindelsesnøkkelen, og sendt til den andre enheten.

Enhetene bruker det tilfeldig valgte tallet til den andre enheten sammen med adressen, og finner da forbindelsesnøkkelen til mottakeren. Denne nøkkelen blir slått sammen med forbindelsesnøkkelen til enheten (vha en enkel modulo 2-addering), og man får ut en 128 bit forbindelsesnøkkel.

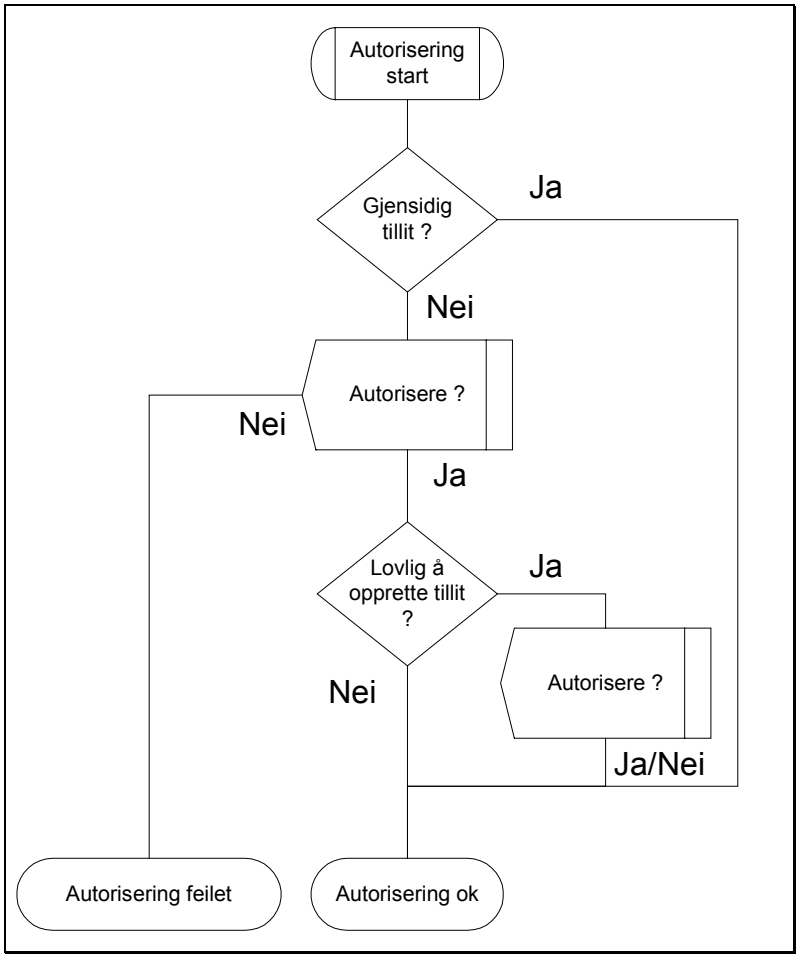
Den nye forbindelsesnøkkelen utveksles, og den gamle slettes hvis den nye stemmer overens.

Hvis kryptering skal benyttes vil krypteringsprosedyren bli utført (Se kapittel 3.4)

3.3 Autorisering

Autoriseringsrutinen er skjematisk beskrevet i figur 11. Ved autorisering må det sjekkes om enhetene har gjensidig tillit til hverandre mot en sikkerhetsdatabase. Hvis enheten finnes i autorisasjonsdatabasen er autorisasjonen godkjent, og enhetene kan kommunisere med hverandre. Har derimot ikke enhetene gjensidig tillit må det utføres en autoriseringsprosedyre. Denne prosedyren sjekker mot sikkerhetsdatabasen om enheten skal ha tilgang, og hvis ikke vil ikke enheten bli autorisert. Hvis enhetene blir autorisert

sjekkes det om det er lov å legge til enheten til sikkerhetsdatabasen. Uansett om informasjon om enheten blir lagt inn i databasen eller ikke, så vil autoriseringen være godkjent og enhetene har oppnådd gjensidig tillit.



Figur 11 - Informasjonsflyt i autoriseringsprosedyren

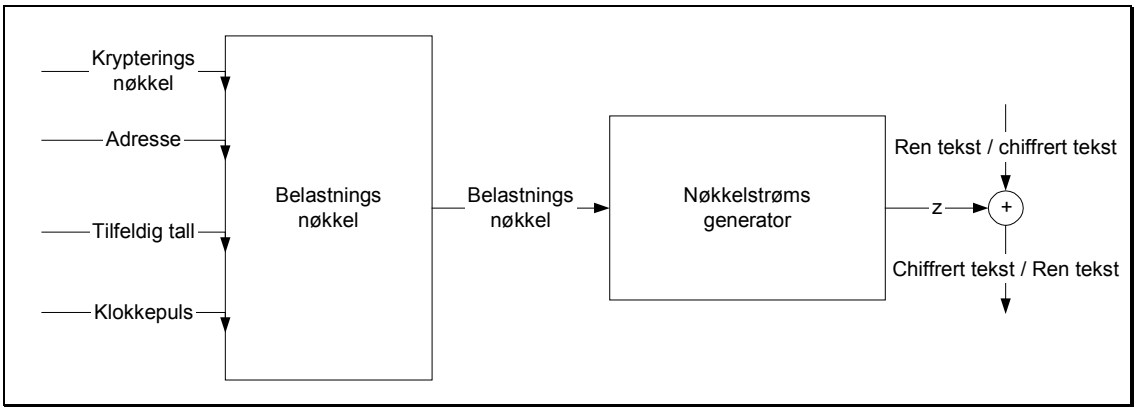
3.4 Kryptering

Krypteringsnøkkelen har en variabel størrelse på 2 til 128 bit som blir satt fra fabrikk. Lengden på krypteringsnøkkelen kan ikke overstyres av to grunner. For det første har en del land offisielle holdninger til hemmeligholdelse og / eller strenge krav til eksport med henblikk på kryptering. Den andre grunnen er at man ikke ønsker å designe protokollen på nytt for en senere oppgradering av krypteringsalgoritmene.

Krypteringsnøkkelen er ikke den samme som autentiseringsnøkkelen, selv om autentiseringsnøkkelen brukes for å lage krypteringsnøkkelen. Hver gang kryptering blir aktivert, blir en ny krypteringsnøkkel laget. Grunnen til at man har to forskjellige lengder på autentiseringsnøkkelen og krypteringsnøkkelen er at man ikke ønsker å svekke autentiseringsrutinene selv om man har en svak kryptering. (Det finnes ingen offisielle begrensninger på hvor lang autentiseringsnøklene kan være, kun krypteringsnøkler)

Chiffreringssystemet består av tre deler. Del én (initialiseringen) er generering av belastningsnøkkelen. Denne er meget enkel, og bruker de fire LFSR (Linear Feedback Shift Registers) fra nøkkelstrøm generatoren (Del to). Del to genererer nøkkelstrøm bits

som er utledet av en metode fra Massey og Rueppel [7]. Metoden har blitt utforsket meget grundig, og det finnes gode estimater på hvor solid denne er med henblikk på krypteringsanalyse. Rutinen har svakheter som kan brukes i såkalte gjentatte angrep, men dette motvirkes ved at frekvensen byttes ofte og at det innføres ventetid. Hver gang det feiles må det ventes ett gitt tidsrom, og dette tidsrommet øker hver gang man feiler fra samme adresse. Den tredje delen i chiffreringssystemet utfører kryptering og dekrypteringen.



Figur 12 - Chiffreringsrutine for kryptering av data.

For å bestemme lengden på krypteringsnøkkelen, må man finne maksimal lengde på nøkkelen som er tillatt hos de forskjellige Blåtann enhetene. Dette nummeret er i mengden 1 til 16 oktetter. Masteren (Enheten som initierer kryptering) sender ut forslag om hvor lang krypteringsnøkkelen skal være (maksimal lengde), og slaven svarer hvor lang nøkkel denne tillater. Hvis dette tallet er høyere enn det tallet som masteren foreslo, vil man bruke masterens lengde på krypteringsnøkkelen. Hvis ikke vil slaven sende ut forslag til masteren som er det høyeste slaven støtter. Slik fortsetter man til man er kommet frem til en løsning, forutsatt at krypteringsnøkkelen er lengre enn det som er satt som minimum av masteren.

Generering av krypteringsnøkkelen skjer i E3 algoritmen som blir kjørt hver gang forbindelsesorganisatoren aktiverer kryptering. Dette gjør at krypteringsnøkkelen blir forskjellig fra hver gang man velger å bruke kryptering. E3 algoritmen bruker forbindelsesnøkkelen, en 96 bit temporær forbindelsesnøkkel og ett tilfeldig tall på 128 bit som input. Den temporære forbindelsesnøkkelen kan bestemmes på to måter avhengig av forbindelsesnøkkelen. Hvis forbindelsesnøkkelen kun kommer fra masteren så vil den temporære forbindelsesnøkkelen bli utledet av adressen til denne enheten. Hvis forbindelsesnøkkelen ble utledet fra begge enhetene, vil den temporære forbindelsesnøkkelen bli satt til verdien fra autentisert forbindelsesnøkkel.

Algoritme E3

$$E_3 : \{0,1\}^{128} x \{0,1\}^{128} x \{0,1\}^{96} \rightarrow \{0,1\}^{128}$$

$$(Forbindelsesnøkkel,\ Tilfeldig\ tall, Temporær\ forbindelsenøkkel)$$

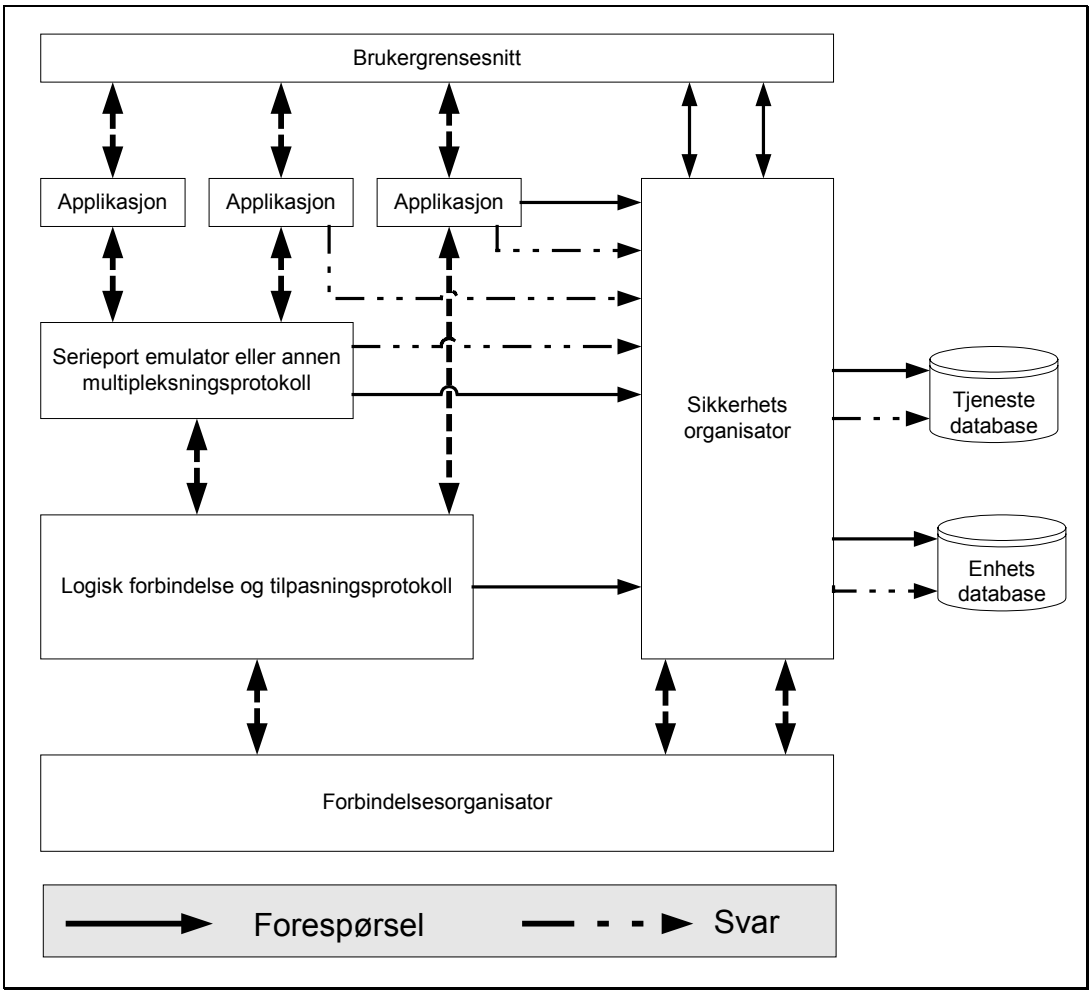
$$\rightarrow Hash(Forbindelsesnøkkel,\ Tilfeldig\ tall, Temporær\ forbindelsenøkkel, 12)$$

Kryptering for baseband forbindelsen trenger ikke noe data fra brukeren. Etter en vellykket autentisering og mottak av forbindelsesnøkkelen genererer denne funksjonen en ny krypteringsnøkkel fra forbindelsesnøkkelen. Dette blir gjort hver gang enhetene

autentiseres med hverandre. Krypteringsnøkkelen blir lagd til 128 bit, men antall bit som blir brukt varierer mellom 8 og 128 avhengig av hvilken sikkerhet som er påkrevd og eksport reguleringer. Maksimum krypteringsnøkkel lengde er begrenset i maskinvaren.

4 Forbedringer

Det finnes mange måter å forbedre og effektivisere sikkerhetssjekkingen i Blåtann. En av disse mulighetene er å innføre en sikkerhetsorganisator til protokollen for sikkerhetsnivå to og tre. Dette er en modul som vil ligge oppå forbindelsesorganisatoren, men på siden av alle protokollene over denne som f.eks. serieportemulatoren. Se figur 13 nedenfor.



Figur 13 - Sikkerhetsarkitektur med sikkerhetsorganisator

I følge denne modellen kan ikke pakkene på det logiske forbindelses og tilpasningslaget være forbindelsesløse. Hvis pakkene skal være forbindelsesløse må hver enkelt pakke sjekkes i lagene over, og man kan ikke verifisere hvem pakkene kommer fra. Derfor er det i resten av denne rapporten gått ut fra at alle forbindelsesløse pakker blir forkastet i det logiske forbindelse og tilpasningslaget. Det er derimot en mulighet å forespørre alle tjenester i enheten om det er noen som bruker forbindelsesløse pakker, for så å slippe gjennom alle pakker eller stenge for alle forbindelsesløse pakker til disse tjenestene.

Sikkerhetsorganisatoren kan inneholde følgende tjenester :

- Lagre sikkerhetsrelatert informasjon om tjenester
- Lagre sikkerhetsrelatert informasjon om enheter
- Svare på forespørsler fra protokoll implementasjoner eller applikasjoner om det skal gis tilgang eller ikke.

- Påtvinge autentisering og/eller kryptering før oppkoblingen til en applikasjon.
- Initiere eller prosessere input fra en ekstern sikkerhetsmodul (ESCE - External Security Control Entity) for å sette opp en gjensidig tillit på enhetsnivå.
- Initiere paring og forespørre etter PIN kode. (Forespørsel etter PIN kode kan også gjøres fra applikasjonene.)

4.1.1 Sikkerhetsnivå 1 : Ikke sikkert

Da sikkerhetsnivå én skal være åpen for alle, er det ikke lagt til noe forandringer på dette nivået.

4.1.2 Sikkerhetsnivå 2 : Tjeneste nivå

Tillitsnivåer (Trust):
Det finnes to forskjellige tillitsnivåer til Blåtannenheter. Enten har man gjensidig tillit eller så har man det ikke. De enhetene som har gjensidig tillit har allerede vært autentisert, forbindelsesnøkkelen er utvekslet og enheten er merket med tillit i enhetsdatabasen. De enhetene som ikke har gjensidig tillit har enten autentisert seg tidligere, forbindelsesnøkkelen er utvekslet, men enheten er ikke merket med tillit i enhetsdatabasen eller så er enheten ukjent. Ukjente enheter har ikke autentisert seg tidligere, og er derfor kategorisert uten tillit.

Tjenestenivåer :
Sikkerhetsnivået til en tjeneste er definert med tre attributter :

- Krever autorisering :
  Tilgang blir kun gitt automatisk til enheter som får automatisk gjensidig tillit eller andre enheter etter en autorisasjonsprosedyre.
  Autorisering krever alltid autentisering for å være sikker på hvem mottakeren er.
- Krever autentisering :
  Det må skje en autentisering før oppkobling til applikasjoner
- Krever kryptering :
  Forbindelsen må gjøres om til kryptert før tilgangen til tjenesten blir gitt.

Informasjon om tillit og kryptering blir lagret i databasen til sikkerhetsorganisatoren.

Hvis det ikke har vært noen registrering for tjenesten (se kapittel 4.1.5) blir standard sikkerhetsnivå brukt. Disse nivåene er for inngående forbindelse autorisering og autentisering, og for utgående forbindelse autentisering.

4.1.3 Sikkerhetsnivå 3 : Enhetsnivå

Selv om denne arkitekturen ikke tar sikte på sikkerhetsnivå tre kan allikevel arkitekturen støtte dette sikkerhetsnivået. Sikkerhetsorganisatoren kan tvinge forbindelsesorganisatoren til å utføre autentisering før man aksepterer en basebånd forbindelse.

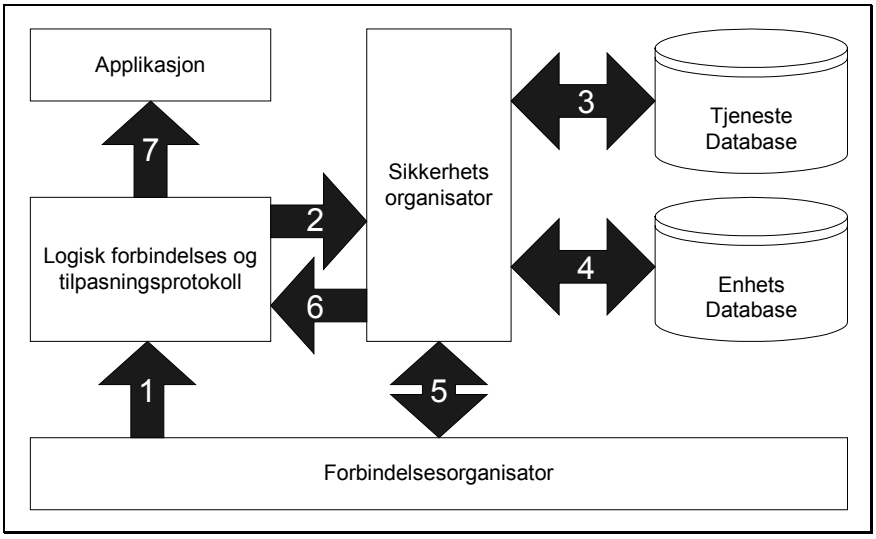
Forskjellige moduler kan bli implementert i parallell :

- Autentisering på basebånd forbindelsen
- Autentisering på forbindelse til applikasjoner

Disse to modulene kan ikke brukes samtidig, så hvis begge er implementert må man passe på når man går fra sikkerhetsnivå to til sikkerhetsnivå tre at ingen enheter får uønsket tilgang. For å få dette til kan sikkerhetsorganisatoren fjerne forbindelsesnøkler for enheter uten gjensidig tillit som er lagret i radiomodulen. De enheter som da ikke har gjensidig tillit må da autentisere og bli autorisert før de kan få tilgang til tjenestene.

4.1.4 Oppsett av en oppkobling

Etter at sikkerhetsnivået til tjenestene er kartlagt og en oppkoblingsforespørsel er gitt , kan autentisering starte. Figur 14 viser dataflyten for tilgang til en tjeneste som krever gjensidig tillit.



Figur 14 - Informasjonsflyt for tilgang til enheter med gjensidig tillit

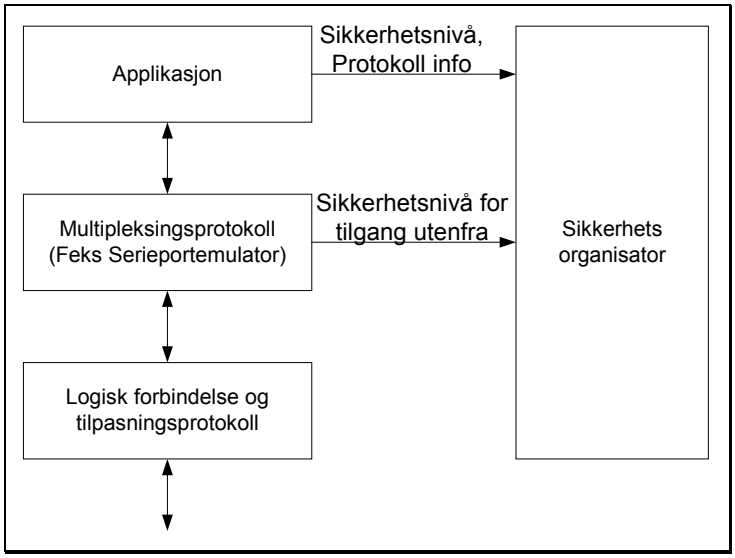
Følgende prosedyrer er utført :

1. Forespørsel om forbindelse til logisk forbindelse og tilpasningsprotokoll
2. Logisk forbindelse og tilpasningsprotokoll forespør Sikkerhetsorganisator om tilgang.
3. Sikkerhetsorganisator sjekker tjeneste databasen
4. Sikkerhetsorganisator sjekker enhetsdatabasen
5. Sikkerhetsorganisator godkjenner tilgang
6. Logisk forbindelse og tilpasningsprotokoll fortsetter med oppsett av forbindelse.
7. Applikasjonen får beskjed om at autentisering er ok.

Autentisering kan utøres i begge retninger : Klient autentiserer server og omvendt.

4.1.5 Registrering av applikasjoner i sikkerhetsorganisator

Sikkerhetsorganisatoren vedlikeholder sikkerhetsinformasjon for tjenestene i sikkerhetsdatabasen. Applikasjoner må registrere seg hos sikkerhetsorganisatoren før de blir tilgjengelige. Se figur 15 nedenfor.



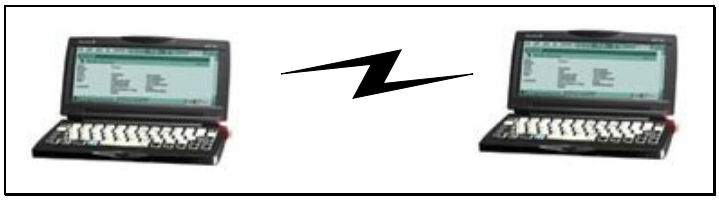
Figur 15 - Registreringsprosessen til applikasjoner

Applikasjoner må registrere sikkerhetsnivå til sikkerhetsorganisatoren hvis ikke applikasjonen ønsker å bruke standard sikkerhetsnivå. Disse nivåene er for inngående forbindelse autorisering og autentisering, og for utgående forbindelse kun autentisering.

5 Drøfting

For å finne begrensninger i sikkerheten til Blåtann kan man ta utgangspunkt i fire tenkte scenarioer :

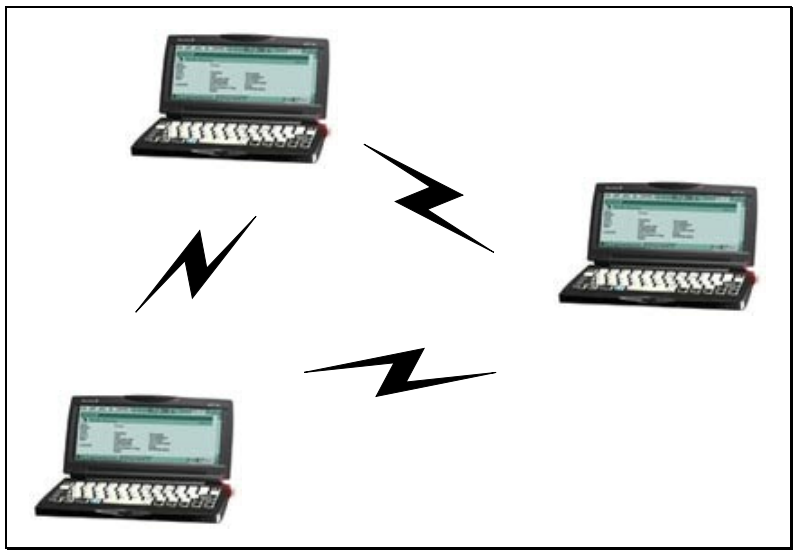
Scenario én. Det er to Blåtannenheter (F.eks. to PDAer). Begge disse enhetene har et sett av applikasjoner som kalender, telefonbok, fil synkronisering osv. De to enhetene vil kommunisere med hverandre over Blåtann for å utføre en fil synkronisering.



Figur 16 - Scenario 1. To PDAer som utveksler informasjon

Ved å se på scenario én vil man finne at tilgangsnivå for tjenester ikke er definert. Dette betyr at har man først satt opp en forbindelse mellom to Blåtannenheter og blitt autentisert så har man full tilgang til alle tjenester i enheten. Derimot kan en mer fleksibel sikkerhetspolicy implementeres uten å forandre Blåtannprotokollstakken, men kun modifisere sikkerhetsorganisatoren og registreringsprosessene til applikasjonene. Denne sikkerhetsorganisatoren er omtalt i kapittel 4.

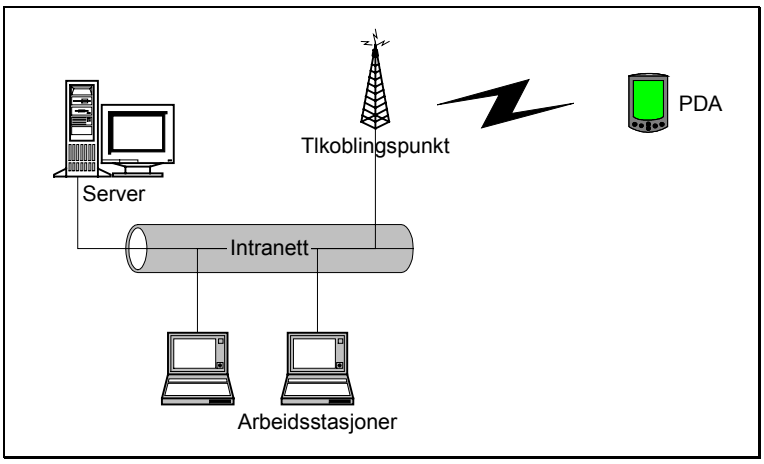
Scenario to. Det finnes mer enn to enheter i scenario én. Alle enhetene vil kommunisere over Blåtann for å utveksle informasjon som ikke trenger autentisering som f.eks. mottak av visittkort (man ønsker ikke å gi fra seg sitt eget).



Figur 17 - Scenario 2. Tre PDA'er som utveksler visittkort

Da enhetene kun blir sjekket mot hverandre ved oppsett av forbindelsen, og det etter dette i er prinsipp full toveis kommunikasjon mellom enhetene vil det ikke være mulig ved hjelp av dagens arkitekturen å få kun enveis kommunikasjon. Applikasjonen må da designes slik at ikke andre enheter kan hente informasjon fra andre åpne tjenester.

Scenario tre. En PDA ønsker å få tilgang til intranett via Blåtann til ett lokalnettverk for å få tilgang til tjenester som email, Internett, firmadatabaser osv. Vi har tre noder som skal kommunisere. Node en er PDA'en, node to er tilkoblingspunktet til nettverket og node tre er intranettet.

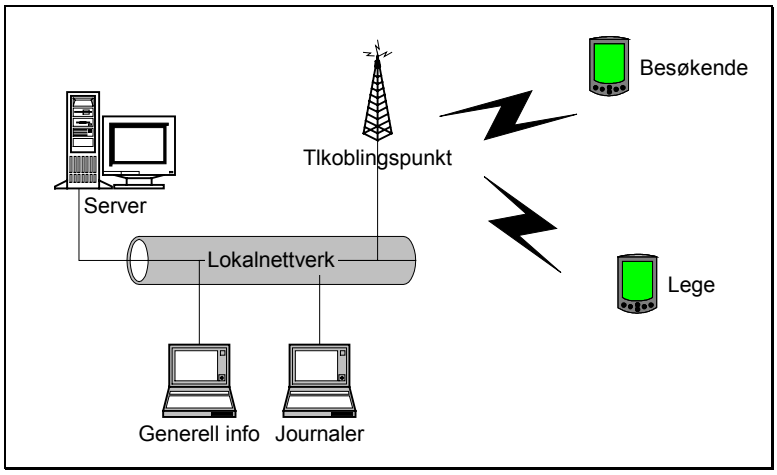


Figur 18 - Scenario 3. Tilgang til lokalnettverk via Blåtann

Ser på scenario tre med de tre nodene. I Blåtannspesifikasjonen blir det kun sett på linken mellom Blåtannenhetene, i dette tilfelle PDA'en og Tilkoblingspunktet. Kommunikasjonen mellom PDA'en og Serveren er utenfor Blåtannspesifikasjonen, og sikring av data må skje på applikasjonslaget. Dette er utenfor området til denne oppgaven.

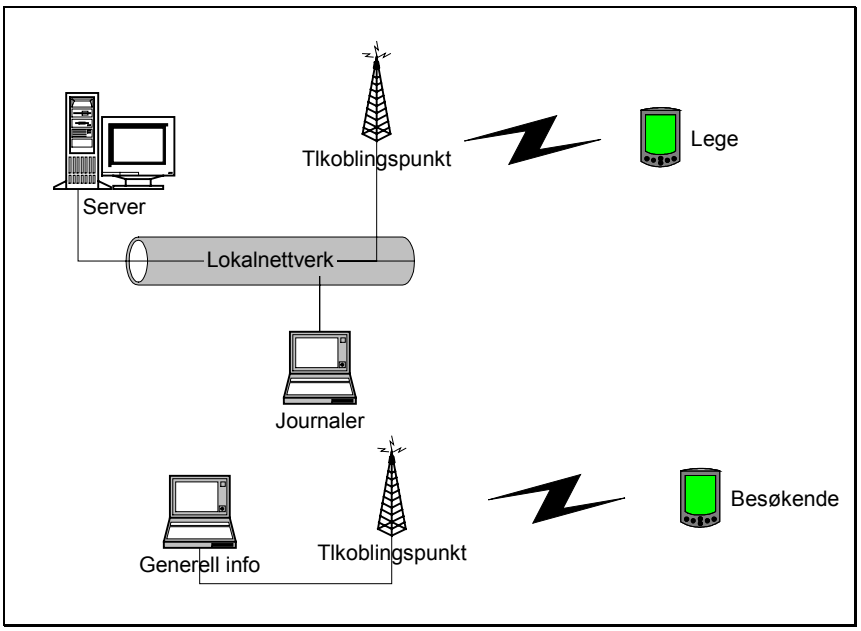
Scenario fire. En besøkende kommer inn på ett sykehus med sin PDA som støtter Blåtann. Sykehuset bruker Blåtann til å utveksle informasjon mellom pasientdatabaser og legenes PDAer. Det er ikke ønskelig at den besøkende skal få tilgang til sykehusets personregister og pasientjournaler med PDA'en sin. Den besøkende skal derimot få tilgang til andre tjenester som veibeskrivelse til avdelinger og informasjon, informasjon om parkeringstider og steder, hvor nærmeste blomsterbutikk er, informasjon for rullestolbrukere osv.

Forutsetter i scenario 4 at lokalnettverket til sykehuset er sikkert, og at pasientinformasjon ikke er tilgjengelig for besøkende, kun leger. Det må også forutsettes at ikke den besøkende har brukernavn og passord som gjør at han kan autentisere seg som en lege.



Figur 19 - Scenario 4. Besøkende med PDA på sykehus

For å få ovennevnte scenario til må man innføre sikkerhet på tjenestenivå. Slik dagens Blåtannprotokoll er vil ikke det være mulig å bruke samme Blåtannode for informasjonen fra eksemplene over, da man ikke kan styre tilgang på tjenestenivå for forskjellige tjenester. Det kan løses ved å la besøkende få tilgang til enkelte Blåtannoder, men ikke alle. Problemet med denne løsningen er at legene som skal ha sensitiv informasjon ikke kan bruke de samme nodene som besøkende da det automatisk blir full toveis kommunikasjon mellom Blåtannenheten til legen og Blåtannoden. Dette gjør at all informasjon må lagres minimum to steder. En database for legene, og en for besøkende.



Figur 20 - Mulig løsning på scenario 4 med dagens spesifikasjon

Løsningen i forrige avsnitt med flere databaser vil ikke være optimal da man nødvendigvis ikke ønsker at alle leger skal ha tilgang til alle journaler. Dette kan løses ved å innføre en sikkerhetssjekk på tjenestenivå. Legene og de besøkende kan da bruke samme noder da de besøkende ikke får tilgang til legenes tjenester. Journalene til avdeling B er implementert som en tjeneste og journalene for avdeling F er implementert som en

annen. Disse tjenestene kan være avgrenset slik at man må ha gjensidig tillit for å få tilgang til dem. Det vil si at en lege som har tillit på informasjon til avdeling B men ikke F vil kun ha tilgang til avdeling B sine journaler. Den besøkende har hverken autorisert seg, foruten som besøkende, eller blitt autorisert og har derfor ikke tilgang til de lukkete tjenestene.

Det er ikke sikkert at dette vil være en god nok løsning for datatilsynet med henblikk på personsikkerhet, men prinsippet for oppdeling av tilgangsnivå bør være godt nok for de fleste bedrifter.

Etter en mer grundig analyse av fellestrekk til disse fire scenarioene finner man følgende begrensinger :

Da det kun er enheter som kan autentiseres, og ikke brukerne, må det implementeres en sikkerhetsrutine i applikasjonslaget. Dette kan omgås ved å innføre sikkerhetsorganisatoren som er omtalt i kapittel 4. Applikasjonene kan da registrere hvilke brukere eller grupper som skal ha tilgang og hvilke som ikke skal ha tilgang i databasen til sikkerhetsorganisatoren. De forskjellige lagene vil da sjekke mot sikkerhetsorganisatoren, og denne vil da gi eller nekte tilgang til tjenestene eller applikasjonene.

Tilgjengelighet for bruk av applikasjoner med arv er ikke definert. Applikasjonene med arv kan ikke kontakte sikkerhetsorganisatoren direkte i noen tilfeller. Dette kan løses ved å lage ett adapter som kan gjøre spørringer til sikkerhetsorganisatoren fra applikasjonen. Denne løsningen er ikke definert i denne rapporten, da den vil være i applikasjonen.

Blåtann kan ikke løse alle problemene forbundet med trådløs kommunikasjon. Selv med innføringen av en ny sikkerhetsorganisator vil man ha svakheter. Noen av disse kan være :

- Det vil alltid finnes applikasjoner og brukere som vil klare å avlytte data som sendes over eteren. Løsningen på dette problemet er å sikre selve dataene som overføres på best mulig måte ved hjelp av kryptering på applikasjonssiden.
- Trådbunden kommunikasjon vil ikke opphøre selv om Blåtann innføres i de fleste elektriske enheter. Det er begrenset overføringshastighet, og ved mange enheter i samme geografiske område vil pakkekollisjonene øke. Ved økt kollisjonsrate blir båndbredden enda mer redusert.
- Da man ikke trenger å fysisk koble seg til en enhet ved trådløs kommunikasjon kan det være enklere for uautoriserte enheter å avlytte informasjon enn det ville ha vært hvis de måtte fysisk koble seg til. Blåtann bruker frekvenshopping til å redusere avlyttingsmulighetene.

Selv om trådløst nettverk er lettere å avlytte enn trådbundne, ligger sikring av data hovedsakelig i krypteringsrutiner. Vil noen avlytte data er det kun spørsmål om hvor mye prosessorkraft og tid som er tilgjengelig. Trådløs kommunikasjon kan gjøre selve avlyttingen av dataene enklere da man kun trenger å lytte på eteren, men datautstyr i seg selv lekker informasjon via elektromagnetisk stråling. Ved hjelp av ett modifisert TV og en retningsantenne kan man lett kopiere skjermbildet fra en PC som står bak en tykk vegg ett stykke unna.

På tross av avlyttingsmuligheten vil trådløs kommunikasjon gjøre at tilgangen til informasjon vil øke. Alt i alt vil nytten av en trådløs kommunikasjonsenhet som Blåtann være med på å nå det fremtidige økende kravet til informasjonstilgjengelighet.

Thomas Mueller sitt whitepaper [4] om sikkerhetsarkitekturen tar for seg omtrent den samme sikkerhetsorganisatoren som er foreslått i kapittel 4. Forskjellen mellom disse to sikkerhetsorganisatorene er at Mueller sin løsning kun tar for seg sikkerhetsnivå to. Dette gjør at man ikke kan få sikkerhet på enhetsnivå, men med hjelp av kun enkle rutiner vil man kunne implementere sikkerhetsnivå tre også.

Problemet med å innføre sikkerhetsorganisatoren fra kapittel 4 er at det vil ta lengre tid å autentisere en enhet ved oppsett. I tillegg til de vanlige sikkerhetsrutinene må man sjekke mot sikkerhetsorganisatoren som vist i figur 14. Det må opprettes en sikkerhetsdatabase, og denne må jevnlig vedlikeholdes. Dette er igjen prosess og plasskrevende i databrikker.

6 Konklusjon

Sikkerhetsaspektet til Blåtann er ikke fullt ut definert i spesifikasjonen. De tre sikkerhetsnivåene : ikke autentisering, autentisering på tjenestenivå og autentisering på enhetsnivå er ikke nok til å få en fullgod sikkerhetsregulering. Det er ønskelig å kunne skille tilgangen til de forskjellige tjenestene og brukerene av Blåtannenheter ikke bare på applikasjonslaget. Dette kan gjøres ved å implementere sikkerhetsorganisatoren fra kapittel 4. I denne modulen er det mulig å lagre informasjon om tilgang for enheter, tjenester og brukere til forskjellige tjenester og applikasjoner. Denne informasjonen kan det sjekkes mot i de forskjellige lagene i Blåtannspesifikasjonen [1]. Ved å bruke sikkerhetsorganisatoren vil man kunne innføre enda ett sikkerhetsnivå til tjenestene. Dette nivået er kalt autorisering og er en sjekk på enheter, tjenester eller brukere etter autentisering på tjenestenivå. En bruker kan være autentisert mot en tjeneste, men ikke nødvendigvis være autorisert til å bruke den. Sikkerhetsorganisatoren gjør det mulig med forskjellig tilgangsnivå til de forskjellige tjenestene og applikasjonene i en Blåtannenhet. For implementasjonsinformasjon for sikkerhetsnivå to, se Thomas Mueller sitt whitepaper på sikkerhetsarkitekturen til Blåtann [4].

Det er viktig å huske at Blåtann i hovedsak er en kommunikasjonskanal, og ikke ett sikkerhetssystem. Hensikten med Blåtann er å få trådløs kommunikasjon der man i dag bruker trådbunden. Dette gjør at man fremdeles må tenke sikkerhet når man lager applikasjoner, og ikke leve i den tro at Blåtann skal sikre hemmelig data mer enn trådbunden overføring gjør.

7 Litteraturreferanser

Publikasjoner :

1. Bluetooth SIG, (1999) The Bluetooth Spesification version 1.0b [Online 15. jan 2000] (http://www.bluetooth.com/developer/specification/specification.asp)
2. Bluetooth SIG, (1999) The Bluetooth Profiles versjon 1.0b [Online 15.jan 2000] (http://www.bluetooth.com/developer/specification/specification.asp)
3. Massey, J.L. Rueppel, R.A. (1985) Linear Chipers and Random Sequence Generators with multiple clocks" Advances in Chryptology: Proceedings of EUROCRYPT 84, Springer-Verlag
4. Muller, T. (1999) Bluetooth Security Architecture (Version 1),[Online 22.04.2000] (http://www.bluetooth.com/developer/whitepaper/whitepaper.asp)
5. Muller, T. (1999) Bluetooth Security Proceedings Bluetooth'99, London June 1999
6. Persson, J. (1999) Bluetooth Baseband Security Consept Proceedings Bluetooth'99, London Jun 1999
7. Scneier, B. (1996) Applied Chryptography (Second edition), John Wiley & Son Inc, ISBN : 0-471-11709-9
8. Stallings, W. (1995) NETWORK AND INTERNET SECURITY – Principles and Practice, Prentice-Hall International, ISBN : 0-13-180050-7

Websteder :

9. IEEE sin ofisielle internettside (www.ieee.com) (10. feb 2000)
10. Blåtanns ofisielle internettside (www.bluetooth.com) (15. jan 2000)
11. Cap Gemini sine sider på Sikkerhet : (http://www.capgemini.no/tjenester/veilhefter/datasikkerhet/index.asp) (15. april 2000)
12. EFN's (Elektronisk Forpost Norge) side om Kryptering : (http://www.efn.no/faktfil-om-kryptografi.html) (16. mai 2000)
13. Handsfreeløsning fra Ericsson : (http://www.bluetooth.com/product/qualified_p/products.asp?action=consumer&type=audiodevices) (20. mai 2000)
14. Kabelerstatning fra TDK : (http://www.tdksys.com/bluetooth/info.html) (20. mai 2000)
15. Trådløslaninfo fra 3 COM : (http://www.3com.com/wireless/pdf/bluetooth_position.pdf) (20. mai 2000)

Andre generelle linker til Blåtann :

16. Palo Pacific Technology sin samleside for Blåtanninfo : (http://www.palopt.com.au/bluetooth) (20. mai 2000)
17. Ericsson sin Blåtannside http://bluetooth.ericsson.se/bluetooth/ (15. mai 2000)
18. http://www.bluetoothcentral.com (25. mai 2000)
19. Intel sin Blåtannside http://www.intel.com/mobile/bluetooth/index.htm (20. mai 2000)
20. Motorola sin Blåtannside http://www.motorola.com/bluetooth/ (20. mai 2000)
21. IIR Telecom & Technology (Seminaroversikt) http://www.iir-bluetooth.com/bluetooth.htm (25. mai 2000)
22. Widcomm sine Blåtannprodukter og informasjon : (http://www.widcomm.com/index.htm) (26. mai 2000)
23. Toshiba sine fremtidsvisjoner om Blåtann : (http://www.toshiba-europe.com/computers/tnt/bluetooth.htm) (20. mai 2000)
24. Ensuretech lager sikkerhetsløsninger bland annet til Blåtann : (http://www.ensuretech.com) (10. mai 2000)
25. En uavhengig Blåtannside med eksempler på produkter : (http://www.bluetooth.net) (11.feb 2000)
26. Atmel er produsent av Blåtannbrikker : (http://www.atmel.com/bluetooth) (18. mai 2000)

8 Ordliste

SIG : (Special Interest Group) – Den offisiellle Blåtannorganisasjonen.
ESCE : (External Security Control Entity) – En ekstern sikkerhetsmodul.