Roaming mellom WLAN og UMTS: 3G håndholdt enhet som helsekommunikator

av

Pavneet Singh
Kjetil Ødegaarden

Hovedoppgave til mastergraden i informasjons- og kommunikasjonsteknologi

Høgskolen i Agder
Fakultet for teknologi
Grimstad, 29. mai 2006

SAMMENDRAG

Oppdragsgiver i denne oppgaven er CARDIAC AS. CARDIAC leverer i dag en løsning til St. Olavs Hospital i Trondheim som tar seg av all internkommunikasjon på sykehuset. Dette innbefatter bl.a. telefoni, pasientsignalvarsling, tilkalling og alarmering. Helsepersonell bærer flere kommunikasjonsenheter med seg mens de er på vakt (trådløs IP-telefon, jobb-/personlig- mobiltelefon, p-søker). IP-telefonene betjenes med et enkelt menysystem for å bekrefte/avvise oppdrag med flere funksjoner avhengig av rollen til personellet.

Den store ulempen med IP-telefonene er at de er avhengig av å ha kontakt med sykehusets WLAN for å fungere. Dersom de ikke har kontakt blir oppdrag sendt ut som SMS til mobiltelefon og svar må dermed manuelt skrives inn og sendes som en vanlig SMS tilbake. I dag består store deler av St. Olavs Hospital av gammel bygningsmasse uten trådløs dekning og bruken av SMS er derfor omfattende.

I denne oppgaven skal vi se på mulighetene for å løse problemstillingen ved å bruke en 3G mobil enhet med WLAN-støtte for å erstatte IP-telefonene. Denne enheten skal ha de samme funksjonsmulighetene som IP-telefonene, men kunne fungere i gammel bygningsmasse ved å bytte over til UMTS-mobilnettverk (3G) og tilbake til WLAN uten at bruker trenger å bytte manuelt (sømløs roaming). Første del av oppgaven belyser teknologier som kan benyttes i forhold til å oppnå sømløs roaming for en mobil enhet. Andre del av oppgaven foreslår et teoretisk design og viser gjennomføringen av konseptuelle tester (Proof of Concept).

Teknologier som kan benyttes for å oppnå roaming befinner seg på ulike nivåer i OSI-modellen. Roamingen kan løses på linklaget ved hjelp av UMTS, UMA eller IMS, men alle disse vil være avhengig av implementeringer fra operatørsiden. Det kan oppnås en løsning via nettverkslaget ved hjelp av MIP. MIP gir mobilitet over ulike nett og kan benyttes sammen med mobilitetsløsninger for applikasjonslaget (SIP). Prosjektgruppen mener den beste løsningen vil være å benytte seg av mulighetene som finnes i MIP og SIP for meldingsutveksling og mobilitetshåndtering. For at løsningen skal gi de samme mulighetene som dagens to enheter (mobiltelefon og IP-telefon) vil det være viktig å ivareta sikkerhet og tilgjengelighet. Algoritmer for å ivareta dataintegritet (SHA-1) og konfidensialitet (AES) har blitt valgt. I forhold til at enheten skal være tilgjengelig og dermed kan kontaktes gjennomgås adressering i Internett og mobile nett.

To caser er blitt brukt som utgangspunkt for arbeidet i oppgaven: Portøroppdrag og NurseCall. Det teoretiske designet og PoC-testingen er bygget opp rundt disse to casene. PoC-testingen tar for seg meldingsutvekslingen mellom en mobil enhet i 3G-nettet og en PC i WLAN. Testene har vist at denne kommunikasjonen er mulig og at enheten vil være tilgjengelig i det mobile nettet. Sikkerhet ivaretas i PoC ved AES-kryptering av selve meldingsinnholdet. For å bevare dataintegritet benyttes det en sjekksum generert ved hjelp av SHA-1 algoritmen. Målinger gjort av RTT viser at transmisjonsforsinkelsen i GPRS og UMTS vil være liten.

Roaming mellom WLAN og UMTS vil kunne gjennomføres med dagens teknologi, men vil være en kompromissløsning avhengig av hvilke teknologier som velges, og i

hvilken sammenheng det benyttes. Helsepersonell vil kunne spare tid ved å benytte seg av predefinerte svar på SMS der de tidligere måtte taste inn svarmeldinger manuelt. Dersom løsningen implementeres fullstendig ved innføringen av en håndholdt enhet vil personellet spare tid, og oppnå enklere arbeidsflyt i form av at de kun trenger å forholde seg til en enhet. I tillegg til tidsbesparelsen vil systemet spare kostnader da meldingene som sendes er små og samlet vil prisen på datatrafikk i mobilnettet være mindre enn kostnadene ved SMS-bruk. Systemet vil også kunne sammen med en posisjoneringsløsning i mobilnettet kunne benyttes i andre former for virksomheter der flåtestyring gir fordeler.

FORORD

Denne oppgaven avslutter masterutdanningen i Informasjon og Kommunikasjonsteknologi ved Høgskolen i Agder. Oppgaven ble gitt av CARDIAC AS og utført i deres lokaler i Porsgrunn. Oppgavens varighet har vært fra januar 2006 til juni 2006.

Vi vil gjerne takke vår veileder på HiA, Magne Arild Haglund, for veiledning og innspill under prosjektperioden. Samtidig vil vi takke CARDIAC AS ved Vidar Undem og Sigurd Juvik for oppgaven og veiledning underveis.

Porsgrunn, 29.05.06

Pavneet Singh

Kjetil Ødegaarden

INNHOLDSFORTEGNELSE

FIGURLISTE

TABELLISTE

1 INTRODUKSJON

St. Olavs Hospital i Trondheim har en visjon om integrerte sykehusløsninger med pasienten i fokus. Det er stadig etterspørsel etter videre forbedring for å effektivisere arbeidet på sykehuset, men dette er begrenset av stramme budsjetter. Effektivisering gjøres ved å investere i nytt moderne utstyr som øker kapasiteten uten å øke bemanningen. Et slikt produkt er IMATIS som leveres av CARDIAC AS (Computer Aided Research, Development, Instrumentation and Control) i Porsgrunn. Dette produktet tilbyr en løsning som dekker mange av kommunikasjonsbehovene for sykehusets ansatte og pasienter.

1.1 CARDIAC

CARDIAC er et teknologiselskap basert i Porsgrunn som leverer løsninger til industriell og medisinsk IT. CARDIAC leverer produkter som bygger direkte på deres hovedprodukt, IMATIS Meldingstjener (Integrert Modulbasert Administrativt Teknisk Informasjons System). IMATIS fungerer som et felles rammeverk og plattform for sanntids datainnsamling og datahåndtering. IMATIS muliggjør at alle deler i en organisasjon skal kunne agere på hendelser i sanntid. Dette gjøres ved å integrere mobile enheter som IP-telefoner, mobiltelefoner og PDAer sammen med stasjonære enheter i en felles mellomvare (IMATIS).

CARDIACs største helseprosjekt pr. i dag er St. Olavs Hospital i Trondheim der de er svært sentrale i ombyggingen av infrastrukturen.

1.2 Problemstilling

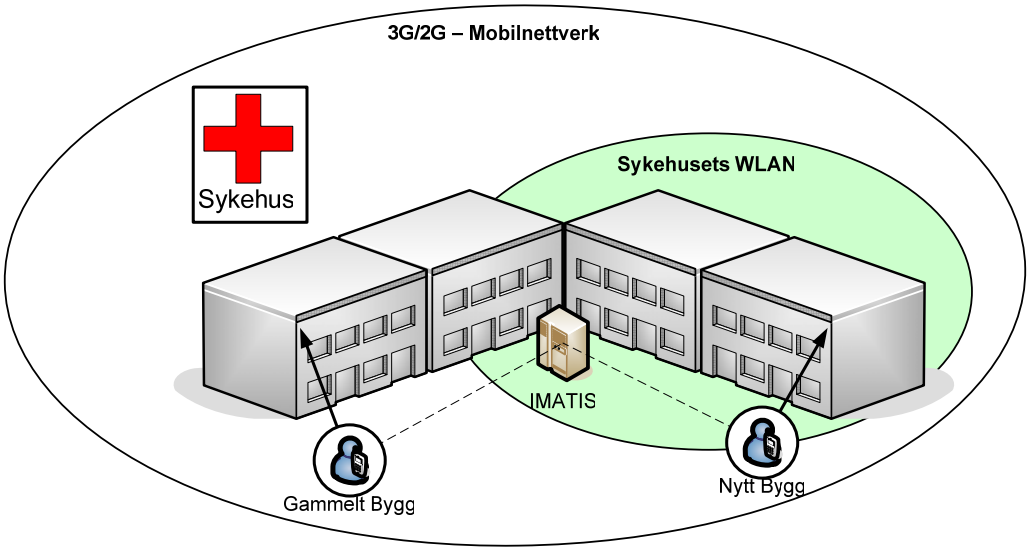
På sykehus i dag forholder helsepersonell seg til ulike systemer i forhold til trådløs informasjonsflyt. Det er ikke uvanlig at en lege må ha flere ulike kommunikasjonsenheter, som eksempelvis trådløs IP-telefon, p-søker(e), PDA og mobiltelefon. På St. Olavs Hospital i Trondheim er de ulike tjenestene (telefoni, pasientsignalvarsling, tilkalling og alarmering mfl.) samlet på én IP-telefon gjennom IMATIS. De trådløse IP-telefonene gir muligheten for helsepersonell å reagere på alarmmeldinger effektivt gjennom et enkelt menysystem. Ulempen med denne løsningen er at den ikke fungerer utenfor sykehusets trådløse dekningsområde. Dette er en stor ulempe i forhold til effektivisering av hverdagen da store deler av St. Olavs Hospital består av gammel bygningsmasse der det ikke finnes WLAN. I dagens system benyttes mobiltelefoner og SMS for å nå personell som befinner seg utenfor den trådløse dekningen.

Oppgaven går ut på å undersøke hvorvidt dagens 3G (UMTS) mobiltelefoner vil kunne benyttes i stedet for IP-telefoner (WLAN) som håndholdte enheter, med høy nok tilgjengelighet og sikkerhet på og rundt selve sykehuset ved bruk av WLAN eller UMTS.

Det må utarbeides en analyse av mulighetene for sømløs roaming for en håndholdt enhet mellom sykehusets WLAN og utenforliggende UMTS-/GPRS-nett[1]. Det vil si hvorvidt det går for en enhet å bytte mellom to nettverk uten brukerinteraksjon, samtidig som forbindelsen ikke mistes.

Oppgaven begrenses til å omfatte meldingsutveksling, og ikke IP-telefoni. Det legges opp til en teoretisk analyse av problemstillingen, med dette menes arkitektur, mobile enheter, brukerscenarioer, sikkerhet, og modellering og utarbeiding av designet for løsningen.

Resultater skal vises gjennom en rapport og det skal lages Proof of Concept (PoC)-tester basert på valg gjort i den teoretiske analysen og modelleringen. Modelleringen skal vise deler av funksjonaliteten i systemet. Det tas sikte på at eventuell utvikling av PoC utføres i Java eller Visual Basic.

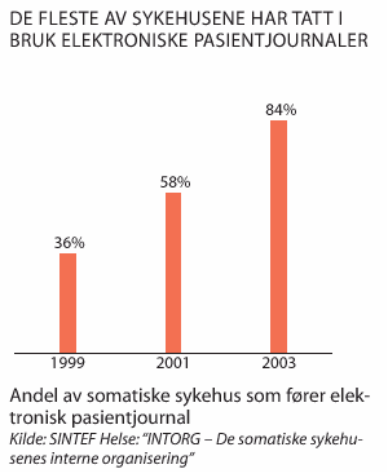


Figur 1 - Overordnet oversikt over problemstilling [Egen figur]

1.3 Bakgrunn og Problemområde

Sykehus i Norge er i dag under stadig utvikling der det legges stor vekt på at det skal implementeres elektroniske løsninger for effektivisering av arbeidet. Følgende er hentet fra Moderniseringsdepartementets hjemmesider [1]:

[1] UMTS og GPRS fungerer på samme måte i kjernenettet (kap 2.2.1). Heretter betegnes som UMTS.



Figur 2 - Graf som viser overgang til elektroniske pasientjournaler [1]

"IT for en enklere helse- og omsorgssektor

Hvert år bruker det offentlige om lag 140 milliarder kroner på helse, pleie og omsorg

…

Regjeringen ønsker å bruke informasjonsteknologi for å gi brukerne av helsetjenester et bedre tilbud. Dette skal ikke gå på bekostning av personvernet

…

Papir- og IT-løsninger eksisterer ofte side om side, og det gir dobbeltarbeid. For å bøte på dette, skal papirer fjernes der det finnes elektroniske løsninger. Pasientjournalen er kjernen i informasjonsflyten i helsetjenesten. Innføring av elektronisk pasientjournal i allmennlegetjenesten har gitt store gevinster. Rutinearbeid som skriving av resepter og sykemeldinger går raskere. I tillegg har journalen blitt mer lesbar og legene finner journalen lettere når de trenger den.

…"

En del av CARDIACs IMATIS-løsning som leveres til St. Olavs baserer seg på bruk av IP-telefoner som kommunikasjonsenhet for helsepersonell lokalt på sykehuset. IP-telefonene gir muligheten for helsepersonell til å reagere på meldinger effektivt gjennom et enkelt menysystem. Telefonene kobles direkte til det trådløse datanettverket på sykehuset. På denne måten er all internkommunikasjon på sykehuset uavhengig av operatør og kostnader. De trådløse IP-telefonene som benyttes av helsepersonell kobler seg til datanettverket gjennom trådløse aksesspunkter som er plassert rundt på sykehuset (WLAN). Disse telefonene er avhengig av å være i kontakt med det trådløse nettverket på sykehuset for å fungere.

Utover vanlig kommunikasjon, brukes IP-telefonene til å håndtere bl.a. *NurseCall* og *Portøroppdrag*. Ved pasientalarm/oppdrag, blir det sendt meldinger til telefonene som må kvitteres av personellet. Denne kvitteringen foregår gjennom et enkelt menysystem på IP-telefonene. Grunnet ny og gammel bygningsmasse, er det ikke dekning på hele sykehuset, det oppstår da problemer når IP-telefonene er utenfor WLAN-dekning. I tilfeller der en enhet befinner seg utenfor dekning, sendes det en SMS til personellets vanlige mobiltelefon som da må leses og manuelt kvitteres. Menyen som vanligvis er tilgjengelig på IP-telefonene er ikke tilgjengelig på mobiltelefoner, og det må da svares med en vanlig tekstmelding. Ved en alarmmelding, holder det at brukeren sender svar med én bokstav/tegn for å kvittere for at meldingen er mottatt. Et *Portøroppdrag* kvitteres ved å manuelt kopiere Meldings ID og respons (Godta/Avvis) i en ny melding og svare tilbake.

I tillegg til å samle all intern mobilkommunikasjon med IP-telefonene, leverer IMATIS en felles plattform for all informasjon som går på sykehusnettverket (alarmer, internmeldinger, TV/Video/radio, Internett, m.fl). CARDIAC ønsker å utvide IMATIS slik at tjenestene *NurseCall* og *Portøroppdrag* skal være tilgjengelig gjennom 3G-

mobilnettet når det ikke er dekning av WLANet. På denne måten vil all informasjon bli samlet over IP og kan håndteres fortløpende i IMATIS Meldingstjener.

Det er tenkt en løsning som baserer seg på en moderne mobiltelefon med støtte for WLAN og 3G, slik at det ikke lenger skal være nødvendig å ha med seg to kommunikasjonsenheter for helsepersonell mens de er på jobb. På denne måten skal all kommunikasjon foregå over IP, og brukeren skal kunne forflytte seg uten å ta hensyn til at den mobile enheten bytter nettverk. Dersom man holder på med en aktiv dataoverføring mens det byttes nett, skal ikke forbindelsen mistes Dette omtales som sømløs roaming.

Mobilitetshåndtering er en betegnelse på det å kontrollere hvilke nettverk en terminal eller en bruker er koblet til i et gitt tidspunkt. I dette ligger også det å oppdage nye nettverk og koble til disse. I utgangspunktet er WLAN og 3G to forskjellige nettverkstyper som ikke kommuniserer direkte med hverandre. Alt som foregår på et lokalt WLAN skjer uten noen form for tjenesteleverandør involvert. 3G er tilgjengelig gjennom mobilnettet og krever at man leier lufttid fra en mobiloperatør. Når en enhet flytter seg til et nytt nett må det autentiseres. Dette skaper komplikasjoner siden enheten vil bli tildelt forskjellig adresse i WLANet og mobilnettet. Et annet problem er at enheten må koble ned på det gamle nettet, for så å koble opp på det nye nettet. Dette medfører at enheten er uten kontakt i en gitt periode. Målet med sømløs roaming er at denne autentiseringen og oppkoblingen skal skje automatisk uten at enheten mister sin hjemmeadresse. Med sømløs roaming oppnår man at helsepersonell kan bevege seg utenfor sykehusets trådløse dekningsområde, men likevel ha mulighet til å kommunisere med IMATIS Meldingstjener. På denne måten vil det være mulig å få tilgang til tjenestener som ellers leveres på sykehuset mens personell er i bevegelse. Slik kan sykehuset ha bedre kontakt med sine ansatte og dermed sikre raskere responstid og bedre tjenester for pasienter.

1.4 Avgrensninger og Forutsetninger

CARDIACs eksisterende meldingssystem, IMATIS, skal ikke endres på, men bygges videre slik at det kan benyttes utenfor dekningen til sykehusets WLAN.

En side ved valg av håndholdt enhet det ikke skal tas hensyn til, er brukervennlighet og opplæring. Det vil sannsynligvis være en overgang for helsepersonell fra de nåværende IP-telefonene som er tilgjengelig til en avansert håndholdt enhet med operativsystem og ikke-standard tastaturløsning. For å implementere dette, ville det vært nødvendig med konkrete brukerundersøkelser om hva slags rutiner og krav helsepersonell har i forhold til en slik enhet. Ved en fullstendig implementering av en ferdig utvidelse av IMATIS systemet ville dette vært rimelig sentralt. Siden målet med denne oppgaven først og fremst er en PoC, faller dette utenfor vår oppgave.

Med hensyn til sikkerhet, vil vi i oppgaven vår konsentrere oss hovedsakelig om datasikkerhet. Det blir ikke tatt hensyn til problematikken ved tap av utstyr, det vil si at det ikke blir tatt hensyn til å sikre tilgang til informasjon/meldingstjenesten (autentisering av brukeren) hvis en enhet blir frastjålet. Det antas derfor at enheten befinner seg hos den rette brukeren til enhver tid.

Helsevesenet stiller strenge krav til personvern og sikkerhet i form av sikring av personlig informasjon. Fra sykehusets og pasientens side, er dette noe av det viktigste når det først er snakk om informasjonsdeling mellom sykehuset og andre parter. Ved å implementere håndholdte enheter som får tilgang til sykehusets nettverk fra utsiden av nettverket, åpnes muligheten for et sikkerhetsbrudd. Meldingene fra IMATIS Meldingstjeneren til brukere er alarmmeldinger som forteller kort om det er alarm på et rom eller om det er portøroppdrag som skal gjennomføres. Det sendes ikke store mengder sensitiv data fra meldingstjeneren slik som medisinske journaler med personopplysninger. Derfor vil det ikke være nødvendig for oss å sette oss inn en problemstilling som dekker alle helsevesenets krav. Ved en senere implementering og mulig utvidelse der sensitiv informasjon vil være tilgjengelig, vil denne problemstillingen være aktuell. Derimot kan det sendes med pasientnavn og fødselsnummer ved portørmeldinger. Dette medfører at vi må ta hensyn til sikring av selve meldingsinnholdet

Ved å implementere støtte for håndholdte enheter utenfor sykehusnettverket i IMATIS meldingstjener, vil det sannsynligvis være snakk om en selgbar utvidelse. Da vil den praktiske gjennomførbarheten og kostnadseffektiviteten være sentrale faktorer ved en endelig løsning. I den forbindelse kommer det mulige prissammenligninger mellom håndholdte enheter, teleoperatører, internettleverandører og det som ellers må til av ekstra utstyr og opplæring med i bildet. Siden vi skal gjennomføre en PoC, vil det ikke være nødvendig for oss å se på hvordan dette systemet skal selges sammen med, eller i tillegg til, resten av IMATIS.

2 TEORI

Dette kapittelet gjennomgår ulike teknologier som må berøres i en roamingløsning. I tillegg gjennomgås fremtidig utvikling av valgte nettverksteknologier.

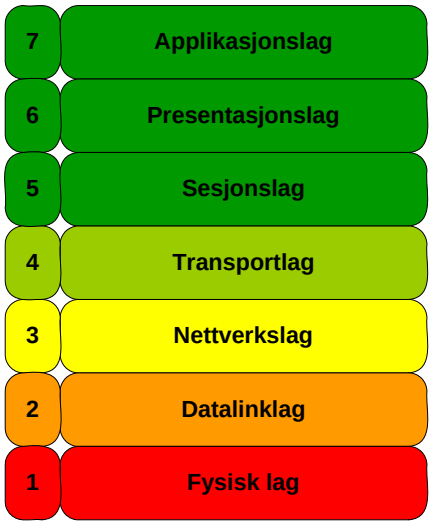
2.1 Introduksjon

Problemstillingen med å få til sømløs roaming mellom WLAN og mobilnett på en håndholdt enhet er ikke ny, men har vært vanskelig å gjennomføre siden den teknologiske utviklingen ikke har tilrettelagt dette. I dag har enheter som bærbar PC, PDA og mobiltelefoner fått moduler som muliggjør kommunikasjon på begge type nett i en enhet. En annen begrensende faktor har vært, og er fortsatt, den trege hastigheten over mobilnettet og kostnadene knyttet til overføring av store datamengder.

Behovet for en slik løsning har ikke vært stort siden enhetene som tillater slik kommunikasjon ikke er praktiske å ha med å gjøre pga. strømforbruk, manglende mobilitet eller rett og slett at det har vært enklere å bruke flere enheter. Etter hvert som teknologien har utviklet seg gjennom nye, mindre komponenter for WLAN- og mobilnett-kommunikasjon samt utviklingen av 3G-nettet, har potensialet for kostnadsbesparinger og effektivisering av arbeidsoppgaver begynt å vise seg.

Lanseringen av 3. generasjonens mobilnett (3G) har åpnet mulighetene for utvikling av en roamingløsning mellom WLAN og mobilnett. I dag er det ikke direkte støtte i WLAN eller mobilnettet for denne roamingen. Det er derfor tatt i bruk løsninger som vil holde ut til standarder er definert og tatt i bruk for enheter/moduler, WLAN og mobilnett.

De eksisterende løsningene befinner seg på forskjellige lag i OSI-modellen (Figur 3), alle med sine fordeler og ulemper. Sammen med det rent tekniske aspektet av å kunne roame sømløst mellom nettene, finnes det en rekke sikkerhetsmekanismer som må innføres. Mekanismene som er aktuelle er avhengig av hvilket OSI-lag man befinner seg på.



Figur 3 - OSI-modellen [Egen figur]

Utover kommunikasjonsteknologien som støttes, vil en løsning være avhengig av hva slags håndholdt enhet som blir brukt. Det er da aktuelt å se på hva slags muligheter som finnes i forhold til operativsystem og programvare. Det er også aktuelt å se på implementeringen av sikkerhet på enheten i forhold til meldingsinnhold.

IMATIS Meldingstjener sender ut meldinger som skal tolkes og behandles i forhold til den håndholdte enheten. Det er derfor viktig å se på hvordan IMATIS håndterer disse meldingene og hvordan de blir vist på den håndholdte enheten.

2.1.1 Mobilitet

Når vi snakker om mobilitet er det viktig å skille mellom roaming og sømløs roaming. Roaming er en betegnelse på det å bevare tilkobling når lokasjonen endres fra et nettverk til et annet. Roaming kan brukes som en generell betegnelse, men det mest kjente eksempelet er når en bruker med mobiltelefon forflytter seg slik at den kobler opp mot en ny basestasjon. Det forekommer også når operatøren ikke lenger kan tilby sitt eget nett. Dette vil typisk være når brukeren befinner seg i et annet land, eller et område som operatøren ikke har bygget ut tilstrekkelig. For at det likevel skal være mulig for brukeren å ha kontakt, må operatøren ha en roamingavtale med en annen operatør som tilbyr dekning i det aktuelle området.

Sømløs roaming er også kjent som *handover*. Dette innebærer at en bruker ikke mister forbindelse mens den mobile enheten roamer mellom basestasjoner eller nettverk. Hva dette betyr i praksis er at en bruker kan ha en aktiv forbindelse som overfører data / tale og samtidig flytte seg uten brudd på kommunikasjonsstrømmen. Det vil si at nett byttes, og kommunikasjonen opprettholdes uten at brukeren trenger å gjøre noe for at det skal skje. Dette benyttes i mobilnettet.

Handoff er et annet begrep som også dukker opp i forbindelse med roaming. I motsetning til *handover*, vil det oppstå et brudd i kommunikasjonen når en bruker bytter mellom flere basestasjoner. Brukeren vil ikke ha behov for å gjøre noe for at enheten bytter til neste bastestasjon, men det vil være nødvendig å initiere eller fortsette kommunikasjonen på nytt. Dette benyttes i WLAN.

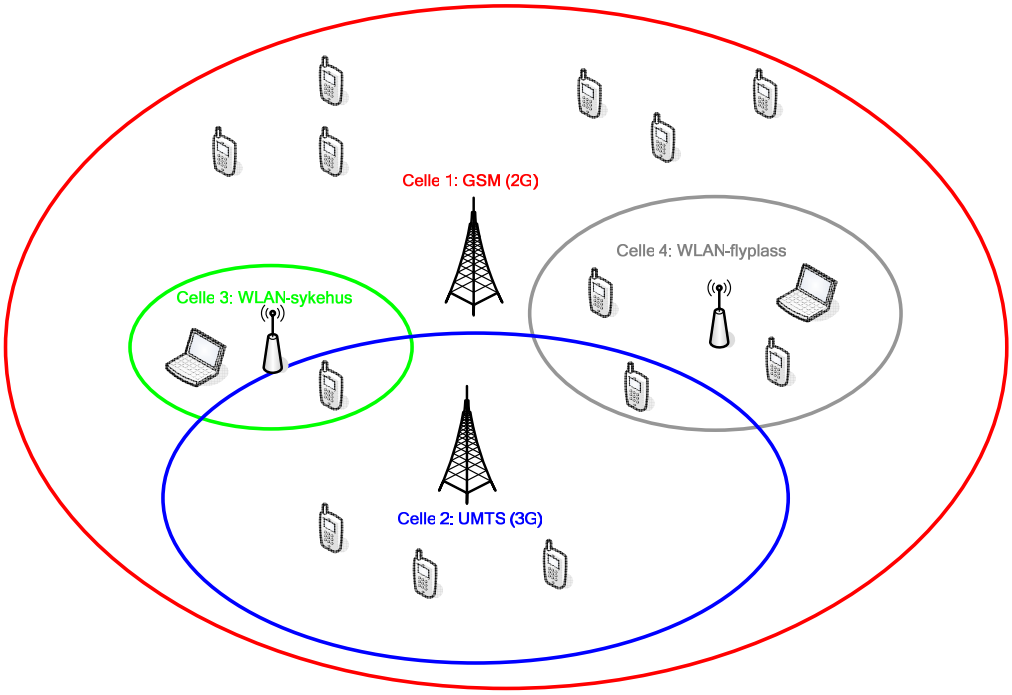
I tillegg til mobilnettets behov er det de siste årene blitt stadig mer aktuelt med roaming mellom forskjellig typer nettverk der det tidligere dreide seg om forflyttning mellom mobilnett. Det er vanlig i dag at WLAN hotspots dukker opp stadig flere plasser. Dette kan være på hoteller, flyplasser, offentlige steder, kontoret og i hjemmet. På grunn av denne stadige utbyggingen, blir roaming mellom mobilnett og andre type trådløse nett en stadig mer aktuell problemstilling. Å gjennomføre sømløs roaming mellom WLAN og mobilnett betyr at en enhet vil alltid kunne bytte til det beste kommunikasjonsalternativet under bevegelse. Slik vil det være mulig å kombinere dekningsområdet til mobilnettet og de store hastighetene i WLAN.

Roaming og mobilitet bygger på en enkel arkitektur av systemet bestående av basestasjoner og klienter. Selve ordet mobilitet brukes ofte om entiteten som beveger på seg, i de fleste tilfellene personen som bruker enheten. Vi kommer til å bruke ordet mobilitet om det å flytte seg fra ett nett til et annet uavhengig om det er enhet eller person som flytter seg.

En basestasjon er et tilkoblingspunkt som samler sammen flere mobile enheter og gir disse aksess. Eksempler på basestasjoner kan være aksesspunkter i WLAN eller radiotårn i GSM/UMTS. Basestasjonen står på et fast sted og gir aksess i et dekningsområde. Disse dekningsområdene kalles ofte celler. Vi definerer roaming som evnen til å bevege seg mellom disse cellene uavhengig av nettverkstyper. Disse cellene kan være innenfor hverandre som for eksempel i vårt tilfelle et WLAN-nett innenfor et UMTS-nett, roaming vil da være muligheten til å bytte mellom disse. For at denne

roamingen skal foregå sømløst må denne byttingen foregå automatisk, uten brukerinteraksjon.

Figur 4 viser en generell fremstilling av celler med tilhørende aksesspunkter og klienter.



Figur 4 - Aksesspunkter og klienter i WLAN og mobilnettet [Egen figur]

2.1.2 Mobile nettverk

Andre generasjons mobilsystemer (GSM) benytter en digitalisert talebånd og et kontrollsignal for å overkomme de analoge begrensningene som fantes i 1. generasjons mobilsystem [2]. General Packet Radio Service (GPRS - 2,5G) utvider GSM ved å tilby et pakkesvitsjet domene som muliggjør IP-trafikk [3]. GPRS utgjør, sammen med Enhanced Data rates for GSM Evolution (EDGE - 2,75G), et mellomsteg på veien fra 2G (GSM) til 3G (UMTS) [4]. Ved å benytte andre modulasjonsteknikker (Vedlegg B) kan EDGE oppnå større hastigheter enn GPRS alene. Avhengig av implementering kan EDGE ansees som både 2G og 3G på grunn av hastigheten det kan tilby [5].

2G kan sammen med GPRS og EDGE tilby både tale- og data- trafikk, men det vil gå på bekostning av hastigheten. 3G benytter seg av W-CDMA i Europa og av CDMA-2000 i Amerika [5]. CDMA multipleksing ligger til grunn for begge systemer og er en teknikk som gir brukere mulighet til å forbli koblet til uten å måtte dele lufttid som tilfellet var med det TDMA-baserte GSM-systemet. I tillegg til dette tilbyr 3G større hastigheter; >384 kbps. UMTS er den europeiske 3G-standarden [5] og kan ved bruk av W-CDMA tilby hastigheter opp til 1920 kbps.

Tabell 1 - Oversikt over datarater i mobile nettverk [6]

Mobilsystem	Datarate (Hastighet)	Modulasjon
GSM – (2G)	9.6 kbit/s	GMSK
GPRS – (2,5G)	114 kbit/s	GMSK
EDGE – (2,75G)	384 kbit/s	GMSK, 8-PSK
UMTS – (3G)	1920 kbit/s	Q-PSK

2.2 Mobilitet på Linklaget

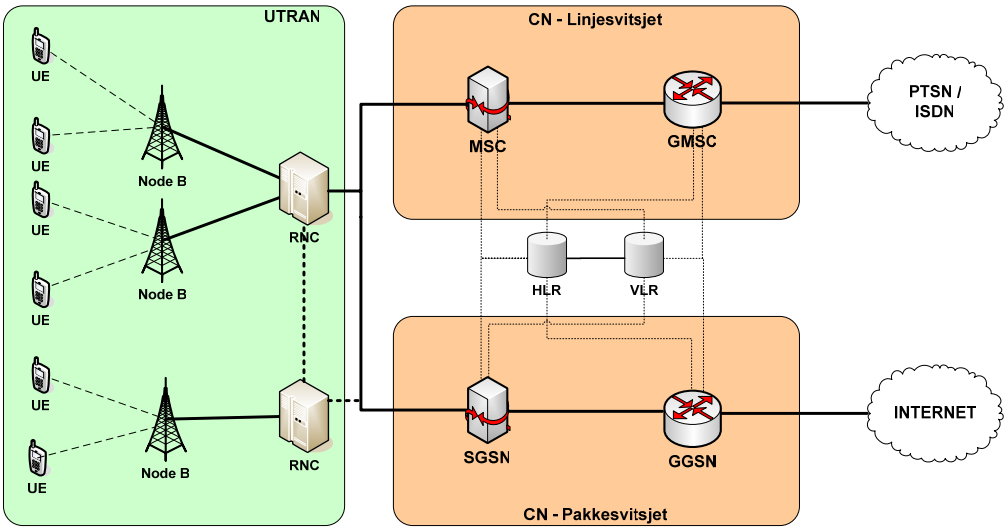
2.2.1 UMTS

IMT-2000 (International Mobil Telecommunications-2000) er en felles betegnelse for alle godkjente tredje generasjons mobilnettverksstandarder. Alle disse standardene er delt i 5 kategorier i IMT-2000; IMT-DS (Direct Spread), IMT-MC (Multi-Carrier), IMT-TD (Time-Division), IMT-SC (Single Carrier), IMT-FT (Frequency Time) [7]. Den ledende og aksepterte standarden i Europa er Universal Mobile Telecommunications System (UMTS) [8], denne ligger under kategorien IMT-DS.

UMTS arkitektur
Første utgave av UMTS blir kalt UMTS R3 (også kjent som Release '99). Den bruker W-CDMA som underliggende standard og blir standarisert av 3GPP[2]. Nyere håndholdte enheter støtter i dag UMTS og nettverkene blir gradvis bygget ut over hele Europa. UMTS er en viderføring av GSM med GPRS og gir større hastigheter og inneholder flere sikkerhetsfunksjoner enn GSM. Det beste eksempelet på forbedret sikkerhet er at UMTS autentiserer både nettverk og bruker i motsetning til GSM som kun autentiserte brukeren [10].

GPRS blir ofte kalt 2,5G [3] og er en pakkesvitjset teknologi. Dette betyr at mange brukere kan dele samme kommunikasjonskanal. Dette medfører at tilgjengelig båndbredde kan bli dedikert til brukere som sender ved et gitt tidspunkt. Dette gir brukere forbedret Quality of Service (QoS). Fremtidige versjoner av UMTS, vil ta store steg mot foreningen av UMTS og WLAN, dette ved å innføre IP-baserte tjenester over UMTS (kap 2.6.1)

[2] Third Generation Partner Project - http://www.3gpp.org/



Figur 5 - Overordnet arkitektur av UMTS Release3 [9]

- UMTS-Terminal (UE, User Equipment)
  Den mobile enheten med USIM brikke, som kommuniserer mot UMTS. Telefon, PDA, laptop osv.

- Basestasjon (Node B)
  Den fysiske enheten for radiotransmisjon/mottakelse med UE.

- Kontrollere (RNC, Radio Network Controller)
  Kontrollerer basestasjonene og tar seg av radioressursene i UTRAN. Kommunikasjonsleddet mellom UTRAN og CN (Core Netork – kjernenettet)

Dataoverføring i UMTS

En dataoverføring fra eksternt nett i UMTS foregår over den pakkesvitsjet del av kjernenettet. Det vil si at selve den fysiske overføringen foregår over GPRS-nettverket. Brukerplanet i denne forbindelsen kalles for en PDP-kontekst (Packet Data Protocol), som igjen deles opp i to deler: RAB-bærer (Random Access Bearer) og GTP-tunnel (GPRS Tunneling Protocol). Under brukerplanet, finner man kontrollplanet, som tar seg av selve oppkoblingen til PDP-konteksten [9].

Oppsett av dataoverføring i UMTS:

UE utfører en *GPRS attach* prosedyre, denne prosedyren forteller SGSN at UEen vil etablere en forbindelse. Deretter må det foregå en autentisering av UEen hos SGSN, som da oppdaterer lokasjonen av UE til HLR. HLR svarer med abonnementsdata til UE. Det er dermed opprettet en MM-kontekst (Mobility Management) for abonnenten. Etter dette, vil SGSN svare UE med en melding som inneholder P-TMSI (Packet – Temporary Mobile Subscriber Identity) parameteren. Denne brukes for å identifisere abonnenten [9].

Koble opp overføringstunnelen:
For at datapakker (IP-trafikk) skal sendes, må *PDP Context Activation* utføres først. Denne foregår mellom UE og SGSN, som da setter opp RAB-tunnelen på aksessnettet, og GTP-tunnelen til GGSN. GGSN er direkte koblet til det eksterne nettverket som dataoverføringen skal sendes til (Internett, Intranett). Sammen utgjør disse tunnelene en PDP-kontekst [9].

Overføring av data:
Nå som PDP-konteksten er etablert, kan det sendes IP-trafikk gjennom denne. Før det, må UE først få en unik IP-adresse som da muliggjør datakommunikasjon. Siden disse IP-adressene er en del av PDP-konteksten, kalles de PDP-adresser. Disse adressene kan være statiske eller dynamiske. Dersom den er statisk, vil denne informasjonen være lagret hos HLR, sammen med abonnementstdata og sendes sammen med denne informasjonen. Siden det er begrenset hvor mange IPer som er tilgjengelig i IPv4 [37], vil det oftest være snakk om å få en dynamisk IP, som frigjøres etter en enhet kobler ifra nettverket. Denne IPen tildeles av en DHCP-server hos den GGSN UEen er koblet gjennom [9]. Prisen for datatrafikk i GPRS og UMTS er hos Telenor 10 kroner pr. MB [99].

Mobility Management
For å spore en UE gjennom aksessnettet, benyttes det *Mobility Management*. Den deler opp aksessnettet i flere nivåer; Innenfor en *Location Area* (lokasjonsområde) forskjellige *Routing Areas* (rutingområder). Disse består av flere forskjellige cellegrupper kalt *URA* (UTRAN Routing Area), som igjen er delt i celler [9].

Innenfor det pakkesvitsjede nettverket, finnes det en PMM-tilstand (Packet Mobile Mobility). En UE veksler mellom 3 PMM-tilstander; *Detached*, *Connected & Idle*. PMM forteller om en enhet er *Detached*, dvs. at en enhet ikke kommuniserer med nettet, mao. ukjent lokasjon i nettet. Dersom en enhet er *Connected*, vil enhetens plassering i aksessnettet følges aktivt mellom celler. Da vil SGSN vite hvilken RNC å kommunisere med. Den siste tilstanden forteller om en enhet er *Idle*. Med dette betyr det at en aktiv kobling har blitt passiv, og at enhetens lokasjons følges på rutingnivå (i forhold til sist kjent basestasjon, ikke lokasjon i selve cellen). For å vekke en passiv enhet, må det utføres *paging*.

Mens en enhet er *Connected* eller *Idle*, vil den ha mulighet for å ha aktive PDP-kontekster. En *Idle* enhet må vekkes når data skal overføres.

Sikkerhet i UMTS
Følgende informasjonen er hentet fra [11].

Ved å bygge videre på den eksisterende 2G infrastrukturen, arvet UMTS de samme sikkerhetsmekanismene som allerede var til stede. Samtidig arvet UMTS også de åpenbare sikkerhetshullene som eksisterte i GSM. UMTS innfører derfor nye sikkerhetsmekanismer for å fjerne svakhetene i 2G.

Ved standardiseringen av 3G, ble det laget en spesifikasjon for sikkerhetsprinsipper og mål i 3G. Kort forklart, kan det konkluderes med at:

- UMTS skal bygge direkte på sikkerhetsmekanismene i 2G, og det skal være fullt kompatibelt med tidligere versjoner.
- UMTS skal forbedre sikkerhetsmekanismene som finnes i 2G. UMTS skal se på og fikse klare svakheter ved 2G. Slik som toveis-autentisering av nettverk og mobil. Det skal også innføres sterk kryptering ved bruk av 128-bits nøkkel.

Den største svakheten i 2G har vært at det bare foregikk enveis-autentisering. Det betyr at teleoperatøren autentiserte den mobile enheten slik at den visste at mobilen var godkjent. Det som ikke ble gjort var autentisering fra den mobile enheten mot operatørnettet. Det vil si at den mobile enheten ikke verifiserte at det faktisk var et godkjent mobilnettverk den koblet til. Det har vært et problem i GSM-nettet at det dermed er mulig å sette opp falske basestasjoner og lure brukere.

I autentiseringsprosessen brukes det en ny krypteringsalgoritme. Metoden som ble valgt ut er Rijndael block cipher. Rijndael har blitt mer kjent som krypteringsalgoritme i AES (kap 2.7.3). Det håpes at denne krypteringsmekanismen holder ut levetiden til UMTS. Det er også innført nye algoritmer som tar seg av konfidensialitet og integritet i UMTS.

2.2.2 WLAN

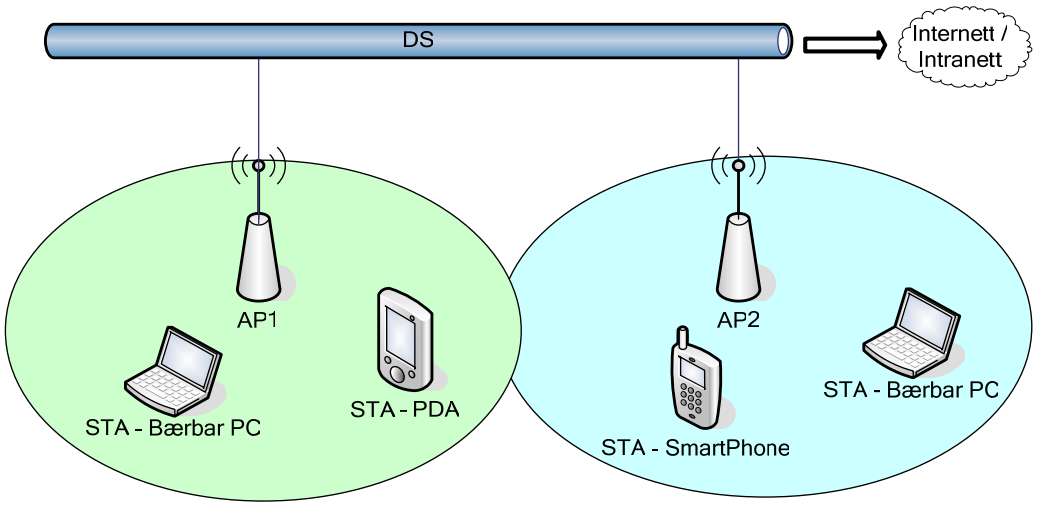
Wireless Local Area Network (også kjent som WiFi og IEEE 802.11) er en trådløs utvidelse av det kjente lokale nettverket (LAN). Den bruker radiobølger for sending av informasjon og har en rekkevidde på ca 100 meter [12]. WLAN har etter hvert blitt vanlig å finne i kontormiljø, hjemmet og på skoler. Ved å bruke WLAN spares det hundrevis av meter med kabler som ellers måtte benyttes. Til vanlig Internett-bruk, er det heller ikke noen merkbar forskjell mellom trådløst og kablet nettverk. Dette medfører muligheten for mobilitet; det er mulig å flytte seg mens man er på et nettverk uten å være avhengig av et bestemt tilkoblingspunkt. Dette gjøres ved at brukeren kobles fra et fast aksesspunkt og kobler på et nytt mens man er på det samme nettverket.

Innenfor dagens IEEE 802.11, er det definert mange forskjellige standarder. Disse standardene er definert i formatet 802.11x, der x kan være en bokstav fra a til z. Disse tar for seg forskjellige tekniske aspekter ved kommunikasjon over 802.11. De vanligste standarder i dag er: 802.11, 802.11a/h, 802.11b, 802.11g [13]

Tabell 2 - Oversikt over vanlige WLAN-teknologier i dag [13]

	802.11	**802.11 a/h**	**802.11 b**	**802.11 g**
Datarate	1-2Mbit/s	54 Mbit/s	11 Mbit/s	54 Mbit/s
Modulasjon	FHSS	OFDM	DSSS	DSSS/OFDM
Frekvensbånd	2,4 GHz	5,1 GHz	2,4 GHz	2,4 GHz

I dag er det vanlig å finne 802.11 a/h i USA og b/g i Europa på grunn av de forskjellige frekvensbåndene som blir benyttet. I Norge er det svært sjelden det finnes skoler eller andre offentlige arenaer som ikke har en form for WLAN-dekning. Den neste WLAN standarden som tilbyr hastighets- og rekkevidde- forbedring er 802.11n. Denne er i fortsatt på planleggingsstadiet, men det forventes en økning i hastiget og rekkevidde (kap 2.6).



Figur 6 - WLAN arkitektur [Egen figur]

WLAN arkitektur
Et enkelt WLAN er bygd opp av ett aksesspunkt (AP) og en eller flere stasjoner (brukere/enheter). Det området som er dekket av et aksesspunkt blir kalt for BSS (Basic Service Set). Aksesspunktet blir koblet til det videre nettverket gjennom DS (Distrubution System). Hvis flere AP er koblet sammen på samme nettverk gjennom DS, vil alle de forskjellige områdene som APene dekker bli kalt for et ESS (Extended Service Set) [14]

Aksesspunkt (AP)
Aksesspunkt har som regel en dekning på ca 100 meter der det er fri sikt. Der det er vegger/tak/gulv i veien blir dette redusert betraktelig avhengig av hindringens tykkelse og materiale. Det er likevel vanlig å si at ett AP dekker et stort hus dersom den er plassert sentralt [15]. For å skille ulike AP fra hverandre, blir det satt opp en SSID (Service Set IDentfier). SSID er navnet som kommer opp når man søker etter nettverk. Uten SSID kan man ikke få til kommunikasjon med aksesspunktet [16].

AP bestemmer hvilke enheter som får lov å koble seg til. Denne begrensningen innføres ved hjelp av eksempelvis liste over tillatte MAC-adresser, eller bruk av hemmelig nøkkel. Det er også mulig å begrense hvilke standarder får lov til å koble seg til f.eks. bare 802.11g, eller at flere kan koble seg til f. eks 802.11b og 802.11g. Ved å skjule SSID kan man også begrense tilgangen. Dette medfører at man må vite nettverksnavnet på forhånd, og skrive denne inn manuelt på klienten for å opprette kommunikasjon.

Det finnes også krypteringsmekanismer innebygd i APer slik som WEP (Wired Equivalent Provacy) og WPA (Wi-Fi Protected Access). De enkleste formene for kryptering er brutt, og dermed er WLAN som benytter disse sårbare. I nyere krypteringsmekanismer, finnes det en del sterkere kryptering som gjør det vanskelig for uvedkommende å få tilgang til det trådløse nettet [14].

Basic Service Set (BSS)
BSS er området som dekkes av et aksesspunkt, og kan sammenlignes med en celle i mobilnettet. Det er slik at i området nærmest et AP vil man oppleve de raskeste hastighetene med minst mulig feilrate [14]. Desto lenger man beveger seg fra AP, jo svakere blir signalet og dermed oppstår det oftere feil i under overføring.

Stasjon/Enhet
En stasjon er en enhet med et trådløst nettverskort med en fysisk MAC-adresse. Denne enheten får tilgang til resten av det lokale nettverket gjennom aksesspunkter. For at en trådløs enhet skal få lov til å koble seg til et AP, må begge operere med samme standard (802.11 b/g) [14]. Det er også nødvendig at det gis tilgang av AP gjennom nøkkel/MAC-autentisering. Det finnes steder der man får gratis tilgang til WLAN (hotspots), der finnes det som regel ingen begrensning på hvilke maskiner som får tilgang. I et by-miljø, er det også mulig å finne private WLAN som ikke er beskyttet. Dette er oftest pga feilkonfigurasjon av WLAN-routeren.

Enheter kan også benyttes i ad-hoc modus, dette betyr at enheter har mulighet å koble direkte til hverandre uten å benytte seg av et AP som sentral. Dette benyttes sjeldent i WLAN siden det skaper unødvendig mye trafikk og forstyrrelse i signalet, som igjen medfører svekket ytelse og tjenestekvalitet [17].

Mobilitet i WLAN
Roaming er ikke spesifisert innenfor et ESS, dvs. at det i utgangspunktet ikke er mulig å lokalisere en enhet i et WLAN. Det har i senere tid blitt utviklet teknikker for å finne posisjonen til enheter basert på målinger av signalstyrke fra flere AP.

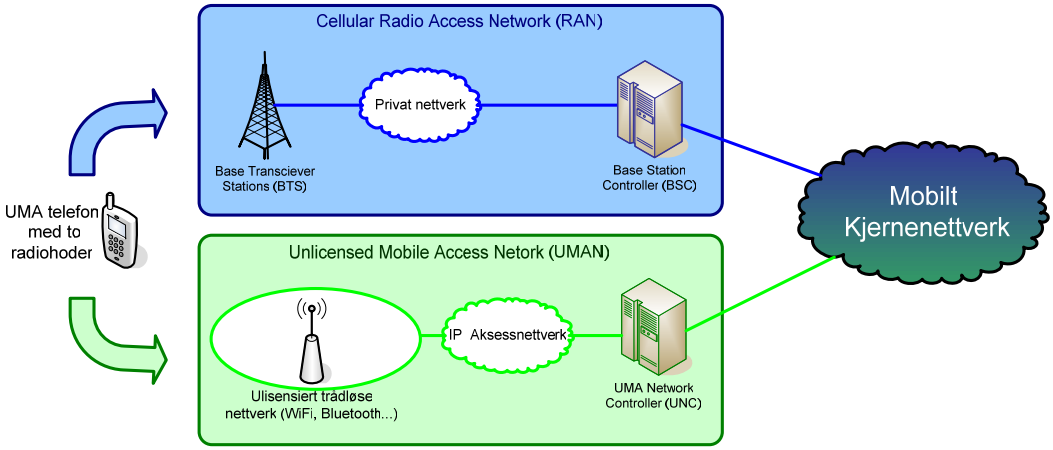
Det er tillatt med *Handoff* mellom AP, dvs. bytting av kommunikasjonspunkt med tillatt brudd i sambandet. En enhet kobler ned fra et aksesspunkt, for så å oppdage og koble opp på et nytt aksesspunkt. Det som ikke er tillatt i et WLAN er *Handover* mellom APer, dvs. bytting av kommunikasjonspunkt uten tillatt brudd i sambandet. Det er ikke mulig for en enhet å bevege seg fritt mellom APer uten å miste forbindelse siden det bare er mulig for en STA å kommunisere med én AP om gangen [13].

Sikkerhet i WLAN
Standarder opp til 802.11h tilbyr sikkerhet i form av aksessbegrensning. Dette kan gjøres ved å skjule nettverksnavnet (SSID Broadcast), eller ved å implementere nøkkelkryptering i form av WEP eller WPA. Problemet med disse sikkerhetsmekanismene er at de har blitt brutt slik at andre kan, med riktig verktøy/software, lese data som blir sendt eller koble seg til aksesspunktet og få tilgang nettverket/Internett gjennom det. Ved å innføre begrensning på MAC-adresser i tillegg, er det mulig å kun tillate gyldige enheter å koble til et aksesspunkt. Dette hindrer likevel ikke at det går an å lytte til data mellom enhet og aksesspunkt [14].

For å løse dette problemet, har IEEE utviklet 802.11e og 802.11i. 802.11e introduserer tjenestekvalitet (QoS - Quality of Service) i WLAN, mens 802.11.i implementerer WPA2. WPA2 bruker en AES krypteringsalgoritme (kap 2.7.3), noe som regnes for å være sikrere enn RC4-algoritmen som blir benyttet i WEP og WPA [14].

2.2.3 UMA

Unlicenced Mobile Access (UMA), gir tilgang til telekomtjenester som GSM, GPRS og UMTS over ulisensierte nett[3] som Bluetooth og WLAN. Operatøren kan tillate roaming for klienter med mobile terminaler med støtte for f.eks. WLAN og UMTS. En rekke spesifikasjoner er utgitt av 3GPP og støtte for dette er i ferd med å implementeres både i terminaler og hos operatører. UMA definerer dermed hvordan operatører kan gjøre WLAN-soner om til kommunikasjonspunkter for tale og data mens man er i bevegelse. Dette vil gjelde WLAN-soner som man definerer på forhånd, og gratis WLAN hotspots som man beveger seg gjennom [18].



Figur 7 - UMA arkitektur [16]

Arkitektur og teknologier

For at en mobil bruker skal kunne benytte seg av UMA må flere ting støttes i infrastrukturen. For det første må det implementeres UMA-støtte i håndsettet. Enkelte aktuelle mobiltelefonmodeller er allerede i salg (Nokia 6131/Qtek 8300). I tillegg må det være støtte for et mobilnettverk (2G/3G) og Bluetooth eller WLAN. Telefonen må ha to radiohoder for å kunne håndtere denne biten og software som snakker med radio-laget (linklaget) i telefonen.Det må også være implementert en UMA Network Controller (UNC) i kjernenettet. Denne ligger i IP-nettet og kan kontaktes i ethvert IP-nettverk. UNC kobler til IP-nettet på den ene siden og til GSM-kjernenettet på den andre. For å gjøre dette benyttes standardgrensesnitt. UNC brukes til autentisering og autorisasjon, og all trafikk rutes gjennom det eksisterende nettet. Dette vil medføre at dagens kostnadssystem vil fungere uten endringer [18].

[3]Nettverk som er åpne for alle og ikke tar betalt for tilgang og bruk. Oftest WLAN-hotspots på kafeer, bensinstasjoner, flyplasser ol.

UNC består av flere komponenter:

- En sikkerhetsgateway som terminerer IP-forbindelsen.
- En mediagateway som oversetter fra pakkesvitsjet til linjesvitsjet nettverk (GSM/UMTS).
- IP nettverkskontroller som tar seg av sikkerhet over IP-nettverket, kontroll av pakke og linjesvitsjede tjenester og signallering.

Når autentisering og autorisasjon er gjennomført og godkjent oppdateres lokasjonsinformasjonen i kjernenettet. Fremtidig trafikk, tale eller data, vil deretter bli rutet via det ulisensierte IP-nettet i stedet for det vanlige mobilnettet. Dersom brukeren flytter seg til mobilnettet flyttes brukeren og abonennentsinformasjon tilbake til dette. Eventuelle pågående samtaler eller GPRS-sesjoner blir også fortsatt i det nye nettet uten merkbar forstyrrelse. Denne handoveren er også sømløs for brukeren [18], [19].

GPRS-trafikken kjøres fra den håndholdte enheten via en sikker "GSM-tunnel", det vil si via IP-nettverket til UNC. UNC ses på som en BSC[4] fra kjernenettet. Dette medfører at det å forflytte seg fra f.eks. et WLAN hotspot på et hotell og ut i GSM (UMTS) nettet vil fungere på samme måte som det å flytte seg fra en celle til en annen i dagens GSM [18], [19].

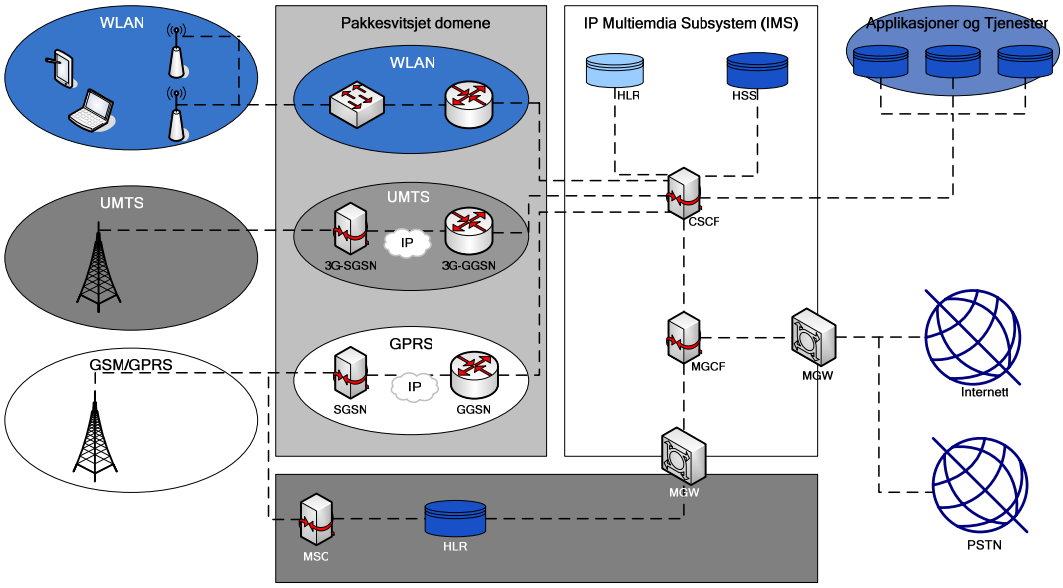
Fordeler ved bruk av UMA [20]

- Mobiltelefon blir eneste håndholdt enhet for brukeren. Samme enhet håndterer telefonsamtaler, SMS, websurfing, e-post osv.
- Utnytter bedre telefonen på lokasjoner som ikke kunne benyttes tidligere pga. kostnader eller fravær av dekning.
- Optimalisering av mobilnett (GSM,UMTS) ved å bruke nett med lavere kostnad og høyere båndbredde.
- Redusere utgifter i forhold til operasjonelle kostnader.
- Bedre dekning.
- Forbedret kvalitet på overføring av tale.
- Levere bredbåndshastigheter til håndsettene noe som utvider mulighetene for bruk og tjenester som kan leveres. Bl.a avanserte tjenester over både LAN og trådløse nettverket.

2.2.4 IMS

IP Multimedia Subsystem (IMS) er en arkitektur utviklet av 3GPP i UMTS R5-standarden [21], som muliggjør IP-basert multimedietjenester over mobilnettet [22]. IMS erstatter mye av de gamle kontrollkomponentene i UMTS-kjernenettet, og tilbyr sanntidskoblinger mellom bruk av tjenester og abonnementsinformasjon. Dette vil i realiteten bety at fakturering av kunder knyttes direkte opp mot den reelle bruken uten at operatør trenger å ha egne databaser og koblinger og rutiner som binder dette sammen [23].

[4] BSC i GSM-nettet tilsvarer RNC i UMTS-nettet (kap 2.2.1).



Figur 8 - IMS Arkitektur [24]

Tanken bak IMS er å tilby tjenester for mobilnettet som ellers bare ville vært tilgjengelig over Internett. IMS bruker SIP-protokollen for overføring av multimedia og for å sette opp sanntidskommunikasjon mellom brukere. Ved å bruke IP som underliggende kommunikasjonsprotokoll i IMS, blir håndteringen for å koble seg opp mellom 2 eller flere parter over mobilnettet eller Internett akkurat det samme [22].

I praksis betyr det at en bruker kan ha gående flere sesjoner samtidig på en mobil enhet. Det er fullt mulig å chatte (Instant Messaging - IM) med en person, samtidig som man foretar en filoverføring og har en samtale med en annen. Dette understreker at det skal tilbys alle tjenester som ellers tilbys over Internett, også fremtidige tjenester. [22],[23].

2.3 Mobilitet på nettverkslaget

2.3.1 MIP

En mobil enhet vil få forskjellig IP-adresse avhengig av hvilket nettverk den er koblet til. Dette er et resultat av begrensninger i dagens IPv4-protokoll. Det finnes ikke nok adresser til alle enheter som er koblet til Internet på de ulike nettverkene rundt i verden. Mobile IP (MIP) er en IETF-standard[5] som brukes for å tilltate mobile brukere som befinner seg innenfor et nettverk å flytte seg til et annet nettverk uten å miste sin IP-adresse [25]. I et IP-nettverk har man vanligvis en hjemmeadresse (home address). Det vil si at man har en IP-adresse som knytter deg til et kontaktpunkt f.eks. et aksesspunkt i WLAN. Denne adressen byttes eller oppdateres hver gang man kobler til et nytt punkt. For at det skal være mulig å kommunisere mellom to maskiner på Internett, er man avhengig at adressene til disse forblir statiske under kommunikasjonen. Dette er altså ikke tilfelle når en enhet endrer sin tilkoblingspunkt til Internett (typisk bytting mellom to forskjellige WLAN). Så fort en av adressene i kommunikasjonen endres vil

[5] Internet Engineering Task Force

forbindelsen termineres og man får en ny IP-adresse ved det nye kontaktpunktet. Deretter vil forbindelsen bli satt opp på nytt. Forenklet kan det sies at MIP brukes for å gjemme alle adresseforandringer og dermed gi transparent mobilitet.

MIP Arkitektur
MIP kontrollerer tunnelering, videresending, routing og mapping av pakker og de forskjellige nettverkene. MIP fungerer ved hjelp av en node i hjemmenettverket kalt Home Agent (HA). HA funger som et mellomledd mellom den mobile enheten og resten av Internett. Når en bruker vil koble opp til Internett med MIP, må den gå gjennom HA. Det tildeles en offentlig IP-adresse for brukeren, dvs. all trafikk til og fra brukeren blir tilegnet en offentlig adresse som er synlig på Internett. Når abonnenten er koblet til et annet nettverk enn hjemmenettverket vil alle pakker til abbonenten automatisk gå til HA.

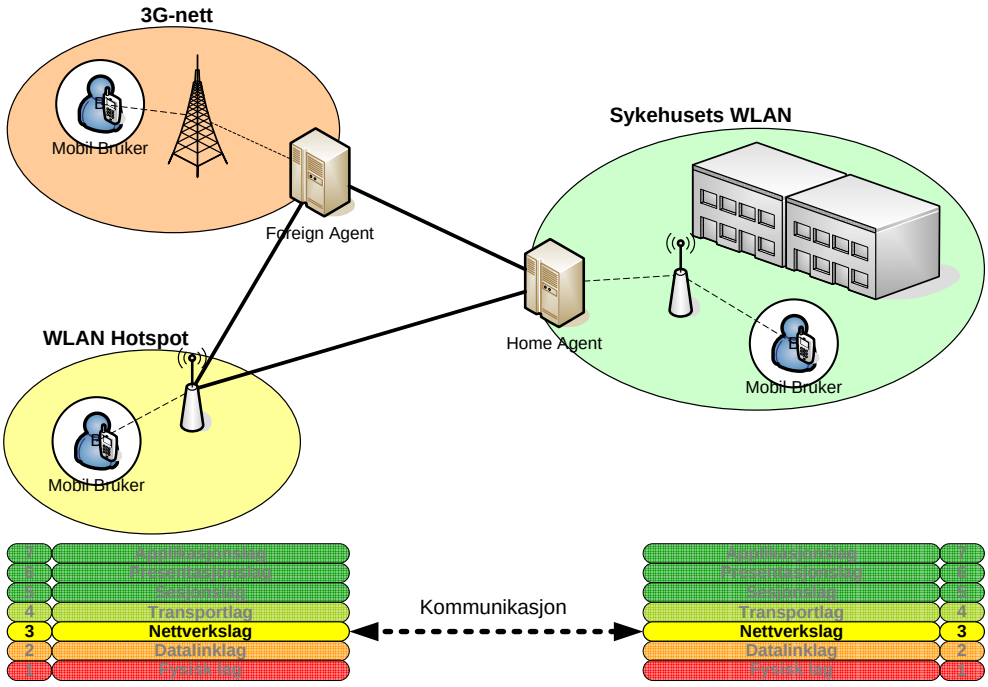
Når enheten beveger seg over i et fremmed nettverk, vil den automatisk få tildelt en IP-adresse. Denne adressen er ikke det samme som den som er tildelt via HA. Når denne nye adressen mottas av enheten, sender den melding til HA og oppdaterer med den nye adressen. Deretter vil all trafikk til og fra enheten sendes til den opprinnelige offentlige IP-adressen. På denne måten, vil enheten i teorien aldri miste sin opprinnelige IP-adresse.

HA vil deretter sørge for å rute pakkene videre ut i nettet til abonnentens nåværende IP-adresse; en Care-of-Address (CoA). Abonnenten oppdaterer hele tiden til HA når man bytter nettverk. To forskjellige metoder kan brukes for å håndtere abbonenter i ”fremmede” nettverk. Man kan implementere en Foreign Agent (FA). En FA er en spesiell ruter som tar vare på og tildeler CoA og sender videre trafikken med HA som kilde til destinasjonen. MIP kan også fungere uten en FA, dette skjer via en sammenstillet CoA. Dette gjøres ved at det implementeres et virtuelt nettverksinterface i terminalen og at man via disse blir tildelt en CoA. På denne måten implementerer man MIP uten å være avhengig av en FA [26], [27].

Micromobility
Micromobility er en samlebetegnelse på teknologier basert på Mobile IP, blant disse Cellular IP [28], for å håndtere lokal mobilitet uten å kontakte et større nett med MIP. Cellular IP erstatter, som Mobile IP, IP som nettverksprotokoll, men uten å endre på IPs pakkeformat og videresendingsmekanismer. Det eneste den erstatter er IPs rutingmekansimer i nettverket. Cellular IP bruker rutingmekanismene også som lokasjonsregister for hostene ute i nettet.

MIP-løsninger
Mobile IP er utgangspunktet for Birdsteps forskning på området som har resultert i produkter som tilbyr sømløs roaming uavhengig av underliggende transmisjonteknologier [29]. Det er også foreslått teknologier som bruker Mobile IP for mobilitetsstyring og en applikasjonslagsprotokoll for sesjonskontroll [27] eller andre hybridimplementeringer der man velger hva slags mobilitetsteknikk som skal brukes utfra en policy-tabell, der man definerer hva slags kommunikasjoner som skal bruke MIP og hva som skal bruke en protokoll på et høyere lag [30].

MIP tilbyr en generisk løsning for alle IP-baserte tilkoblinger i forhold til langtidssesjoner som FTP og HTTP. MIP får derimot problemer så fort det blir snakk om sanntidsapplikasjoner som streaming av audio og video. Timing er ikke tatt høyde for, og kan ikke bli tatt høyde for, i MIP fordi det opererer på nettverkslaget. Den store ulempen med dette vil være det at man ikke kan påvirke nettverksspesifikasjonene som for eksempel båndbredde. Et eksempel på dette vil være om man har en sesjon bestående av audio og video og kommer inn i ett nett med mye mindre båndbredde, så mye mindre at man ikke kan støtte video-overføring lenger. Man har da ingen mekanismer for å påvirke dette slik at man slår over på kun audio. I Figur 9 ser vi hvordan den grunnleggende arkitekturen til Mobile IP er bygget opp og tilhørende plass i OSI-modellen. I figuren er det illustrert en arkitektur med FA.



Figur 9 - Grunnleggende arkitektur for Mobil IP [Egen figur]

MIP har også tre andre problemer som må løses eller i det minste tas hensyn til [30]. For det første et problem kalt triangulær ruting. Triangulær ruting betyr at pakkene fra og til en mobil node går to forskjellige veier for å nå den korresponderende noden. Dette kan løses ved route optimization [31] som går ut på å sende oppdateringer til hostene om motpartens posisjon slik at motparten kan sende direkte, men dette har også sine ulemper [30]. Et annet problem er rett og slett adressemangel i dagens Internett med IPv4. Dagens Internett er allerede på bristepunktet når det kommer til tilgjengelige IP-adresser, dersom Mobile IP skal implementeres på stor skala, kan det bli problemer med tilgjengelig adresser (kap 2.5.1). Det tredje problemet er også knyttet til selve rutingen. Pakkene til den mobile noden tunnelleres ved hjelp av ”IP i IP” [32] fra Home Agent i hjemmenettverket. Dette fører til mye ekstra overhead og dermed mer trafikk og forsinkelse i nettverket. Performance for MIP avhenger primært av avstandene mellom den mobile hosten, hosten i andre enden av kommunikasjonen og hjemmenettverket til den mobile hosten [28]. Undersøkelser har vist at ved innføringen av mobile IP i et campus økte forsinkelsen i nettverket med 45% [33].

2.4 Mobilitet på applikasjonslaget

2.4.1 SIP

Session Initiation Protocol (SIP) ble definert av IETF i samarbeid med 3GPP og formalisert som RFC 3261[6]. SIP er en request-response protokoll på applikasjonslaget som initierer, endrer og terminerer en interaktiv brukersesjon som kan inneholde multimedieelementer som video, tale og lynmeldinger [34]. Dette betyr at SIP kan brukes uavhengig av underliggende nettverk så lenge det støtter IP. SIP kan dermed gi propreitære løsninger på applikasjonslaget som kan brukes i f.eks. Voice over IP—applikasjoner (VoiP) [35] eller som basis for et pasientmonitoreringssystem i en helsekontekst [36].

IMS er en standardisert arkitektur for mobiloperatører for å gi mobile multimedialøsninger. Målet til IMS er å kunne tilby funksjonaliteten og protokollene som eksisterer i dagens og fremtidens Internett over 3G-nettet. Dette skal gjøres på applikasjonslaget ved hjelp av SIP-protokollen. Eksempler på funksjonalitet kan være aksessuavhengighet, nettverksarkitekturer og mobilitet. IMS vil fungere uavhengig av hvilken pakkesvitsjet nettverksteknologi som benyttes. Tjenestene som tilbys er stort sett alle IP-baserte tjenster i dagens Internett blant annet VoIP, Push to talk over Cellular (POC), multiplayer gaming og lynmeldinger.

SIP-meldinger
SIP opererer utfra en sentral server som tolker meldinger fra sine abonnenter. SIP håndterer mobilitet utfra to meldinger: INIVTE og REGISTER.
REGISTER gir en registrering hos en SIP lokasjonsserver og INVITE initierer en SIP-sesjon med en annen host i samme eller fremmed nettverk. Disse to meldingene utgjør alt av registrering og initiering av sesjoner i et SIP-basert sytem.

Eksempel på REGISTER-melding i SIP

REGISTER sip:company.com SIP/2.0
To: sip:user@company.com
From: sip:user@company.com
Contact: sip:user@host.company.com
Call-ID: k345asrl3fdbv@10.0.0.1
CSeq: 1 REGISTER
Via: SIP/2.0/UDP 135.180.130.133
Contact: <sip:user@example.com?Route=%3Csip:sip.example.com%3E>

En SIP-adresse består av brukernavn og domene. Domenet peker til hjemmedomenet for et SIP-abonnement. I meldingen over er brukerens adresse *user@company.com* der user er brukernavnet og company.com er domenet der brukeren hører hjemme. I dette hjemmedomenet ligger også registreringsserveren som tar imot REGISTER-meldingen og mapper brukernavnet opp mot den gjeldende IP-adressen i sin lokasjonsdatabase.

[6] "SIP: Session Initiation Protocol": http://rfc.net/rfc3261.html

Dette medfører at så fort en bruker vet SIP-adressen til en annen bruker vil den kunne kontakte den på tross av at brukerens IP-adresse endres som følge av at den andre flytter seg til nye domener.

Eksempel på en INVITE-melding i SIP

INVITE sip:user@company.com SIP/2.0
To: sip:j_user@company.com
From: sip:caller@university.edu
Call-ID: 0ha0isndaksdj@10.1.1.1
CSeq: 8 INVITE
Via: SIP/2.0/UDP 135.180.130.133

INVITE meldingen vil typisk bestå av to deler. En header som inneholder SIP-felter og en body som gjerne inneholder en Session Description Protocol (SDP).

SDP er en protokoll som beskriver mediasesjoner og brukes kun for dette, den tar ikke hensyn til ulike transportprotokoller. SDP gir muligheter for å spesifisere applikasjoner, kodeker og mottakeradresser. Dette er forslag til parametre som motparten mottar og dersom motparten støtter disse kopieres disse og sender SDP-bodyen tilbake med sin OK-melding. Dette betyr at initiering av sesjoner tar minimum tre SIP-meldinger. Dersom den andre brukeren ikke støtter de foreslåtte parameterne endres disse i svarmeldingen.

For å sikre full mobilitet sendes en REGISTER (re-REGISTER) som oppdaterer en brukers adresse dersom denne byttes. Ved hjelp av dette og sending av en ny INVITE (re-INVITE) sikres full mobilitet og opprettholdt av gjeldene sesjoner også dersom en av partene bytter IP-adresse.

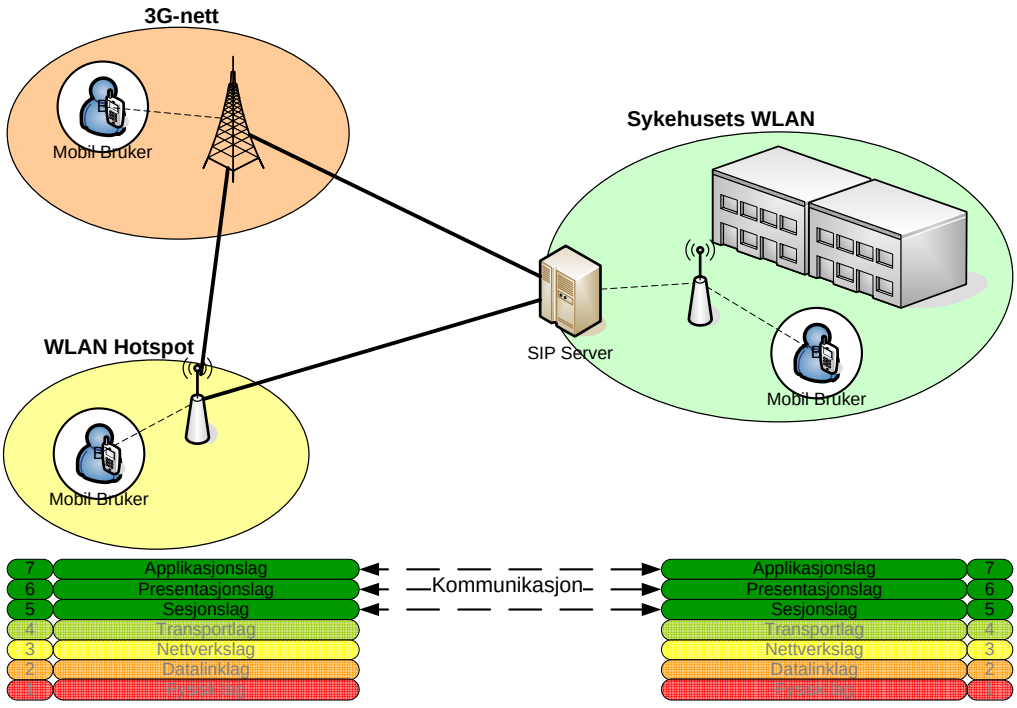
Utvalgte SIP komponenter

SIP User Agent
Hver bruker har en User Agent på sin enhet. En User Agent er en klientapplikasjon som kan kjøres på PC, PDA, mobiltelefon eller andre personlige enheter. I utgangspunktet vil dette være alle enheter med mulighet for å ta imot og sende IP-trafikk. På denne enheten må det også være en eller annen form for funksjonalitet som støtter SIP. Det vil si at det må implementeres støtte for SIP-stacken i enheten og denne må kunne aksesseres via programmet som kjører SIP-klienten. Et eksempel på dette er JRE 180. En SIP API for J2ME (kap 2.8.2) som muliggjør kjøring av SIP-applikasjoner på enheter med begrenset minne som f.eks mobiltelefoner.

SIP Proxy Server
En SIP-Proxy Server er ansvarlig for å videresende SIP-meldinger til neste SIP Proxy eller eventuelt til abonnenten. Den bruker Via-feltene i SIP-headeren for å påføre sin egen adresse slik at meldingene hele tiden kan rutes tilbake samme veien som de blir sendt. Når svarmeldingen går tilbake fjernes Via-feltene fra headeren. I tillegg til videresending kan det utføres autorisasjon, autentisering, aksesskontroll og sikkerhet.

SIP lokasjonsserver
En lokasjonsserver kan ligge på samme enhet som proxy-serveren og er i sin aller enkleste form en database som mapper fra brukernavn (SIP-adresser) til IP-adresser. Utvidet funksjonalitet utover dette kan være lagring av brukerpreferanser ol.

SIP Gateway
En SIP gateway gir en gateway mellom det pakkesvitsjede SIP-domenet og utenforliggende linjesvitsjede nett. Gatewayen vil da typisk oversette mellom SIP-signalleringen og VoIP-signalering på telefonnettet.



Figur 10 - Grunnleggende arkitektur for SIP [Egen figur]

Generelt sett har applikasjonslagprotokoller, som SIP, problemer i forhold til handoff-delay, signalering, overhead og transparency [37]. I forhold til innføring av et ”all IP”-nett vil det bli behov for en protokollstack og endringer av infrastrukturen i nettverkene. Dette er løst på forhånd med SIP og andre applikasjonslagprotokoller. Et annet argument for SIP er at ideelt bør applikasjonene som kjører være med å påvirke eller til og med kontrollere mobiliteten. I applikasjoner som streamer audio og video vil dette være veldig viktig da båndbredden på det aktuelle nettet i prinsippet bestemmer alt av parametre som er viktig for den aktuelle mediastreamen. Applikasjonene kan dermed endre sin oppførsel og gi den beste ytelsen til sluttbruker. I tillegg til dette er det på kort sikt gunstig å bruke applikasjonslagsprotokoller der det lettere kan implementeres likt på forskjellige komponenter fordi det kjører uavhengig av underliggende systemer.

Som nevnt tidligere har SIP to meldinger for håndtering av mobilitet; INVITE og REGISTER. Re-INVITE gjøres for å viderføre en eksisterende SIP-sesjon til en ny IP-adresse. Som i tidligere nevnt eksempel kan dette gjøre at man kan forandre parametre for SIP-sesjonen eksempelvis kutte video fra en audio/video-sesjon dersom det nye

nettets båndbredde ikke er tilstrekkelig. Dette fører også med seg et stort problem for SIP-protokollen. I noen situasjonener kan det hende det nye nettet ikke har kapasitet for audio/video samtidig. Dette gjør at man får en kort periode hvor det sendes feil type data til den nye destinasjonen. Det vil si den korte perioden det tar fra re-REGISTER meldingen kommer frem og trafikken rutes til den nye IP-adressen til re-INVITE meldingen kommer frem til SIP-serveren. I denne perioden kan, dersom den nye kapasitetet er tilstrekkelig lav i forhold til den forrige, tilstrekkelig store mediastrømmer floode nettverket og i ytterste konsekvens hindre signaleringsmeldinger å nå frem til SIP-serveren [37]. I SIP slipper man også problemet med triangulær ruting bortsett fra i initieringsfasen av en sesjon. Det betyr at når sesjonen er satt enten via INVITE eller re-INVITE vil all kommunikasjon foregå P-2-P. I figur 3 ser vi den grunnleggende SIP-arkitekturen og tilhørende plass i OSI-modellen.

2.5 Tilgjengelighet - Adressering i Mobilnett

2.5.1 Adressering i Internett: IP

IP (Internett Protocol) er kommunikasjonsprotokoll på nettverkslaget laget for overføring av data. IP-pakkene blir sendt over linklaget, og IP tilbyr universell adressering som gjelder uavhengig av linklaget. Det muliggjør kommunikasjon mellom enheter som ellers ikke støtter felles kommunikasjonsmekanismer [38].

Tidlig på 1980-tallet ble IPv4 innført som en protokoll på ARPANET – det tidligere testnettet som etter hvert utviklet seg til det vi kjenner som Internett i dag. IPv4-adresser er 32-bits med formatet xxx.xxx.xxx.xxx, der xxx går fra 0-255. Dette gir totalt 4.3×10^{9} (4.3 milliard adresser). Ca 19 millioner av disse adressene er reservert til private og offentlige enheter som står i kjernen i Internett [39].

IP-adresser ble tildelt etter geografisk lokasjon i forhold til tilknytningspunkter på Internett stamnettet. Etter hvert som Internett har spredd seg verden rundt og har blitt like naturlig for private kunder som for store bedrifter, har det vist seg at det ikke er nok adresser til alle. I forbindelse med dette har utviklingen av nye IP-versjonen, IPv6, vært underveis siden 1994. IPv6 løser flere av svakhetene som finnes i IPv4 ved å forenkle håndtering av pakker og også tilby innebygd kryptering av pakker samt kvalitetstjenester. Det er også utvidet til en adresselengde på 128-bit, noe som utgjør 3.4×10^{38} adresser [40]. Det vil nå tildeles geografisk ”smarte” adresser som gjør at man til enhver tid vil få en IP som er riktig plassert i hierarkiet. Dette fører til logisk kommunikasjon innenfor kjernenettet, som igjen fører til kraftig effektivisering av hastighet og responstid på tvers av Internett.

Det er to hovedmetoder for å løse adressemangelen i dagens IPv4-protokoll; dynamisk adressetildeling [41] og NAT (Network Address Translation) [42]. Dynamiske adresser er oftest tildelt av Internett leverandører og operatører til kundene sine. En leverandør leier et visst antall adresser, som regel færre enn det faktisk er kunder, og deler de som er tilgjengelig til de brukerne som til enhver tid er aktive på Internett. Når kundene logger av, frigjøres adressen slik at den kan tildeles neste klient som kobler på. På denne måten oppnår man effektiv gjenbruk av adresser. Etter hvert som ADSL erstatter

oppringt Internett (modem over telefonlinje), blir det vanligere for kunder å være tilkoblet oftere og lengre om gangen. Dette bidrar til enda høyere forbruk av adresser. Det som er viktig å notere seg er at IP-adressene som tildeles klientene er 100 % unike, slik at all kommunikasjon over Internett går direkte til enheten.

Den andre hovedmetoden er NAT. Dette betyr at en eller flere enheter kobler seg opp mot Internett gjennom en annen. Ofte brukes det dynamisk IP-adressering bak denne maskinen igjen. Dette er noe som ofte blir brukt hos privatkunder for å få flere maskiner på nett. Fordelen med denne løsningen er at maskinene som står bak en aktiv NAT ikke bruker offentlige unike adresser, de ”arver” adressen på maskinen/routeren som NATer. På denne måten foregår det et gjenbruk av adresser som ikke påvirker den totale tilgjengeligheten av IP-adresser. Ulempen med dette igjen er at maskinene som står bak, ikke er synlig utenfra. Det vil si at det ikke er mulig å få kontakt med maskinene direkte fra Internett. Applikasjoner som bruker peer-to-peer kommunikasjon blir ubrukelige fordi maskinene som ikke initierer kommunikasjon ikke kan kontaktes utenfra uten at applikasjonen selv håndterer den interne routingen. Det finnes løsninger der man kan videresende informasjon som kommer inn på spesifikke porter til maskiner bak NAT, men dette må sette opp manuelt, og gjelder kun for den ene maskinen og den gitte portrekken.

2.5.2 Adressering i mobilnettet: APN

Kommunikasjon i det pakkesvitsjede mobilnettet foregår med IP i bunn, i likhet med resten av Internett. Dette er en naturlig utvidelse av mobilnettet for å sikre kompatibilitet med det resterende eksterne nettverket. Over de siste årene har bruken av datatjenester for mobile enheter eksplodert. I IP-protokollen er det ikke nok adresser tilgjengelig til å dele ut én IP per person uten at det går tomt. Ved å tilby muligheten for alle mobiltelefoner i verden å koble seg til Internett, er det ikke mulig å støtte alle disse enhetene [43].

I UMTS nettet, er det GGSN som deler ut IPer videre til mobilenheter. Prinsippene for IP gjelder også videre i mobilnettet, dermed brukes de samme teknikkene for å dele Internett til mange enheter uten å faktisk bruke like mange unike adresser (NAT) [44].

Når en mobil enhet skal koble til Internett, kobler den til et APN (Access Point Name) [43] hos sin mobiloperatør. Dette punktet er et tilkoblingspunkt som finnes på GGSN som da tildeler enheten sin egen interne IP ”bak” den unike adressen til GGSN. Mobiloperatøren som eier sitt eget nett, eier også sine egne APN. Det er vanlig å ha flere APN som da brukes til diverse tjenester i nettverket, f.eks. WAP og MMS eller offentlige tjenester. Bak én IP-adresse kan det altså finnes flere tusen mobiltelefoner som laster ned bildefiler eller surfer på Internett. Dersom det skal kjøres en peer-to-peer applikasjon mellom en mobil enhet til en klient/server på det eksterne internett, vil det oppstå komplikasjoner med å sende data til den mobile enheten [44].

2.6 Fremtidige Teknologier

2.6.1 Fremtidig utvikling av UMTS

Dagens UMTS R3 standard, er en relativ enkel forbedring av det eksisterende 2G-mobilnettet. Det eneste som må bygges ut er UTRAN aksessnettet. Med noen få oppgraderinger på eksisterende infrastruktur i kjernenettet, har UMTS samme funksjonalitet, med høyere hastigheter enn GSM/GPRS. Dette er bare første skritt i UMTS utviklingen.

UMTS R4 introduserer en enhet kalt CS-MGW (Circuit Switched-Media Gateway) i det linjesvitsjet kjernenettet [21]. Denne enheten tar over overføringen av data over det linjesvitsjede nettverket, mens kontrollsignalene går gjennom de eksisterende MSC/GMSC. I tillegg til dette, legges det til ekstra funksjonalitet i form av tjenester. HSDPA (High-Speed Downlink Packet Access) integreres [45]. Dette begynner å bli oppgradert tidligst i løpet av 2006.

UMTS R5 introduserer IMS og HSS (Home Subscriber Service) i kontrollplanet for kjernenettverket [21]. Dette er en ganske revolusjonerende oppgradering, som krever at det investeres i betraktelige oppgraderinger i kjernenettverket. IMS tilbyr støtte for tale over det linje- og pakkesvitsjede kjernenettverket (VoIP og vanlig tale), samtidig som det tilbys nye IP-baserte datatjenester (kap 2.2.4). HSS er en tjeneste som benyttes til å håndtere alle samtalene og kostnadene i sanntid og binder dette opp mot abonnementsinformasjon. Dette forenkler arbeidet for teleoperatørene betraktelig [24]. Slik det er nå, må operatørene overvåke og håndtere dette selv, med HSS kan de bare hente ut informasjonen direkte. For å håndtere at enheter får fritt tilgang til tjenester på Internett, er det tenkt å innføre bruk av IPv6 for å løse problemet med adressemangel. HSUPA (High-Speed Uplink Packet Access) integreres [45] og R5 er tenkt på banen ca 2008-2010.

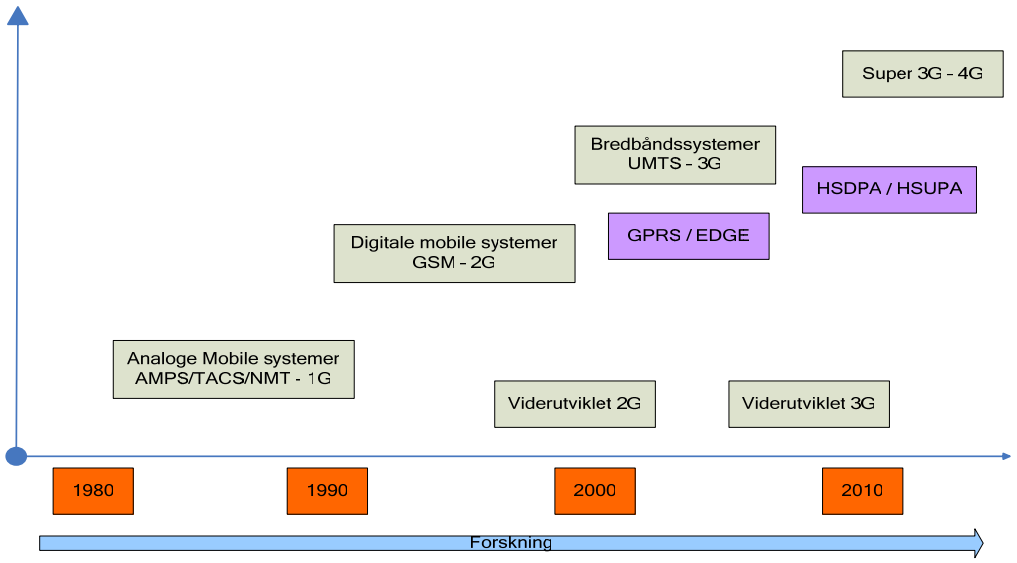
UMTS R6 er ikke ferdig standardisert pr dags dato. Det er tenkt at R6 skal tilby roaming over flere kommunikasjonskanaler i tillegg til mobilnettet. Det er da snakk om å bruke det eksisterende kjernenettverket og implementere felles håndtering for tale- og datakommunikasjon over WLAN knyttet opp mot operatør. R6 kommer til å være et stort steg på vei til neste generasjons mobilnettverk; 4G (kap 2.6.3). Det er tenkt implementert 2010-2012

2.6.2 Fremtidig utvikling av WLAN

Av kommende WLAN-standarder, er det 802.11n som er det neste steget i fart og rekkevidde. Det er forventet hastigheter 10 ganger raskere enn 802.11a/g samt økt rekkevidde. For å fremskynde utviklingen av 802.11n, ble EWC (Enhanced Wireless Consortium) etablert med 27 ledende aktører innenfor Wi-Fi. De 3 partene som har stått for forskjellige forslag for en kommende n-standard har gått sammen og lagt frem et felles forslag for EWC. I mars 2006, ble første utkast av 802.11n sendt ut til medlemmer av EWC for tilbakemeldinger og forbedringer. Det er foreløpig ikke satt inn noen forventet dato for videre implementering [14].

2.6.3 Overgangen til et ”All-IP” nett

Dagens 2. generasjons mobilnett (2G) er på vei til å bli utnyttet til det maksimale. Det er stadig nye teknologier som blir lagt til det for å forbedre ytelsen på de aldrende 2G nettverkene. Implementering av 3G er på fremmarsj i det fleste land i Europa. Denne implementeringen puster nytt liv i gammel i gammel teknologi ved å utnytte den samme infrastrukturen som 2G med en utvidelse av aksessnettet. Denne utvidelsen tilbyr betydelig høyere hastigheter over det pakkesvitsjede nettverket som igjen gir grunnlag for mye høyere hastighet for dataoverføring. Det er en stadig økende fokusering på høyere hastigheter og økte muligheter for brukere til å benytte seg av IP-basert kommunikasjon. Dette fører til at det etter hvert vil bli mulig å implementere trafikken som vanligvis går over det linjesvitsjede nettverket over på det pakkesvitsjede, og dermed få tale over IP (VoIP).



Figur 11 - Utviklingen av mobilnettet[46]

For at tale skal gå over IP uten at brukeren med mobiltelefonen skal trenge å ta hensyn til hvilket nettverk han befinner seg i, må det tilrettelegges for ”universell” sømløs roaming. Dette er målet for 4. generasjons mobilnettverk (4G); all kommunikasjon over IP med sømløs roaming mellom flere trådløse nettverkstyper. Implementering av et slikt system vil gi brukeren muligheten til å hele tiden være på det mest hensiktsmessige alternativet for kommunikasjon mens man flytter seg. Oppgraderingen fra 2G til 3G innebærer at samme infrastruktur benyttes mens aksessnettet må oppgraderes. 4G skal kunne fungere uavhengig av aksessteknologi; løsningen skal sitte bak de forskjellige infrastrukturene og håndtere kommunikasjonsflyten over IP.

I dagens mobilnettverk har brukeren 2 valg; å fortsette å bruke 2G eller gå over til 3G. Det er lite forskjell på maksimalhastigheten som er tilgjengelig over 2.75G (EDGE: maks 236.8 - 473.6 kbit/s) [46] og 3G (UMTS: maks 384 - 1920 kbit/s) [47]. Dette er

hastigheter som endrer seg etter hvert som implementering av 3G fortsetter. HSDPA (3.5G) introduserer en ny W-CDMA kanal som er dedikert nedlasting, og forventet å begynne å bli implementert rundt desember 2006 [49]. Dette blir videre utviklet med HSUPA (3.75G), som vil tilby muligheten til å laste opp med hastigheter opp mot 5 Mbit/s. Det siktes å introdusere dette i 2007/2008 [50]. Å begynne å snakke om å overføre all tale over IP krever at det er støtte for å sende ut like mye data som man mottar. I dagens 3G-infrastruktur er det 4 pakker dedikert til nedlasting pr. 1 pakke dedikert til opplasting. Ved å innføre en større maksimalhastighet på opplasting, vil den totale båndbredden være betraktelig høyere og dermed støtte flere samtaler om gangen. Det åpner også opp muligheten for å bruke mobile enheter sammen med mer linjekrevende applikasjoner.

2.6.4 Behovet for et ”All-IP” nett

Mobiltelefonen er i ferd med å erstatte fasttelefon. I dag er mobiltelefonen en daglig del av livet for over 1 milliard mennesker verden over. Når kostnadene med å benytte mobiltelefon synker vil bruken i hjem og på kontorer øke. I Europa har ¼ av brukerne flyttet noe av sin fasttelefonbruk over på mobiltelefonen og 6 % av brukerne planlegger å fjerne sin fastlinje i fremtiden [51]. Brukere peker spesielt på to ting som må være like bra som faste linjer for mobile nettverk: Kvalitet og pris. Hvis disse to tingene er i orden vil nesten halvparten av mobilabonnentene benytte seg av mobiltelefonen som sin hoved-kommunikasjonskanal.

Bredbånd har opplevd en enorm vekst og det er anslått at det skal fortsette å vokse med 28 % hvert år. [52]. Behovet for trådløs aksess i hjemmet og på kontoret blir drevet frem av den økte bruken av smartphones, PDAer og laptoper. Dette blir i første rekke tilbudt vha. WLAN. WLAN-markedet drives av hjemmemarkedet og opplevde en vekst på 9,2 % mellom andre og tredje kvartal i 2004 [53]. Framveksten av gratis hotspots og åpne nett på bensinstasjoner, kafeer, hoteller, flyplasser gir mobiloperatører en unik mulighet for å tilby tjenester som krever høye hastigheter til den mobile bruker.

Allerede i dag snakkes det om neste generasjons mobilsystemer; 4G. Dagens 3G-teknologi har ikke hatt den innslagskraften som mange spådde på forhånd og killer-applikasjonen har enda ikke blitt levert til 3G-systemet. Med killer-applikasjon menes en tjeneste som brukes i hverdagen til de fleste mennesker slik SMS gjorde for GSM og tale i sin tid for første generasjons mobilsystemer. Videotelefoni skulle være det som gjorde 3G til allemannseie, men dette har ikke slått til som forventet.

En markedsundersøkelse gjort av Infonetics Research i 2005 viser at stadig flere ringer over WLAN. Tall viser at bedrifter er på vei til å legge om til IP-telefoni og 52 % av markedet var i Asia/Stillehavområdet, mens Nord-Amerika står for 25 % og Europa, Midtøsten og Afrika for 21 %. Med 802.11e på vei, som tilbyr tjenestekvalitet i WLAN, vil det også bidra til vekst innenfor bruken av IP-telefoni [54].

Dagens mobilsystemer har hele tiden utviklet seg side om side med datautviklingen. Dataverdenen har fremmet sin pakkesvitsjede teknologi, mens mobilsystemene har benyttet seg av linjesvitsjet teknologi fra televerdenen. Med overgangen til digitale nett

for teleselskapene og Internetts posisjon i dagens samfunn ser det ut til at den videre utviklingen vil dreie i retning av et ”All-IP”-nett.

2.7 Sikkerhetsmekanismer

2.7.1 Introduksjon til datasikkerhet

De tre grunnleggende aspektene i datasikkerhet er tilgjengelighet, integritet og konfidensialitet. Tolkningene av disse tre tingene avhenger av miljøet de befinner seg i og avhenger av personlige behov, skikker og lovverk i den gjeldende organisasjonen. Avsnittene frem til kap 2.7.3 er basert på informasjon fra [55] [56] [57].

Konfidensialitet
Konfidensialitet er skjuling av informasjon eller ressurser. Behovet kommer fra sensitive felter som myndigheter eller industri behøver å skjerme for innsyn. Et eksempel kan være militær informasjon som ofte behandles på en need-to-know basis. Et annet eksempel kan være personlig informasjon fra forskjellige institusjoner eller hvor mye penger en gitt person har på sin sparekonto.

Den vanligste metoden for å gi konfidensialitet er aksesskontroll. Aksesskontroll kan gis ved hjelp av kryptografi. En kryptografisk nøkkel kontrollerer tilgang til data, men blir med dette en ny ting å beskytte mot innsyn. Konfidensialitet omfatter også beskyttelsen av at data finnes. Det kan for eksempel være mer interessant å vite at en idrettsutøver har brukt doping enn å vite nøyaktig hva slags type doping denne utøveren har benyttet seg av. Aksesskontrollmekanismer kan derfor skjule at data eksisterer dersom disse dataene avslører informasjon som burde vært beskyttet. For eksempel kan det være ønskelig begrense tilgang til innholdet i en database, men mange ganger kan det være enda viktigere å begrense tilgangen til selve databasen og dens struktur.

Alle mekanismer som gir konfidensialitet, for eksempel et kryptografisk system med nøkkeldistribuering, er avhengig av en eller annen form for *trust*. Trust vil si at man tar for gitt at deler av systemet i seg selv er sikkert og dermed fungerer som en ”black box”[7] for utenforstående. I en applikasjon som benytter seg av kryptografiske nøkler som genereres og distribueres av systemet tas det for gitt at disse nøklene er riktig generert og ikke distribueres til andre mottakere som dermed skaffer seg fullt innsyn i en gitt prosess.

Integritet
Integritet benyttes for å garantere at dataene eller ressursene kommer fra der de sier de kommer fra. I de fleste tilfeller dreier dette seg om å detektere feilaktige eller uautoriserte endringer av data. Dette inkluderer dataintegritet og opprinnelsesintegritet. Dataintegritet dreier seg om å garantere at selve informasjonsinnholdet, f.eks. meldingsinnhold, ikke er endret. Opprinnelsesintegritet, eller autentisering, skal garantere at kilden til dataene er den som den gir seg ut for å være. Et eksempel på dette

[7] Betegnelse på det å skjule logikk og dermed kun ta hensyn til hva som går inn og hva som kommer ut av operasjonen.

kan være en nyhetssak i en avis som baserer seg på anonyme kommentarer fra en stortingsrepresentant. Informasjonen kan være trykket nøyaktig som denne personen har fortalt, altså ikke endret, men det kan vise seg at denne personen slett ikke er stortingsrepresentant. Det er derfor viktig at man, for å gi fullstendig integritet, både sjekker at selve innholdet stemmer og at innholdet kommer fra rett person. Integritet fungerer på mange måter annerledes enn konfidensialitet. Konfidensialitet er enten brutt eller ikke og må behandles deretter. I motsetning til konfidensialitet er integritet bl.a. avhengig av beskyttelsesnivået hos opphavet og hele transportveien for dataene kan spille inn. Det er to typer integritetsmekanismer; forhindringsmekanismer og deteksjonsmekanismer.

Forhindringsmekanismer
Forhindringsmekanismer bevarer integritet ved å fysisk hindre og blokkere uautoriserte forsøk på å endre selve dataene eller uautoriserte måter å endre dataene på. Forskjellen på disse er at den første blokkerer forsøk på å endre data som en bruker ikke har tilgang til å endre. Den andre hindrer en bruker å endre dataene på en måte som ikke er tillatt. Et eksempel kan være en bankansatt som har mulighet til å endre gitte detaljer i en kundeinformasjon som for eksempel navn og adresse, men ikke endre saldo på denne personens konto. Disse mekanismene i form av autentisering og aksesskontroll vil typisk stoppe tilgang fra utsiden.

Deteksjonsmekanismer
Deteksjonsmekanismer krever andre måter å angripe problemet på. De hindrer ikke angrep på integritet, men rapporterer at dataene som er blitt angrepet ikke lenger er troverdige. Disse mekanismene analyserer hendelser fra bruker eller systemet. De kan også analysere selve dataene for å se om gitte forutsetninger for dataene fremdeles stemmer. Mekanismene kan rapportere både årsaken til problemet eller bare rapportere at disse dataene er korrupte.

Tilgjengelighet
Tilgjengelighet benyttes om det å kunne benytte den informasjon eller ressurs som ønskes. Tilgjengelighet er en stor del av systemdesign og stabiliteten i et sytem fordi et utilgjengelig system er minst like ille som ingen system. I forhold til sikkerhet gjelder tilgjengelighet i den forstand at noen bevisst prøver å sette et system ut av spill. Slike operasjoner kalles gjerne denial-of-service attacks og kan f.eks. gjennomføres ved å floode meldinger slik at en webserver bryter sammen. Disse angrepene kan være vanskelig å detektere fordi forskjellen på hva som er et bevisst angrep og hva som er bare en uvanlig hendelse er liten eller ingen.

2.7.2 Trusler og angrep

De tre grunnleggende delene av datasikkerhet; konfidensialitet, integritet og tilgjengelighet, brukes for å hindre trusler mot et system. Angrep defineres som faktiske tilfeller av disse truslene. Eksempler på trusler kan være:

Snooping
Snooping eller spionering er passiv lytting eller lesing av data når disse brukes i kommunikasjon, filer eller systeminformasjon som må hindres av

konfidensialitetsmekanismer. Et eksempel på dette er *Wiretapping* som brukes som betegnelse for passiv nettverksovervåkning.

Modifisering
Uautorisert endring av informasjon som må hindres av integritetsmekanismer. Målet med dette kan være bedrageri, sammenbrudd for tjenester eller tilgang til tjenester. Skriv endring av data hender som et resultat av at en entitet endrer informasjon. Man-in-the-middle er eksempel på et sånt type angrep og oppstår når en inntrenger leser en melding fra en sender og sender videre sin versjon til den opprinnelige mottakeren. På denne måten oppnås tilgang til informasjon uten at de involverte merker det.

Spoofing
Spoofing brukes om det å benytte seg av en falsk identitet for å få tilgang til data eller tjenester og må motvirkes ved hjelp av integritetsmekanismer (autentisering). Angrepene kan være både passive eller aktive og har som mål å få et system til å tro at den kommuniserer med en ”ekte” entitet. Et eksempel vil være det stadig voksende problemet, phishing som for eksempel kan være å sende en tilsynelatende ekte mail fra en brukers bank. Denne mailen instruerer så brukeren å gå inn på en falsk webside for å oppgi sine kontodetaljer.

Denial-of-service
Brukes ofte sammen med andre bedrageri-mekanismer og går utpå å nekte brukere tilgang til data. Dette kan beskyttes ved hjelp av mekanismer for tilgjengelighet. Denial-of-service i seg selv går ut på å gi en bruker uendelig forsinkelse når man prøver å aksessere en tjeneste.

2.7.3 Kryptering

Kryptering er betegnelsen på å skjule mening og kryptografi er dannelsen av hemmelige ”koder” eller nøkler. Kryptoanalyse er å knekke slike hemmelige ”koder” eller nøkler. Et kryptosystem brukes til å kryptere data og de originale dataene kalles klartekst og de krypterte dataene kalles siffertekst. For alle kryptosystemer vil målet være å gjøre det umulig å finne klarteksten utfra sifferteksten uten den kryptografiske nøkkelen. I kryptografiske systemer finnes det to hovedutgangspunkt: Symmetrisk og asymmetrisk kryptering.

Symmetrisk kryptering
I symmetrisk kryptering benyttes den samme nøkkelen for kryptering og dekryptering. Det vil si at mottaker og sender av data deler kryptografisk nøkkel. Symmetriske kryptosystemer deles gjerne inn i blokk- og flytkryptogram. Forskjellen på disse er at blokkryptogrammer tar blokker eller deler av klarteksten og genererer like store blokker med siffertekst. Flytkryptogrammer genereres utfra hele klarteksten på samme tid og produserer en siffertekst på samme lengde. Under følger noen eksempler på symmetriske kryptosystemer [58].

RC4
RC4 er et flytkryptogram som benyttes i populære protokoller som SSL (Internett-trafikk) og WEP (WLAN). Problemet med RC4 er at det er gammelt og ikke

optimalisert for dagens standarder. Det er blant annet optimalisert for 8-bits prosessorer. Flytkryptogrammer har ikke blitt prioritert de seneste årene og det synes klart at blokkryptorammene er det området der utviklingen går fremover [59].

DES
Data Encryption Standard, DES, ble utviklet i USA på 1970-tallet og lever i dag videre som Triple-DES eller 3-DES. Utviklingen av algoritmen var kontroversiell med hemmeligstemplede komponenter og konspirasjonsteorier rundt National Security Agencys involvering. Hver blokk i den originale DES er 64 bit og benytter en 56 bits nøkkel, noe som i dag betraktes som usikkert. Triple-DES ansees som praktisk sikkert i dag, men den kan angripes teoretisk. Fremgangsmåten som benyttes er at den bryter opp klarteksten i blokker av 64 bit og disse prosesseres gjennom 16 identiske steg eller runder. Triple-DES benytter samme fremgangsmåte, men gjør denne prosessen tre ganger [60].

AES
Rundt 1990, når det ble klart at DES ikke lenger kunne regnes som en sikker algoritme, begynte utviklingen av Advanced Encryption Standard (AES). Algoritmen som benyttes er Rijndael og har en blokkstørrelse på 128 bit. Nøklene som brukes kan enten være 128, 192 eller 256 bit. AES er i dag ansett som umulig å bryte i praksis og er det eneste (til nå) krypteringssystemet som benyttes av amerikanske myndigheter for hemmelige dokumenter samtidig som algoritmen er tilgjengelig for alle [61].

Asymmetrisk kryptering
I asymmetrisk kryptering, eller public-key kryptografi, benyttes det et nøkkelsett for kommunikasjonen. Hver entitet har to nøkler en privat og en offentlig nøkkel. Tanken er at den offentlige nøkkelen skal bli distribuert, mens hver enkelt skal beholde sin egen private nøkkel. Fordi en nøkkel er distribuert og en er privat må public key-kryptografi tilfredstille følgende tre krav [62]:

1. Det skal være lett å kryptere og dekryptere en melding med riktige nøkler.
2. Det skal være umulig å finne den private nøkkelen utfra den offentlige.
3. Det skal være umulig å finne den private nøkkelen utfra en gitt klartekst og korresponderende siffertekst.

Eksempler på public-key kryptosystemer er Diffie-Hellman [63] og RSA [64].

Hash-funksjoner
Hash-funksjoner er gjerne enveis funksjoner som brukes i sikkerhetsapplikasjoner for å gi meldingsintegritet og autentisering. Disse funksjonene tar en melding (tekststreng) med variabel lengde og genererer et resultat med en fast lengde. Den genererte strengen er gjerne kortere enn den opprinnelige strengen og benyttes derfor som en sjekksum eller forkortet melding. De mest brukte hash-funksjonene idag er MD5 og SHA-1. HMAC er også mye brukt og denne benytter seg av en kryptografisk nøkkel i sammenheng med enveis-funksjonen. På denne måten krypteres også selve hashen og gir dermed også autentisering av motparten [65].

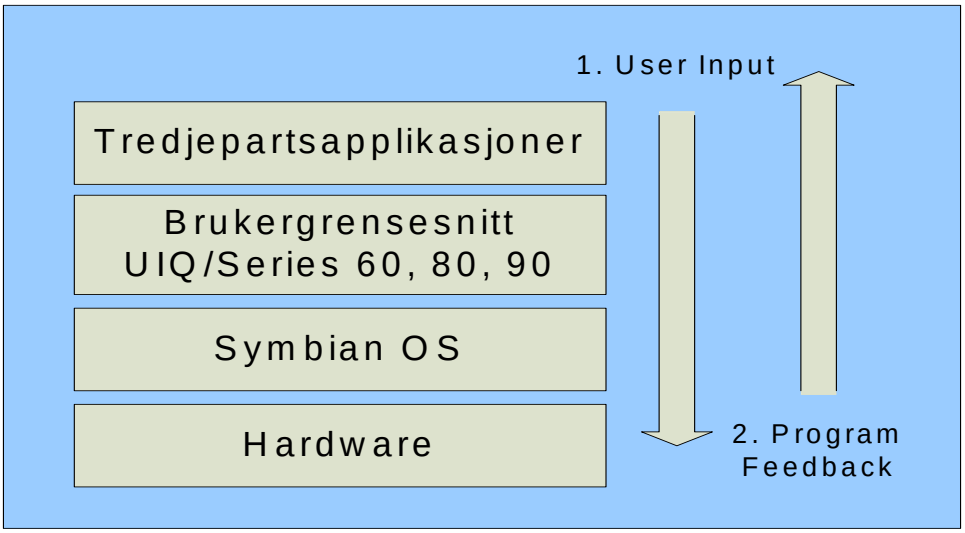
2.8 Håndholdte Enheter

Vår oppgave krever en håndholdt enhet som støtter kommunikasjon over UMTS og WLAN. Dette omfatter flere moderne mobiltelefoner og flere PDAer som har blitt utvidet med støtte for mobiltelefoni. Uttrykket MDA som av og til brukes, er en avart av PDA. Det er forkortelsen for PDA i Tyskland. Bærbare datamaskiner kan også støtte kommunikasjon over WLAN og UMTS med tilleggskort, eller med innebygd støtte for begge deler. Bærbare datamaskiner faller da utenfor vår definisjon på grunn av størrelsen, selv om de kan brukes til dette formålet.

Vi vil i dette kapittelet gå inn på de to mest fremtredende operativsystemene som brukes på håndholdte enheter. Først vil vi gå inn på det mest dominerende operativsystemet for mobiltelefoner (Symbian) før vi går inn på Windows Mobile som er et operativsystem for PDA og enkelte mobiltelefonmodeller.

2.8.1 Symbian OS

Symbian er et operativsystem designet for mobile enheter og ble opprettet av flere av de store aktørene innenfor trådløs kommunikasjon (Ericsson, Nokia, Motorola og Psion) i 1998. Grunnlaget var den stadig økende kompleksiteten i mobiltelefoner og dermed også behovet for et stabilt og anvendelig operativsystem for tredjepartsapplikasjoner. Symbian OS er grunnlaget for en rekke brukergrensesnitt i moderne telefoner. Eksempler på dette kan være de åpne plattformene UIQ (Sony Ericsson), Series 60, Series 80 og Series 90 (alle Nokia). Dette er på mange måter både fordelen og ulempen med Symbian og Java-baserte telefoner ettersom disse mobile enhetene kommer i mange ulike former (tasteoppsett osv) sikres et felles grunnlag for disse forskjellige i Symbian, problemet blir at det blir veldig vanskelig å designe generiske applikasjoner i jungelen av plattformer, brukergrensesnitt og APIer [66].

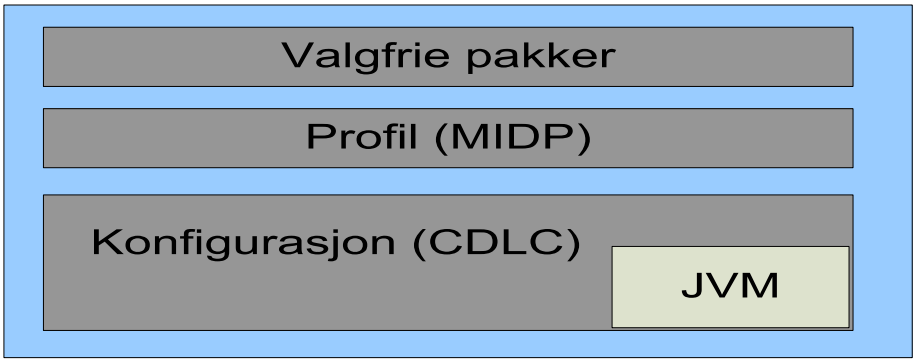


Figur 12 - Skjematisk fremstilling av oppbyggingen av operativsystem og brukergrensesnitt i mobile enheter [Egen figur]

2.8.2 J2ME: Grunnleggende oppbygging

Java Platform Micro Edition eller J2ME ble utviklet som følge av at Suns visjon om en felles Java-plattform for alle enheter ikke lot seg gjennomføre. Java-verden ble delt i tre deler, en for bedriftsmarkedet (J2EE), en for desktopapplikasjoner (J2SE) og en del for mobile enheter (J2ME). Fordi mobile enheter er et ganske vidt begrep vil man være avhengig av forskjellige implementering og bruksområder for J2ME. J2ME kan skreddersys for enheter eller grupper av enheter med en blanding av konfigurasjoner, profiler og valgfrie pakker. Det er faktisk sånn at J2ME i det store og det hele kun er en samling konfigurasjoner, profiler og valgfrie pakker [67].

I bunn av J2ME ligger, som ellers i Java, en Virtual Machine (JVM) spesielt tilpasset for mobile enheter. JVM brukes i mobile enheter til å kjøre MIDlets. MIDlets kjøres ved hjelp av en profil kalt Mobile Information Device Profile (MIDP). MIDP er basert på en konfigurasjon. Konfigurasjoner er egentlig kun en spesifikasjon av et rammeverk, og den mest brukte konfigurasjonen for denne generasjons mobiltelefoner er Connected Limited Device Configuration (CDLC). I CLDC ligger spesifikasjonen for rammeverk til enheter med begrensede ressurser som tilfellet er med en mobiltelefon eller en personsøker. MIDP spesifiserer videre rammeverk for den spesifikke enheten. MIDP 2.0 er den foreløpige siste versjonen laget og denne benyttes i alle nyere mobiltelefoner. MIDP 2.0 inneholder APIer for brukergrensesnitt, mediehåndtering, ende-til-ende sikkerhet, nettverksressurser m.fl. I tillegg til dette kan man også implementere valgfrie pakker [67] [68] [69] [70] [71] [72].



Figur 13 - Skjematisk tegning av oppbyggingen i Symbian OS [Egen figur]

2.8.3 J2ME: Sikkerhet

For å sikre telefonen mot angrep gjennomfører Symbian OS verifisering av alle applikasjoner som skal kjøres. Dette betyr at hver eneste applikasjon som installeres og kjøres må ha et sertifikat som signerer applikasjonen. Dersom dette ikke er til stede må brukere godkjenne de fleste operasjoner applikasjonen gjør. Disse operasjonene kan være tillatelser fra bruker til å benytte seg av nettverksressurser, tilgang til filer osv. JVM opererer i en såkalt sandbox som ikke nødvendigvis har tilgang til viktige systemressurser. Et annet poeng er at trådløse signaler kan snappes opp, og sammen med begrenset minne og prosessorkraft i den mobile enheten kan det utgjøre en mulig trussel [73].

Klientsiden er den desidert svakeste linken i et klient-server system, og det må brukes ressurser for å sikre klienten. J2ME gir skalerbarhet og mulighet for ende-til-ende løsninger uten mellomliggende servere. Teknologier som WAP gjør ikke det. I tillegg gir J2ME mulighet til å lagre og prosessere informasjon lokalt på enheten. Dette medfører redusert nettverkstrafikk noe som er en fordel både i forhold til bruk av båndbredde (kostnader, feil osv.) og i forhold til eksponering av data. Det er en kjennsgjerning at J2ME, i form av CLDC og MIDP 2.0, og selve mobiltelefonen har et begrenset reportoar i forhold til prosesseringskraft, minnekapasitet osv. Et mulig tiltak for å håndtere dette litt bedre kan være å differensiere nettverkstrafikk slik at man på forhånd sjekker hva slags data som skal transporteres og hva slags nettverk dataene skal transporteres over. På denne måten kan man minimere nødvendigheten av kraftige krypteringsalgoritmer dersom nettverkstrafikken for eksempel kun skal gå på et internt nett. De store utfordringene for sikkerhetsapplikasjoner i J2ME på mobile enheter vil være [73]:

- **Hastighet**: CPU på en mobil enhet og spesielt mobiltelefon er ikke veldig stor. I tillegg er Java i utgangspunktet et høynivå programmeringsspråk, og dermed ikke et raskt programmeringsspråk.
- **Small Footprint:** Når man opererer i et begrenset miljø som en mobil enhet med J2ME, gjelder det å skrive enkel og effektiv kode.
- **Støtte for flere forskjellige krypteringsalgoritmer:** For å få en effektiv applikasjon må flere typer algoritmer benyttes. Dette vil si at man bør differensiere behovet for kryptering slik at det å sende en enkel melding på bedriftens eget nettverk og det å sende en sensitiv melding ut til en ekstern mottaker ikke behandles likt i programmet. Dette vil spare ressurser hos enheten.
- **Enkle APIer:** Det er viktig at tilleggs-APIene som brukes ikke blir for komplekse og krever for mye ressurser.
- **Lette nøkkelidentifikasjon og serialisering:** Nøkler må identifiseres og matches i begge ”ender” av kommunikasjonen. Generering av nøkler kan være en tidkrevende operasjon En mulig løsning kan være å generere nøkler på serversiden og så distribuere dem ut.

2.8.4 Microsoft Windows Mobile 5.0

Microsoft Windows Mobile 5.0 er Microsofts siste operativsystem for håndholdte enheter som Pocket PCer, Smartphones og Bærbare Mediasentre. Operativsystemet ble lansert 9. Mai 2005 og opererte lenge under navnet ”Magneto”. Windows Mobile leveres med Office Mobile, en programpakke som inkluderer pocket-versjoner av kjente programmer som Word, Excel og PowerPoint. I tillegg til dette og en rekke andre funksjoner finnes også versjoner av Outlook og MediaPlayer på plattformen [74] [75] [76].

Neste versjon av Windows Mobile er under utvikling og ble kunngjort av Microsoft i desember 2005. Under foreløpig navn Photon vil dette innebære en fusjon av Windows Mobile og MS Smartphone operativsystemet. Det forventes at det ferdige produktet vil bli hetende Windows Mobile 6.0. Lite er sagt foreløpig om Photon men det forventes bl.a. at operativsystemet vil medføre store forbedringer med hensyn til batterilevetid.

Det forventes i tillegg at en versjon 5.0 Second Edition vil komme på markedet i 2006 [77] [78].

2.8.5 .NET Compact Framework

.NET Compact Framework (CF) er designet for å kjøre på mobile enheter som PDAer og mobiltelefoner. CF er en versjon av den vanlige .NET-plattformen og siste versjon, versjon 2.0, ble sluppet høst 2005. Rammeverket gir mulighet for utvikling av applikasjoner på Windows Mobile operativsystemet. CF benytter seg av noen av klassebibliotekene fra den store .NET-plattformen og noen egendefinerte som kun gjelder sammen med Windows Mobile.

2.8.6 Aktuelle enheter

Valget mellom Windows Mobile og Symbian er ikke nødvendigvis så lett. Argumentene som taler for Symbian er omfanget og det store markedet for mobiltelefoner. Fokuseringen på forbrukere er også den store svakheten til Symbian. Det finnes ikke ting som filsystem, og J2ME i seg selv støtter ikke tilgang til terminalspesifikk hardware. Det som først og fremst mangler i Symbian er en felles plattform på tvers av alle enheter som benytter operativsystemet. Den største andelen Symbian-enheter er Nokia-telefoner med Series 60-plattform som solgte mer enn 25 millioner enheter i 2005 [79].

Microsoft Windows Mobile har hatt en gradvis vekst og tiltrekker seg nå flere utviklere enn noen annen plattform for mobile enheter. Det er først og fremst bedriftsløsninger som benytter seg av dette operativsystemet. Den store fordelen med Windows Mobile er at alle enheter som har dette OSet også til enhver tid vil ha det samme grensesnittet. Dette er ikke tilfelle i andre operativsystemer, som Symbian. I forhold til fremtidige versjoner har Microsoft forpliktet seg til at 95 % av applikasjonene som utvikles for nye versjoner skal være bakoverkompatible [79].

Det viktigste i forhold til vår problemstilling vil uansett være at enheten har støtte for både WLAN og UMTS. Det lanseres nå flere telefoner med støtte for dette fra flere produsenter. Nokia kommer med sine E60 [80], E61 [81] og E70 [82]. Sony Ericsson har lansert sin nye P990 [83]. I forhold til litt større håndholdte PDAer finnes allerede QTEK9000 [84] og HP IPAQ h6340 [85]. Ulempen med sistnevnte er at den kun har GPRS-støtte, ikke UMTS.

2.9 Våre Valg

I dette kapittelet vil vi gå nærmere inn på valg av teknologier som skal benyttes videre under arbeid med design og PoC.

2.9.1 Roaming

Det finnes flere veier å gå for å få til en roamingløsning mellom UMTS og WLAN for håndholdte enheter. Store ressurser benyttes i dag for å klargjøre løsninger fra operatørsiden for å oppnå dette. Noen av disse løsningene (UMA) er avhengig av en endring både hos mobiloperatøren og mobilprodusentene. Andre løsninger (MIP og SIP) tilbyr arkitektur for å muliggjøre roaming i nettverk uavhengig av underliggende linklag. Dette betyr at begge disse løsningene gir en stor frihet i forhold til brukersituasjoner og hvilken nettverksteknologi som benyttes.

Vi har valgt å benytte oss av teknologier som er uavhengige av underliggende nettverk. Det vil si at det eneste kravet vil være at det er et IP-nettverk som benyttes. For å oppnå dette kan både CellularIP, Mobile IP (MIP) eller Session Initiation Protocol (SIP) benyttes. Vi har valgt å basere vår løsning på SIP fordi den opererer på applikasjonsnivå og er en standardisert kommunikasjonsprotokoll. Designet vi foreslår baseres på SIP, men vi velger å implementere egendefinerte meldinger for å holde kompleksiteten så lav som mulig. Grunnen til dette er at det kun er meldingsutveksling som går mellom IMATIS og de håndholdte enhetene, ikke medieoverføring. Det ville være bortkastet båndbredde og tid å implementere en fullverdig SIP-protokoll med tilhørende INVITEs og ACKing. Likevel vil det være ønskelig å ”klargjøre” systemet for en mulig utvidelse til SIP-protokollen for å senere kunne håndtere IP-telefoni ved hjelp av VoIP. I tillegg til å basere meldingsformatet på SIP ønsker vi å benytte Birdsteps roamingklient for mobile enheter for selve håndteringen av nettverkstilkoblinger. Birdsteps løsninger baserer seg på Mobile IP.

Fordeler ved å basere meldingsformatet og design på SIP:

- SIP er lettere å implementere fordi det medfører ingen endring i underliggende nettverksutstyr.
- SIP har en fremtredende rolle i fremtidens mobile nett (IMS).
- SIP er basis for dagens VoIP-løsninger.
- Meldingsformatet i SIP er enkelt å håndtere og passer dermed godt til IMATIS-meldingsformatet.
- SIP er en standardisert kommunikasjonsprotokoll som er skreddersydd for proprietære løsninger.
- Den grunnleggende arkitekturen i SIP minner mye om IMATIS-arkitekturen. Det vil si et klient-/tjener- miljø.
- De ulempene som måtte være i forhold til treg handover påvirker ikke i nevneverdig grad et meldingsbasert system som IMATIS.
- Tredjepartsløsninger for håndtering av lokasjon i mobilnettet vil være enklere å implementere på applikasjonslaget. En slik løsning kan være Map Solutions [97] løsninger for posisjonering.

2.9.2 Birdstep SmartRoaming

SmartRoaming er en roamingklient fra Birdstep basert på Mobile IP. SmartRoaming gir mulighet for å beholde sin nettverksforbindelse når man bytter nett basert på prekonfigurerte profiler. Disse profilene kan settes opp med prioriteringsnivå slik at man kobler til et spesielt WLAN med en gitt SSID som første prioritet. Dersom dette ikke er tilgjengelig kan man koble til via UMTS. SmartRoaming håndterer nettverksidentifikasjon, brukerautentisering og nettverksforbindelser for en mobil enhet. Dette gjøres sømløst uten brukerinteraksjon slik at bruker hele tiden kan fortsette sitt arbeid uten avbrytelser i pågående sesjoner. Dette medfører at helsepersonell kan utføre sitt arbeid uten å måtte håndtere hva slags nettverk om er tilgjengelig i en gitt lokasjon og heller ikke trenger å velge hvilken forbindelse som skal benyttes. Birdstep SmartRoaming er tilgjengelig på flere mobile enheter. Noen eksempler er Nokia 9300i, Nokia E-serie og Sony Ericsson P990 [29], [86].

Ting som ikke vil bli belyst i PoC-testing, er konfigureringen av nettverksforbindelser. Dette er teoretisk mulig, men blir ikke testet fordi en enhet med WLAN- og 3G-støtte ikke er tilgjengelig for oss. Løsninger for roaming med slike enheter er på markedet og i bruk. Det antas derfor at en slik løsning vil være mulig.

2.9.3 Sikkerhet

Fire aspekter er viktig i forhold til å sikre innhold i meldinger i et trådløst miljø:

Autentisering
Dette vil i vår oppgave si at brukeren (klienten) autentiseres mot et sentralt system. Dette kan gjøres ved hjelp av digitale sertifikater, men vi har valgt å gjøre dette ved å autentisere den mobile bruker når den logger seg på fra UMTS-nettet. Dette gjøres ved hjelp av brukernavn og et kryptert passord. Autentiseringen med kryptert passord gjøres for å hindre tilfeller av spoofing i systemet. Grunnen til valget er at vi vil holde kompleksiteten på et så lavt nivå som mulig for den mobile enheten. Passordet krypteres med gjeldende krypteringsnøkkel for meldingsinnhold. I forhold til autentisering av server velger vi at den til enhver tid trustes.

Dataintegritet
Meldingene som går fra server og klient må integritetssjekkes i form av en sjekksum for hele XML-meldingen. Dette gjøres for å hindre modifikasjon av data under transport. Vi velger å implementere en deteksjonsmekanisme da det vil være umulig å implementere en forhindringsmekanisme som skal beskytte en melding som går over lufta i UMTS-nettet. Vi velger derfor å gjøre dette ved hjelp av en hash-algoritme; SHA-1. Forhindringsmekanismen benytter seg av SHA-1 for å eventuelt varsle systemet dersom meldingen er blitt forandret under transport. Mekanismene for å sjekke dataintegritet skal kjøres både på klient og på server.

Konfidensialitet
Dette omhandler kryptering av meldingsinnhold. Vi kan velge både asymetrisk og symmetrisk kryptering. Vi velger å benytte oss av symmetrisk kryptering, igjen med bakgrunn i at vi ønsker å holde kompleksiteten i klientapplikasjonen på et så lavt nivå

som mulig. Som krypteringsalgoritme velger vi AES som er allment tilgjengelig og samtidig den mest brukte og sterkeste av dagens symmetriske krypteringsalgoritmer.

Non-repudiation
Meningen med non-repudiation er at man skal vite at meldingen kommer fra rett person (autentisering) og at den ikke er snappet opp og sendt på nytt av mellomliggende enheter (man-in-the-middle). Vi velger å stole på at meldingen kommer fra meldingsserveren til enhver tid og fra klienten ute i nettet når svarmelding skal sendes i og med at klienten allerede er logget på med brukernavn og passord. I tillegg eksisterer det i dag en timestamp i selve meldingsinnholdet som vi velger å bruke for å sjekke tidsopprinnelsen.

2.9.4 Mobil enhet for PoC

Vi har valgt å teste i et J2ME-miljø pga mobiltelefonens markedsposisjon og videreutviklingsmetodene som finnes i forhold til Java på mobiltelefoner. I tillegg til dette har vi også valgt å benytte oss av to enkle APIer[8] som er designet for mobile enheter; BouncyCastle Crypto API og kXML som er designet for XML-håndtering. Disse er valgt fordi de tilbyr mye av det som er essensielt når man skal implementere applikasjoner på mobile enheter. Dette vil i første rekke være å forsøke å holde prosessering på et så lavt nivå som mulig og at applikasjonen setter fra seg så lite fotavtrykk som mulig. Dvs. tar liten plass både i forhold til programmets fysisk størrelse på minnet og antall operasjoner som krever prosesseringskraft i enheten.

2.9.5 Oversikt over valgte teknologier

Tabell 3 gir en oversikt over teknologier som er blitt valgt som grunnlag for videre arbeid i design og Proof of Concept.

Tabell 3 - Oversikt over valgte teknologier

Teknologi	Valg	Kort begrunnelse
Roamingmetode	SIP/MIP	SIP: meldingsformat, MIP: mobilitetsløsninger
Mobilt miljø	Symbian OS/J2ME	Utbredelse, markedsposisjon
Autentisering	Brukernavn/kryptert passord	Ressurssparende
Dataintegritet	SHA-1	Utbredelse, ressurssparende
Konfidensialitet	AES	Styrke

[8] Application Programming Interface

3 DESIGN

Dette kapittelet gir en oversikt over case scenarioer og knytter dette opp mot tenkt design av systemet.

3.1 Overordnet Case scenario

Sentralt i strukturen står IMATIS meldingstjener som inneholder all informasjon og status for oppdrag, alarmer o.l. Den inneholder web-applikasjoner for administrering og bestilling av oppdrag samt grensesnitt for kommunikasjon med mobile enheter. Denne web-applikasjonen er lokalisert på hvert sengetun. Personell på St. Olavs Hospital er i dag utstyrt med en IP-telefon enten fastmontert eller trådløs. Noe personell, avhengig av funksjon og rolle, er utstyrt med mobiltelefon i tillegg. Meldingene som sendes ut fra meldingstjeneren kan sendes på to måter. Dersom enheten er pålogget (på WLAN i ny bygningsmasse) sendes en melding til portøren på IP-telefon. Dersom enheten ikke er pålogget (i gammel bygningsmasse) sendes en SMS til mobiltelefon. SMS-meldingen kvitteres med vanlig svarfunksjon i mobiltelefonens SMS-grensesnitt. Portøren må manuelt skrive svarmeldingen basert på ID i selve oppdraget. I dag må store deler av meldingene sendes på SMS [87].

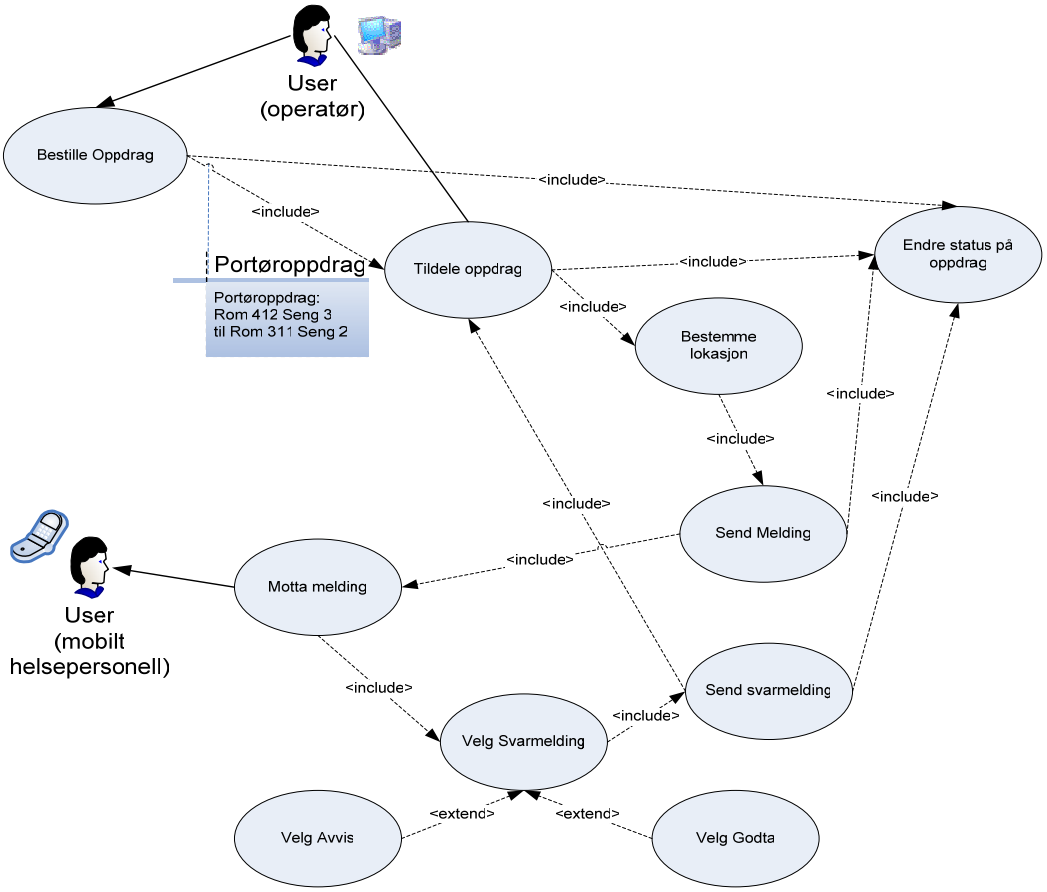
Figur 14 - Oversikt over et sengetun på sykehus [82]

I utgangspunktet er ønsket å samle kommunikasjonen med trådløse enheter til en enhet for å slippe to ting. Det første vil være at helsepersonell vil slippe å til enhver tid ha to enheter på seg. Den andre gevinsten vil være at personellet slipper å manuelt taste inn svarmelding med tilhørende ID. Dette forutsetter utvikling av en roamingløsning for den aktuelle enheten. Enheten må kunne bytte nett fra WLAN og ut i et offentlig mobilnett uten at brukeren trenger å gjøre noe. På samme måte må enheten oppdatere informasjon i lokasjonsserveren slik at meldingsserveren hele tiden vet hvor og på hvilket nett enheten befinner seg. I forhold til alarmering av spesielle roller, vil det være gunstig å ha muligheten å ta med seg vakttelefonen hjem.

3.1.1 Case: Portøroppdrag

Ved St. Olavs Hospital er det gjennomsnittlig 500 portøroppdrag i døgnet [88]. Disse portøroppdragene fordeles til portører som er utstyrt med en mobil enhet. Denne mobile enheten roamer uavhengig av brukerinteraksjon og systemet vil hele tiden ha oversikt over denne enheten selv når den befinner seg i gammel bygningsmasse. Portøroppdrag kan bestilles på to måter, via en webapplikasjon i portalen eller kundesenteret.

Operatører på kundesenteret håndterer tildeling av oppdrag ved å overvåke posisjon og rolle for hver enkelt portør. Denne informasjonen benyttes sammen med et flagg i lokasjonsdatabasen for å bestemme hvorvidt helsepersonellet befinner seg i gammel eller ny bygningsmasse. Oppdraget mottas så av meldingstjeneren som sender gjeldende melding til portøren. Meldingen mottas så hos portøren i nettet som kan velge å godta eller avvise oppdraget. Godtas oppdraget sendes svarmelding og oppdraget legges til i portørens arbeidsliste. Dersom oppdraget avvises gjentas prosessen med tildeling av oppdrag og ny melding sendes til ny portør [89].



Figur 15 - Usecase-diagram for portøroppdrag [Egen figur]

Forklaring til diagrammet (Figur 15)

Endre status på oppdrag: Oppdragstatus kan være ”oppdrag levert til portør”, ”portør bekreftet oppdrag”, ”oppdrag utført” og ”oppdrag ikke sendt”.

Bestille oppdrag: Oppdrag kan bestilles fra operatør eller fra mobilt personell med håndholdt telefon. Status på oppdrag vil da være satt til ”oppdrag ikke sendt”.

Tildele oppdrag: Operatør i kundesenter tildeler oppdrag utfra liste over tilgjengelige portører utfra lokasjonsinformasjon og rolle. Med tilgjengelige menes at portøren er logget på i systemet og antall oppdrag som finnes i portørens arbeidsliste. En portør kan maksimalt ha tre aktive oppdrag om gangen.

Bestemme lokasjon: Når operatøren har tildelt oppdrag sjekkes et flagg i databasen om hvorvidt portøren befinner seg på WLAN eller UMTS.

Send Melding: Dersom portøren befinner seg på WLAN (ny bygningsmasse) sender meldingsserveren ut meldingen på vanlig måte. Dersom portøren befinner seg på UMTS (gammel bygningsmasse) sendes meldingen fra meldingsserver til en RedirectServer som påfører nødvendig kryptering og sender den videre ut til portøren. Status på oppdrag vil da være satt til ”oppdrag levert til portør”.

Motta melding: Portøren mottar meldingen fra meldingsserver, meldingen vises på den mobile enheten.

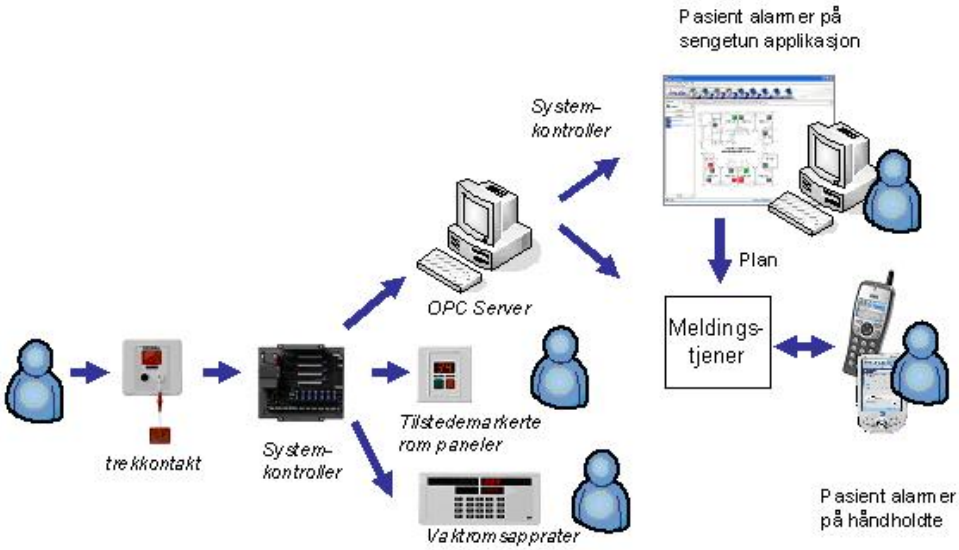
Velg svarmelding: Portøren velger enten godta eller avvis som svar på oppdraget.

Send svarmelding: Portøren sender sitt svar og meldingen håndteres i kundesenteret. Dersom portøren avviser tildeles oppdraget en ny portør. Dersom portøren godtar settes status for oppdrag til ”portør bekreftet oppdrag”. Når oppdraget er utført sendes en kvittering fra portøren på nytt og status settes til ”oppdrag utført”.

3.1.2 Case: NurseCall

NurseCall initialiseres ved hjelp av et pasientsignal som utløses ved å trekke i snoren på anropspaneler også kalt trekkontakter. Alarmene vises så på vaktromsapparater og rompaneler. Deretter sendes alarmeldingen videre til meldingstjeneren for visualisering på telefoner, PDA eller PCer. Meldingen sendes til meldingstjeneren via OPC[9] Server og vises på sengetunapplikasjon. Meldingen går deretter ut til helseressursen som er dedikert til den gitte pasienten eller rommet. Denne anropsplanen administreres via applikasjon på sengetun og helsepersonellet som kan tilkalles består av alle som tidligere har meldt seg inn på det aktuelle sengetunet. Case-beskrivelsen er delvis hentet fra [88] [89].

[9] OLE for Process Control – Standard for sanntidskommunikasjon mellom windows-baserte applikasjoner og prosesskontroll hardware og software.



Figur 16 - Signalflyt for case NurseCall [89]

Figur 17 - Pasientsignal utløses ved at pasient drar i snoren på anropspanelet [88]

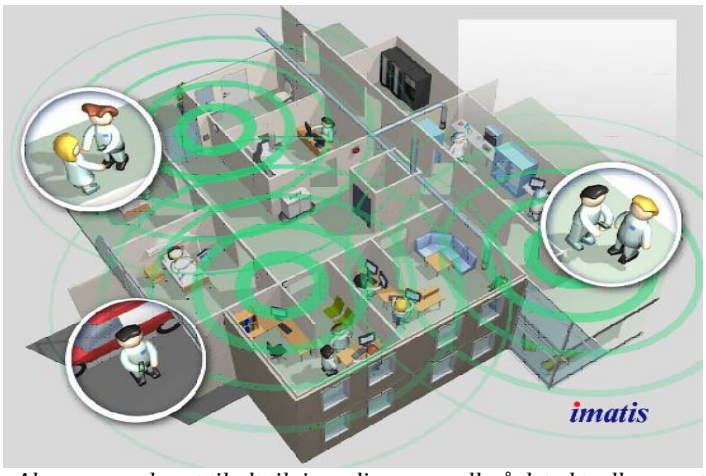
Figur 18 - Alarmen mottas og sendes ut til dedikert helsepersonell [88]

Med sin trådløse enhet kan helseressursen deretter enten godta eller avvise alarmen. Hvis den avvises sendes den videre til neste helseressurs på avdelingen. Dersom ingenting skjer times alarmen ut etter en forhåndsdefinert tid (f.eks. 30 sekunder). Hvis ingen godtar sendes den ut som en fellesmelding til alle på avdelingen. Alarmen blir ikke fjernet før noen aksepterer den.

Figur 19 - Helseressursen mottar alarmen og godtar den [88]

Sykepleieren tilstedet markerer seg på rom ved å trykke på grønn knapp på sengeromspanelet. Alarmen fjernes deretter fra andre tilstedemarkeringspaneler, vaktromsapparat og håndholdte enheter. Dersom situasjonen krever mer personell tilstedet, f.eks. ved hjertestans, tilkalles disse ved å trekke i snoren på anropspanelet mens tilstedemarkeringen er aktiv. Disse hasteanropene vises på vaktromsapparat, tilstedemarkerte rompaneler, sengetunsapplikasjon og håndholdte enheter. Alle ressurser knyttet til det aktuelle sengetunet mottar denne meldingen som er merket ”HASTER”. Disse meldingene kan bekreftes men ikke avvises.

Figur 20 - Helseressursen ankommer rommet og finner ut at situasjonen er alvorlig [88]



Figur 21 - Alarmen sendes ut til alt tilgjengelig personell på det aktuelle sengetunet [88]

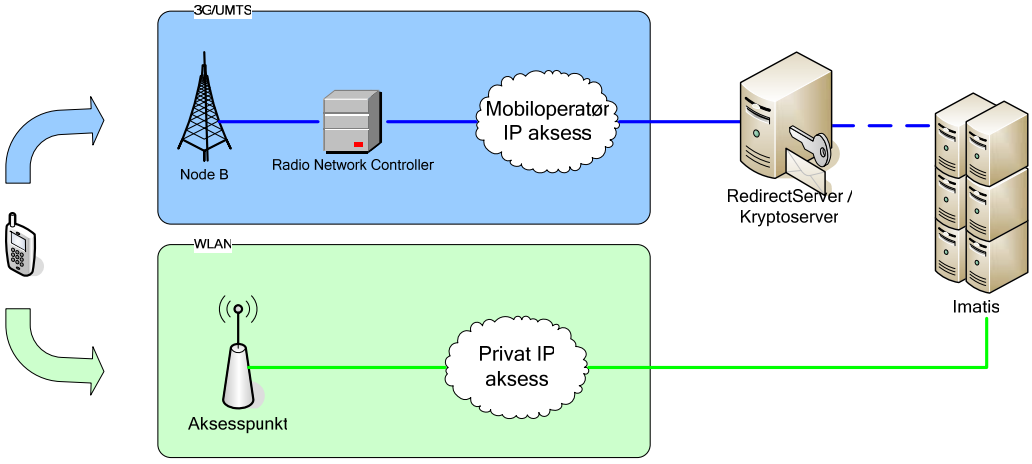
Figur 22 - Tilgjengelig personell ankommer rommet [88]

3.2 Design av Roaming

Vårt forslag er at sykehusets WLAN benyttes så lenge det er tilgjengelig. Når sykehusets WLAN ikke lenger er tilgjengelig skifter brukeren til UMTS. I forhold til gratis WLAN hotspots som brukerne kan koble seg til på utsiden velger vi å ikke gjøre dette av sikkerhetshensyn. I tillegg til sikkerhetshensyn vil også slike hotspots medføre problemer i forhold til adressering siden det vanligste vil være å benytte NAT på innsiden i slike nett. På denne måten finnes det kun to nettverksprofiler som brukerne har mulighet til å benytte: Sykehusets WLAN og UMTS. Grunnlaget for valget av hvilken nettverksprofil som skal benyttes vil være signalstyrken i sykehusets WLAN.

3.2.1 Overordnet systemarkitektur

Vi velger å innføre en egen modul som tar seg av all meldingsutveksling som foregår mellom enheter i UMTS og IMATIS. Dette innebærer at det blir satt opp en egen server (RedirectServer) mellom enheten og IMATIS, slik at den kan ta seg av trafikkflyten fra UMTS.



Figur 23 - Overordnet arkitektur for meldingsutveksling [Egen figur]

RedirectServer skal sammen med KryptoServer stå for all database / meldingsstyring / kryptering / nøkkeldistribusjon. Grunnen til at det er satt opp en egen KryptoServer for å ta seg av kryptering og nøkkeldistribusjon er for å få mest mulig fleksibilitet i systemet. Dette vil fordele ressurskrevende oppgaver som kryptering og dekryptinger og samtidig forenkle muligheten for videre utvidelser. Sammen med RedirectServer og KryptoServer ligger det en database, RoamingDB, som har oversikt over aktive brukeres IP-adresse. Brukernes IP-adresser mappes mot UPN som er en unik identifikator for hver enkelt bruker. RoamingDB oppdateres med gjeldende IP-adresse for enheter når enhetene registrerer seg på UMTS.

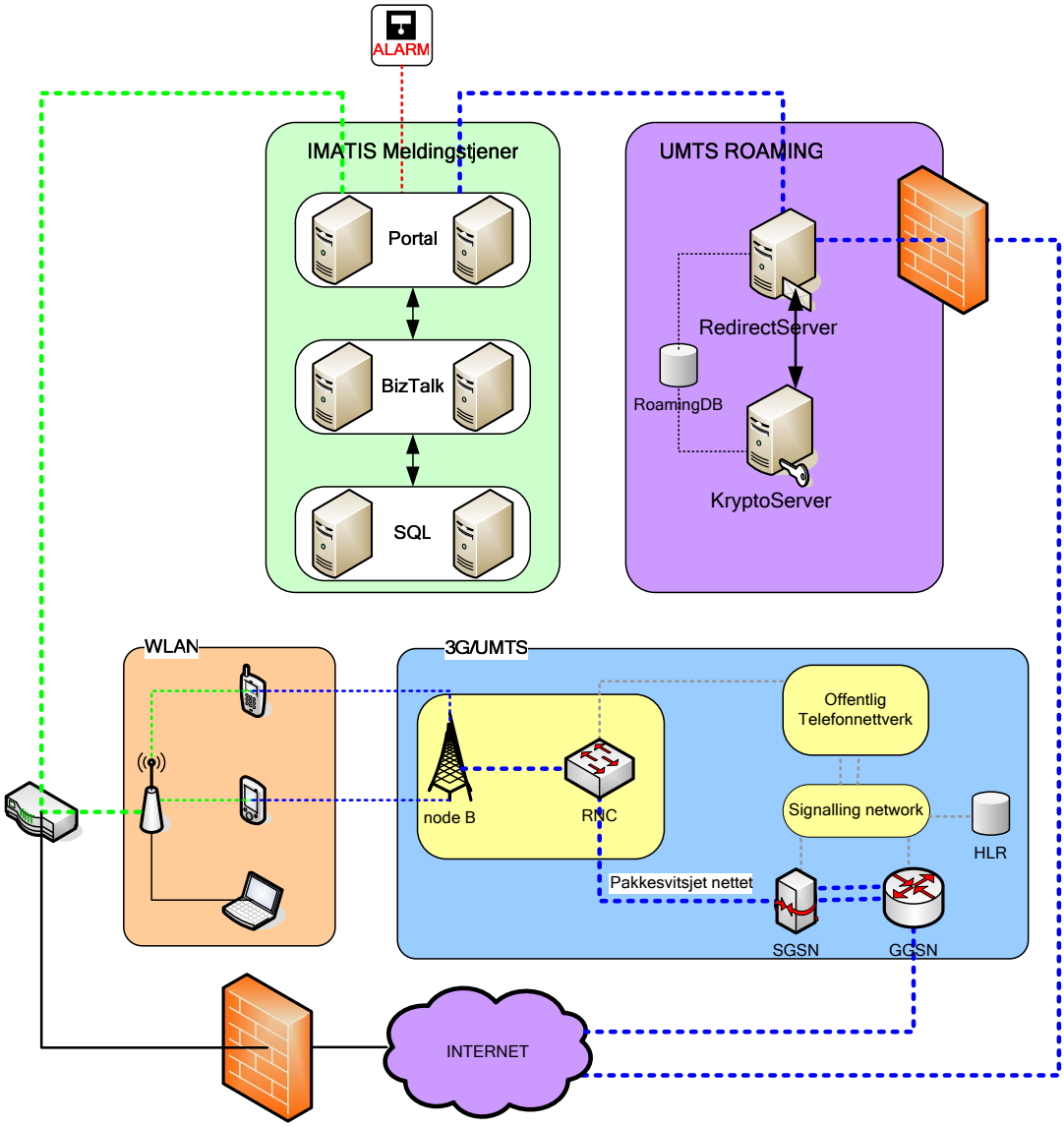
I IMATIS Meldingstjener, finnes det logikk som bestemmer hvem det er som skal være mottaker for meldinger. Dette kan gjøres ved å legge de aktive rollene som er på vakt i en liste og ta de fortløpende, en etter en. Ellers er det også mulighet å la dette styres av lokasjon i forhold til hvor oppdraget/alarmen er. Det vi ønsker å gjøre er å benytte RedirectServer som en mellomstasjon for meldinger som skal ut til enhetene over UMTS, mens IMATIS tar seg av kommunikasjon på WLAN.

IMATIS bruker den eksisterende logikken til å finne ut hvilken rolle/id som kan være mottaker når en ny melding skal leveres. Den videre veien for meldingen bestemmes av hvorvidt den befinner seg på WLAN eller UMTS. IMATIS-databasen inneholder et flagg som indikerer om enheten er pålogget et aksesspunkt i WLANet. Dette flagget er satt til 0 dersom enheten ikke henger på et aksesspunkt. Dersom dette flagget ikke er satt, altså 0, sendes meldingen til RedirectServer som håndterer adressering og sending til enheter på UMTS. Innkommende meldinger fra UMTS håndteres på vanlig måte i IMATIS etter at de er mottatt og håndtert i RedirectServer. Dersom flagget er satt til 1 (enehten henger på et et aksesspunkt) behandles meldingen på vanlig måte i IMATIS.

3.2.2 Detaljert Systemarkitektur

Systemet for å håndtere roaming til UMTS er tenkt å bestå av tre enheter: RedirectServer, KryptoServer og en database kalt RoamingDB. Under følger en

overordnet beskrivelse samt en skjematisk oversikt over disse enhetene og tilknytningen til IMATIS.



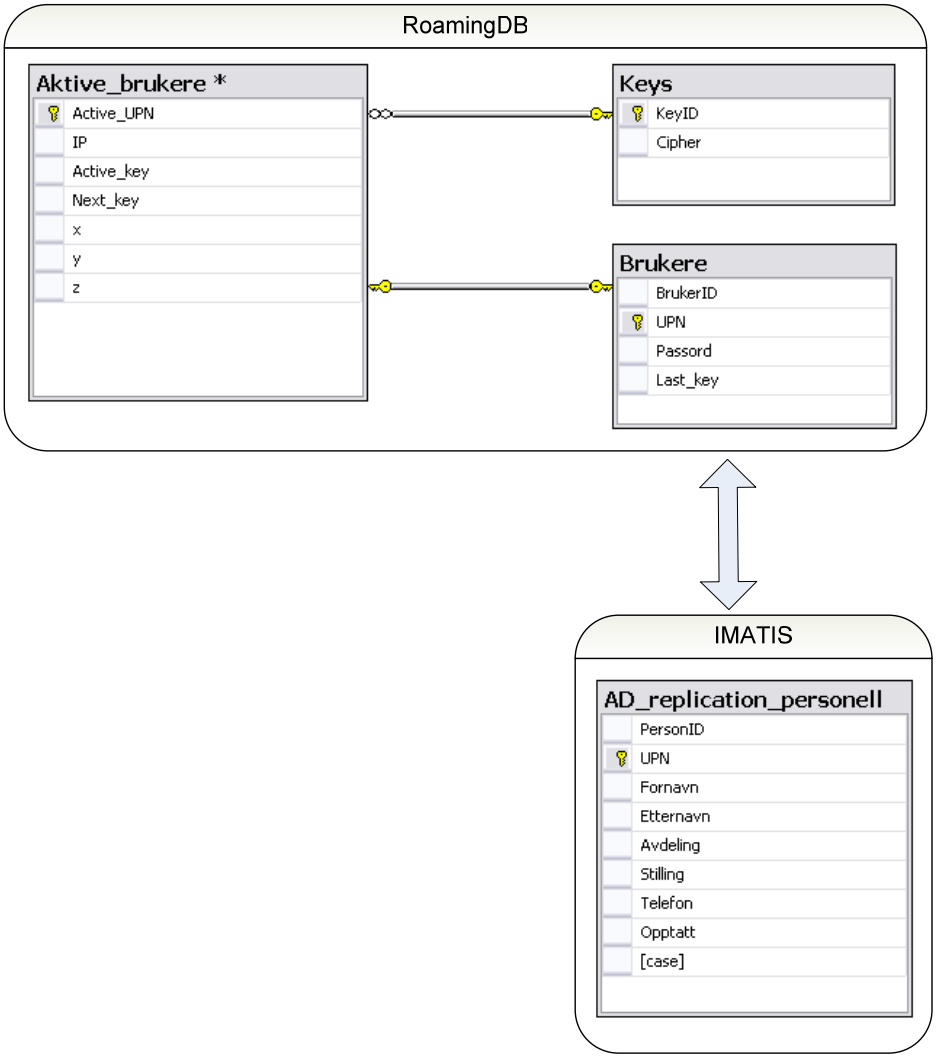
Figur 24 - Skjematisk oversikt over systemet [Egen figur]

RedirectServer
RedirectServer skal håndtere meldinger som benytter UMTS. Den benytter seg av RoamingDB og KryptoServer for å håndtere kommunikasjon mellom bruker (klient/enhet) og server. RedirectServer skal også håndtere kommunikasjonen med IMATIS.

KryptoServer
KryptoServeren har to hovedoppgaver: Kryptering/dekryptering av meldingsinnhold og generering av sjekksummer. For å håndtere kryptering av meldingsinnhold benyttes AES sammen med en nøkkel på 32 tegn, det vil si 256 bits. Kryptoserveren skal

kryptere meldingsinnholdet for utgående meldinger og dekryptere innholdet for innkommende meldinger. I tillegg skal Kryptoserveren stå for generering av sjekksum til meldinger. Denne sjekksummen genereres ved hjelp av SHA-1.

RoamingDB
Databasen vil inneholde en speiling av personelldatabasen i IMATIS. I tillegg vil den inneholde en tabell som viser enhetene som er tilkoblet via UMTS i et gitt tidspunkt. Denne databasen vil også da knytte enheten til sin krypteringsnøkkel. Databasen vil også inneholde en liste over tilgjengelige nøkler som kan velges når krypteringsnøkler skiftes ut for en enhet.



Figur 25 - Roaming Database [Egen figur]

Forklaring til databasen

AD_replication_Personell: Dette er en bit av IMATIS-databasen. Herfra hentes alle oppføringer og personellet som er registrert i Personell-databasen må også registreres i

brukerdatabasen i roamingmodulen med samme UPN. UPN er en unik adresse for alt personell bestående av brukernavn og domene.

Brukere: Her ligger alle brukere som er registrert i IMATIS-databasen.

Aktive_Brukere: Her ligger alle brukere koblet til via UMTS.

Keys: Inneholder nøkler som kan benyttes under kryptering.

3.3 Meldinger og meldingsutveksling

Vi velger å definere meldingene som går mellom RedirectServer og klientene.

REGISTER: Sendes fra den mobile enheten når den flytter seg fra WLAN til UMTS. Meldingen etterfølges av brukernavn og kryptert passord. Brukernavnet er satt sammen av et brukernavn og tilhørende domene.
Eksempel: *REGISTER pav.singh@st.olav.sengetun2 y2r3u99xppt3tra*

CHKSUM: Sendes etter alle meldinger som en egen melding. Etterfølges av en sjekksum generert for gjeldene XML-melding.
Eksempel: CHKSUM mulv8so7ljpj5m7yyo/0oenziji=

LOC: Bruker sender lokasjonsoppdatering hvert halve minutt. Meldingen etterfølges av gjeldende x- og y-koordinat.

200 OK: Ack-melding som sendes som svar på REGISTER til klient dersom login er gjennomført uten feil.

500 Failure: Generell feilmelding. Fører til re-send av sist sendte melding.

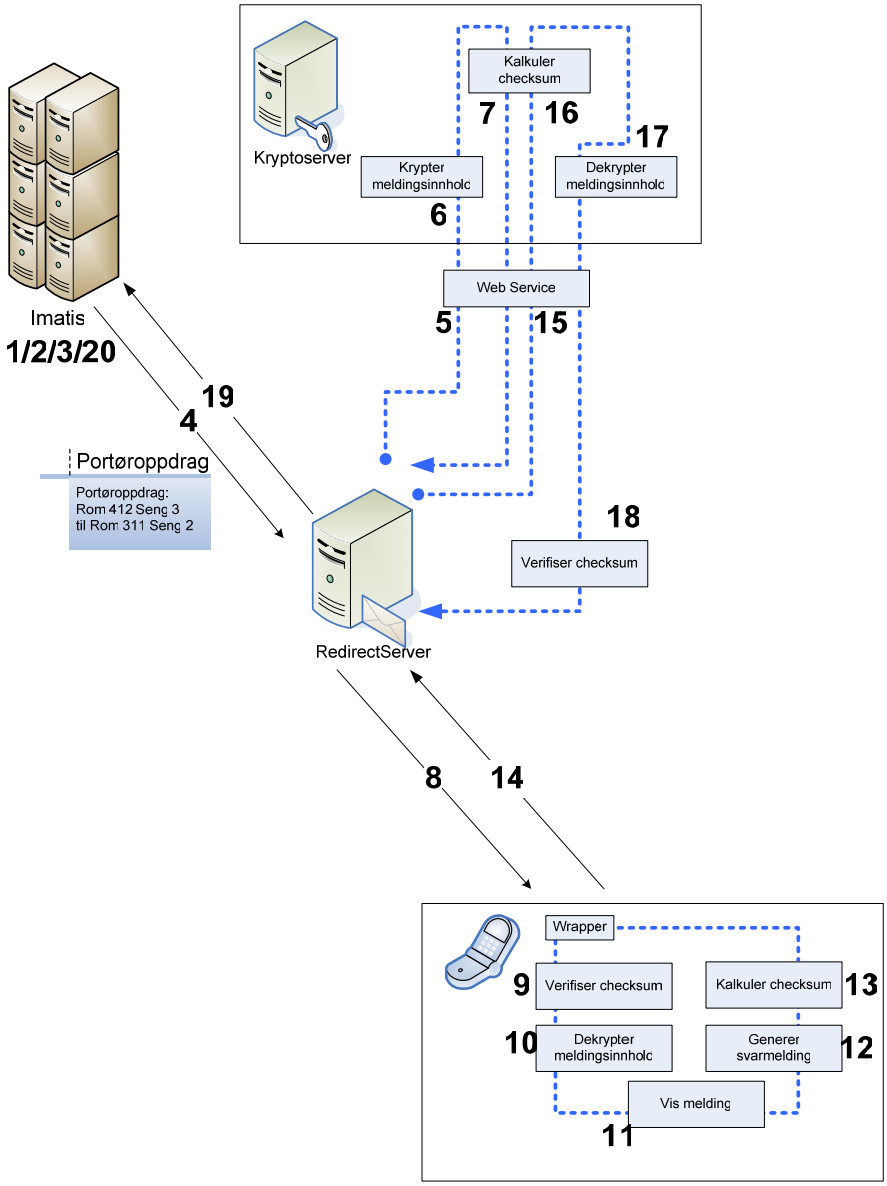
501 Unauthorized: Feilmelding som sendes til klient dersom login feilet.

502 Checksum Failed: Feilmelding som sendes både fra klient og RedirectServer dersom sjekksummen feiler.

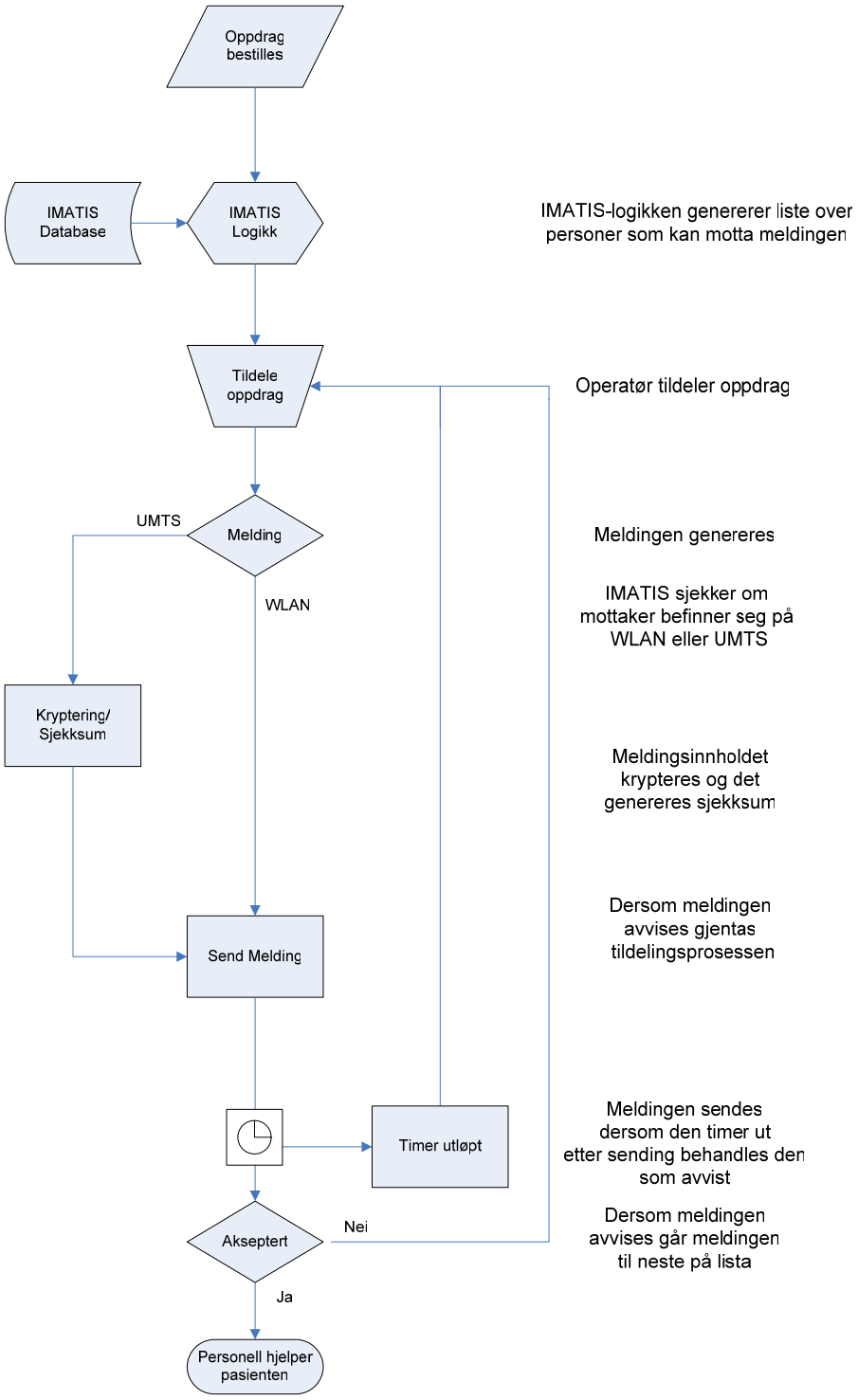
3.3.1 Meldingsgang for case Portørtjeneste

1. Oppdrag bestilles fra klient i nettet eller via webapplikasjon på kundesenteret.
2. **IMATIS** genererer liste over personer som kan motta meldingen utfra lokasjon
3. Oppdraget tildeles en portør av operatør på sengetunsapplikasjon.
4. Meldingen sendes ut til valgt portør dersom denne befinner seg på WLAN. Dersom portøren befinner seg utenfor WLAN sendes meldingen til **RedirectServer.**
5. Meldingsinnhold påføres nødvendig kryptering vha **KryptoServer**. Grensesnittet mot **RedirectServer** er en Web Service.
6. Meldingsinnholdet krypteres med AES.

7. Det genereres en sjekksum av meldingen. Sjekksummen genereres med det krypterte innholdet av meldingen
8. Meldingen med kryptert meldingsinnhold sendes til klienten. Deretter sendes en egen melding *CHKSUM* med sjekksummen.
9. Melding mottas hos klient og sjekksum verifiseres. Dersom sjekksum feiler sendes *502* tilbake til **RedirectServer.** Dette fører til at prosessen starter på nytt fra punkt 8. Dersom dette gjentas to ganger feiler prosessen og meldingen behandles som om den er avvist.
10. Meldingsinnholdet dekrypteres hos klienten.
11. Melding vises på skjerm og respons på oppdraget velges.
12. Svarmelding genereres utfra valg. Dette inkluderer kryptering av meldingsinnhold.
13. Sjekksum for svarmeldingen genereres.
14. Svarmeldingen sendes til **RedirectServer**. Denne meldingen etterfølges av en *CHKSUM*-melding med sjekksummen.
15. Svarmeldingen mottas i **RedirectServer**.og benytter Web Service mot **KryptoServer** for å generere sjekksum og dekryptering.
16. Det genereres en sjekksum av meldingen.
17. Meldingsinnholdet dekrypteres.
18. Meldingen sjekkes ved hjelp av sjekksum. Hvis den ikke stemmer, sender **RedirectServer** *502* til klienten som re-sender meldingen. Dersom denne prosessen feiler to ganger behandles meldingen som avvist.
19. Meldingen videresendes til **IMATIS** Meldingstjener.
20. Status på oppdraget endres på sengetunsapplikasjon. Dersom meldingen er blitt avvist må operatør på sengetunsapplikasjon tildele oppdraget på nytt.



Figur 26 - Forslag til systemoversikt for meldingsgang i case 'Portøroppdrag' [Egen figur]

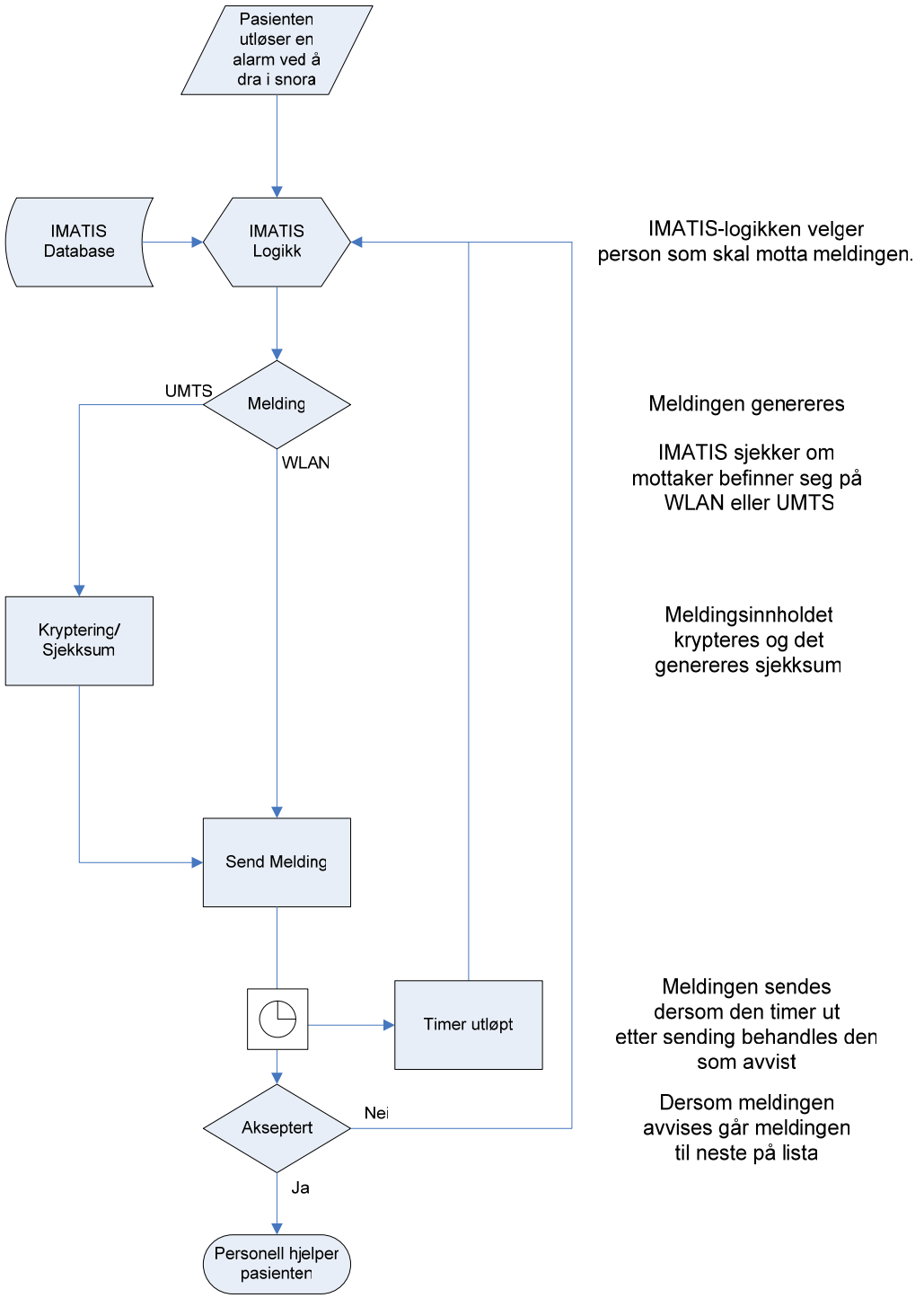


Figur 27 - Flytskjema for case 'Portøroppdrag' [Egen figur]

3.3.2 Meldingsgang for case NurseCall

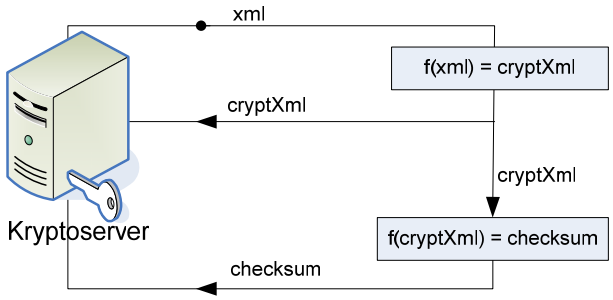
Systemoversikten for NurseCall vil være tilnærmet lik oversikten i Portøroppdrag (Figur 26).

1. Pasient drar i snora ved sin seng.
2. **IMATIS** genererer liste over personer som skal motta meldingen utfra hvem som er dedikert til den aktuelle personen eller rommet.
3. Meldingen sendes ut til helseressursen (klienten).
4. Dersom klienten befinner seg utenfor WLAN sendes meldingen til **RedirectServer**.
5. Meldingsinnhold påføres nødvendig kryptering vha **KryptoServer**. Samtidig genereres en sjekksum av meldingen. Sjekksummen genereres med det krypterte innholdet av meldingen. Når meldingsinnholdet er kryptert og sjekksum generert sendes XML-meldingen til klienten.
6. Deretter sendes en egen melding *CHKSUM* med sjekksummen.
7. Melding mottas hos klient og sjekksum sjekkes. Dersom den stemmer vises melding på skjerm. Dersom sjekksum feiler sendes 5*02* tilbake til **RedirectServer.** Dette fører til at prosessen starter på nytt fra punkt 6. Dersom dette gjentas to ganger feiler prosessen og meldingen behandles som om den er avvist.
8. Dersom sjekksummen stemmer sender klienten svarmelding utfra valg.
9. Svarmeldingen mottas i **RedirectServer**.og sender til **KryptoServer**.
10. Meldingen sjekkes ved sjekksum og dekrypteres dersom denne stemmer. Hvis den ikke stemmer, sender **RedirectServer** *502* til klienten som re-sender meldingen. Dersom denne prosessen feiler to ganger behandles meldingen som avvist.
11. Meldingen videresendes til IMATIS Meldinstjener.
12. Dersom meldingen er blitt avvist gjentas prosessen fra punkt 3.



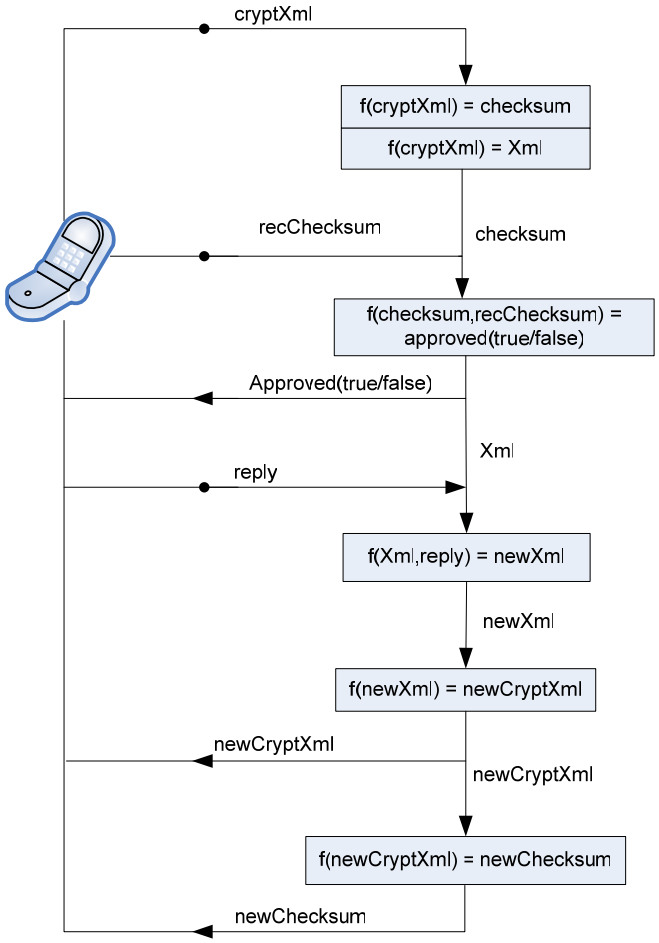
Figur 28 - Flytskjema for case 'NurseCall' [Egen figur]

3.3.3 Utvalgte prosesser i casene 'Portøroppdrag' og 'NurseCall'



Figur 29 - Klargjøring av meldingen som kommer fra IMATIS [Egen figur]

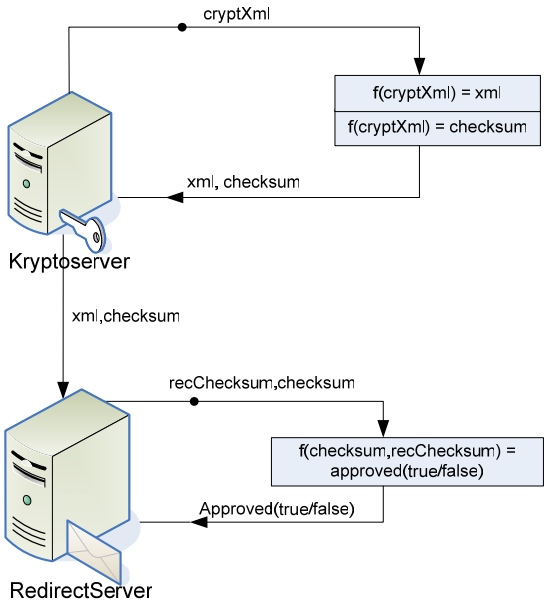
Figur 29: Meldingen mottas av RedirectServer fra IMATIS (*xml*). Deretter krypteres meldingen (*cryptXml*). Det genereres også en sjekksum av denne krypteringen (checksum).



Figur 30 - Behandling av innkommende melding på mobil enhet [Egen figur]

Figur 30: Den krypterte meldingen sendes av RedirectServer og mottas i den mobile enheten (*cryptXml*). Det genereres sjekksum (*checksum*) av denne og den krypterte

meldingen dekrypteres (*Xml*). Sjekksum mottas fra RedirectServer (*recChecksum*). Denne sammenlignes med generert sjekksum (*Approved(true/false)*). Brukeren svarer på meldingen (*reply*) og svarmelding genereres (*newXml*). Denne krypteres (*newCryptXml*) og danner utgagnspunkt for ny sjekksum (*newChecksum*). Både melding og sjekksum sendes til RedirectServer.



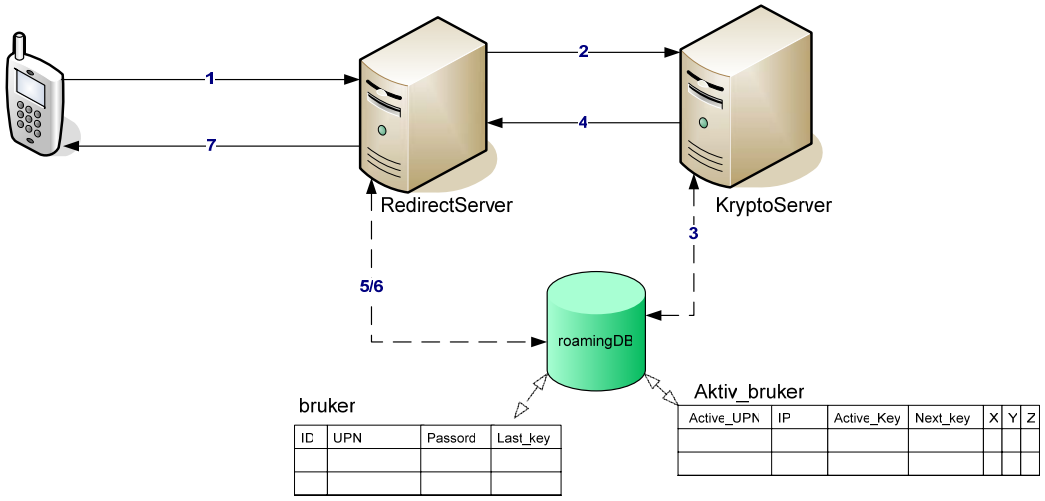
Figur 31 - Behandling av svarmelding fra mobil enhet [Egen figur]

Figur 31: Kryptert melding (*cryptXml*) mottas av RedirectServer og sendes til Kryptoserver. Meldingen dekrypteres (*xml*) og det genereres sjekksum (*chekcsum*) av den krypterte meldingen. Den dekrypterte meldingen og sjekksumen (*xml, checksum*) mottas i RedirectServer og den mottatte sjekksummen (*recChecksum*) verifiseres (*Approved(true/false)*). Meldingen sendes deretter tilbake til IMATIS for håndtering.

3.3.4 Meldingsutveksling i Roamingløsning

REGISTER Prosedyre

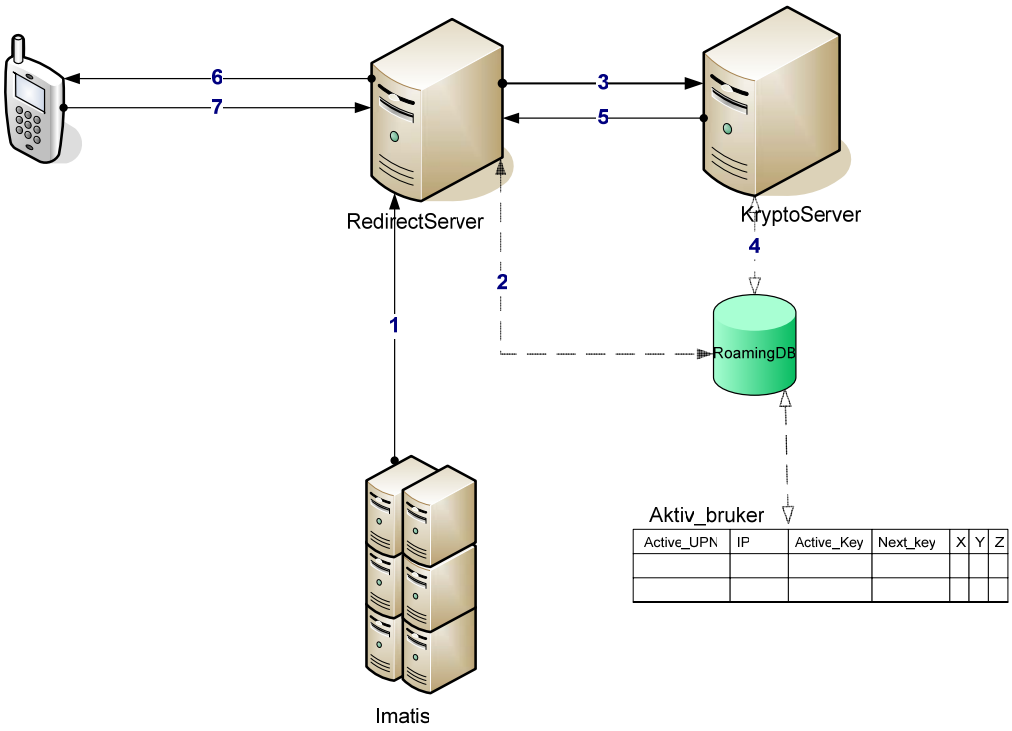
REGISTER er en prosedyre som kjøres for hver gang en enhet kobler systemet fra UMTS. Formålet med denne prosedyren er å oppdatere gjeldende IP-adresse for en enhet når enheten beveger seg ut av WLAN og over i UMTS.



Figur 32 – Oversikt over meldingsflyt for REGISTER-prosedyre [Egen figur]

1. Enhet sender REGISTER ”user1@domene.no” og ”kryptert passord”. Passordet er kryptert med den siste nøkkelen som er på enheten (s. 66 - Distribuering av nøkler til klienter).
2. RedirectServer sender UPN og kryptert passord til kryptoserver for dekryptering.
3. KryptoServer sjekker RoamingDB for å finne siste nøkkel som er brukt på den aktuelle IDen.
4. KryptoServer sender dekryptert passord tilbake til RedirectServer.
5. RedirectServer sammenligner det dekrypterte passordet mot passordet lagret i RoamingDB.
6. Dersom enheten er autentisert (godkjent bruker/pass), lagrer RedirectServer UPN, IP-adresse og gjeldende nøkkel i databasen.
7. RedirectServer sender en ACK tilbake til enheten som bekrefter at den er registrert og kan sende og motta data. Deretter legges enheten til som aktiv bruker. Her kan også lokasjonsinformasjon for brukeren lagres for videresending til IMATIS.

Meldinger fra IMATIS til mobil enhet i UMTS

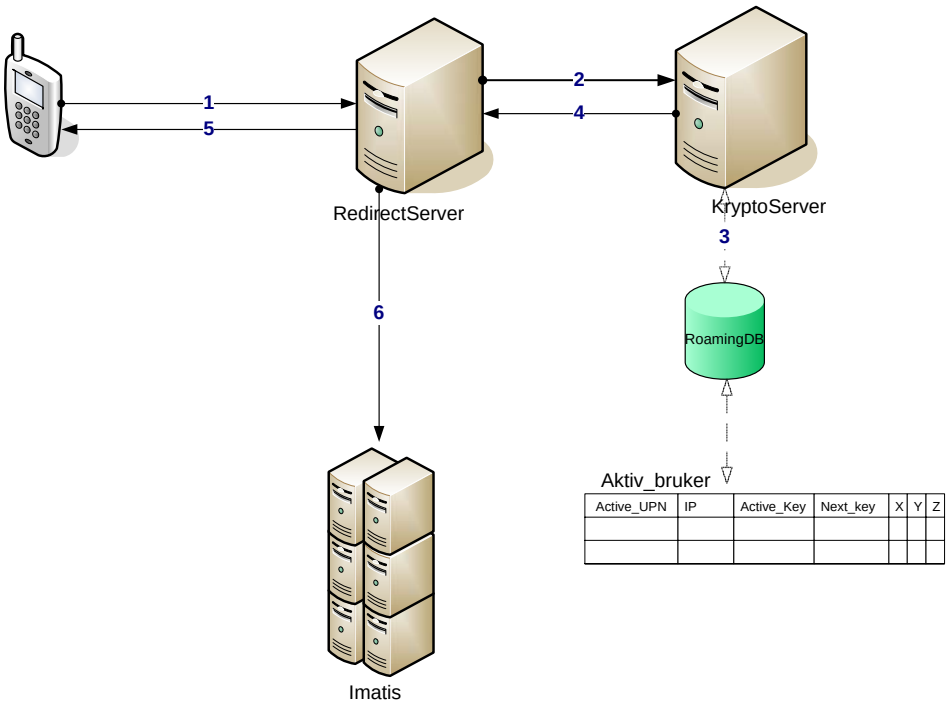


Figur 33 – Meldingsflyt for meldinger fra IMATIS til mobil enhet i UMTS [Egen figur]

1. IMATIS sender ut en adressert XML-melding til RedirectServer. Mottaker bestemmes av IMATIS-logikken.
2. RedirectServer finnner gjeldende IP for mottaker fra RoamingDB.
3. Meldingen sendes til KryptoServer for kryptering av meldingsinnhold og for generering av sjekksum.
4. Kryptoserver henter nøkkel knyttet til mottaker fra RoamingDB og krypterer meldingsinnholdet.
5. Meldingen sendes tilbake til RedirectServer sammen med en generert sjekksum for meldingen.
6. RedirectServer sender først selve meldingen og deretter medfølgende sjekksum for meldingen.
7. Ved mottatt melding med riktig sjekksum, sendes en ACK fra enheten

Grunnen til at telefonen ACKer på meldingen er for å bekrefte at enheten faktisk har fått den. Dersom enheten har byttet nettverk siden XML-meldingen ble sendt ut fra IMATIS, vil den ikke komme frem. For å få effektivisere meldingsflyten, sjekker RedirectServer om det er blitt oppdatert IP fra RoamingDB og sender meldingen på nytt etter 10 sekunder. Dersom det fortsatt ikke er respons, sendes dette tilbake til IMATIS for videre håndtering. Dette medfører at meldingen går til neste på lista.

Meldinger til IMATIS fra enheten



Figur 34 – Meldingsflyt for meldinger fra den mobile enheten til IMATIS i UMTS [Egen figur]

1. Enhet sender ut svarmelding og sjekksummelding utfra tidligere sendt XML. (Avvis/godta oppdrag)
2. RedirectServer mottar meldingen og tilhørende sjekksummelding fra enheten. Selve meldingen sendes videre til KryptoServer for dekryptering.
3. KryptoServer mottar meldingen og finner brukerens krypteringsnøkkel fra RoamingDB.
4. KryptoServer dekrypterer meldingen og genererer sjekksum og sender dette tilbake til RedirectServer.
5. RedirectServer sjekker sjekksummen og hvis den stemmer sendes en ACK til den mobile enheten.
6. Meldingen sendes videre til IMATIS for håndtering.

Distribuering av nøkler til klienter

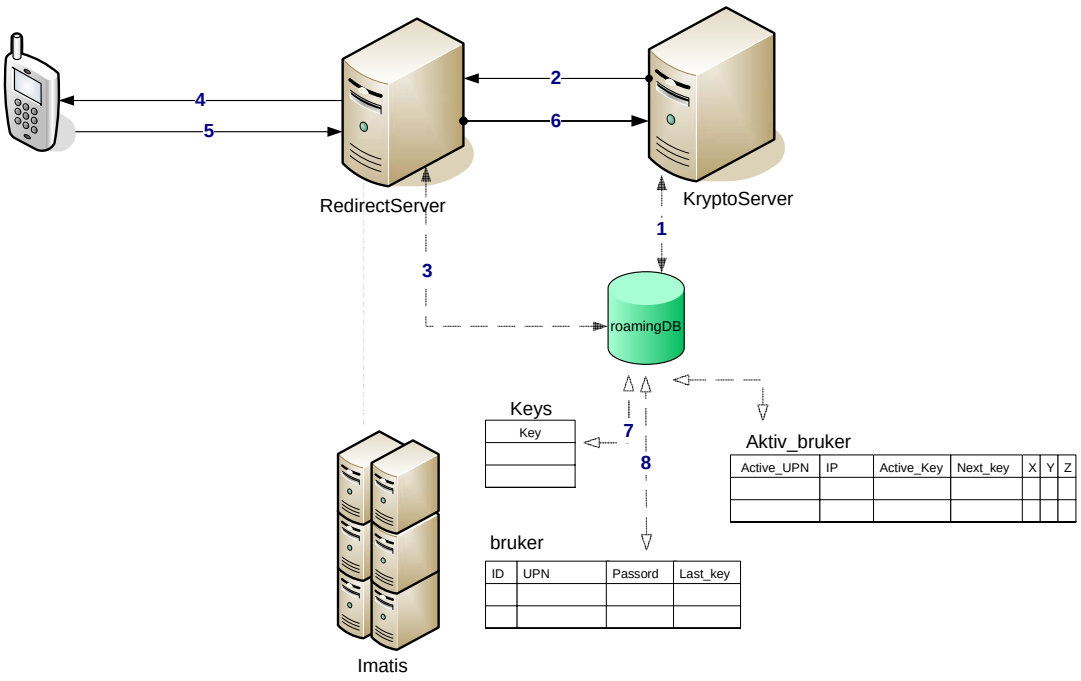
På enheter vil det ligge en settings-fil som inneholder variabler som *ServerIP* og *krypteringsnøkkel*. På serversiden vil det ligge en oversikt i databasen over hvilke nøkler som til enhver tid skal benyttes på påloggede enheter. Når en enhet logger på fra UMTS (sender REGISTER) hentes gjeldende nøkkel fra databasen og denne benyttes for kryptering. Ved videre kommunikasjon, er det tenkt at nøkkel oppdateres daglig for å forbedre sikkerhet. Det blir benyttet en 128/256-bits AES krypteringsalgoritme for meldingskryptering.

I utgangspunktet utsyres alle enheter med en standardnøkkel som knyttes til brukeren i databasen når nye enheter registreres i systemet. Denne standardnøkkelen legges i *Last_key* feltet i brukerdatabasen. I tillegg lagres standardnøkkelen som default i settings på den mobile enheten. Når en enhet logger seg på hentes *Last_key* fra bruker til *Aktive_brukere* og meldingsutvekslingen krypteres med denne. Når nøkkelen oppdateres vil den nye nøkkelen legge seg som Last_key. På enheten oppdateres settings med den nye krypteringsnøkkelen. Dette vil si at standardnøkkelen kun vil benyttes første gang en telefon logger seg på fra UMTS. Standardnøkkelen velges også å benyttes dersom uventede feil oppstår og kommunikasjonen må initieres på nytt.

Kryptering og dekryptering av meldinger foregår på server og klient. På serverside, foretas all kryptering, dekryptering og tilordning til nye nøkler av Kryptoserveren. Databasen inneholder alle tilgjengelige nøkler som kan benyttes. Den gjeldende nøkkel for en UMTS-tilkoblet klient lagres i *Aktiv_Bruker*. Dersom en bruker ikke er tilkoblet via UMTS, men befinner seg på WLAN vil brukeren kun ligge i Bruker. Der lagres også den siste nøkkelen som har blitt benyttet. Dersom en enhet skal kunne kommunisere over nettverket, må den ha en gyldig nøkkel og den skal ligge i Bruker, med tilhørende UPN. Det vil derfor være nødvendig å oppdatere denne databasen når det taes i bruk nye enheter.

Nøkkeloppdateringen initieres av kryptoserver. Når nøkkeloppdatering skal gjøres brukes *Aktive_Brukere* som utgangspunkt. Tidspunktet for dette vil kanskje kunne variere mellom plasser og rutiner. For å sikre at nøkkeloppdateringen skjer så ofte som mulig fastsettes to tidspunkt pr. døgn som utgangspunkt for rutinen. Dersom enheter er slått av, eller er ubrukt på en stund, vil det fortsatt være mulig for disse å få oppdatert sin nøkkel neste gang de er slått på under en distribueringsprosess. En mulighet er å legge tidspunktene for nøkkeloppdatering rett før eller rett etter vaktskifter. Det antas at de som kommer på vakt melder inn sin telefon med en gang og at denne etterpå kan regnes som aktiv.

Når nøkkelbytting starter går Kryptoserver gjennom *Aktive_Brukere* for å se hvilke enheter som er logget på. I *Aktive_Brukere* ligger det også en *Next_key* som indikerer den neste nøkkelen som skal benyttes for den gitte enheten. Alle nøkler velges tilfeldig fra *Keys* som inneholder en liste over tilgjengelige nøkler. Den nye nøkkelen krypteres med den gamle nøkkelen og sendes sammen med UPN til RedirectServer. RedirectServer sjekker så *Aktive_Brukere* for siste registrerte IP-adresse til den gitte UPN. Når enheten mottar den nye kryptonøkkelen sendes det en ACK kryptert med den nye nøkkelen til RedirectServer. Denne videresendes til krypto og dekrypteres, dersom dekrypteringen stemmer oppdateres *Aktive_brukere* ved å flytte *Next_key* til key og ved å hente en ny nøkkel fra *Key*.



Figur 35 - Meldingsflyt for distribuering av nøkler[Egen figur]

1. KryptoServer initierer oppdatering ved å gå igjennom Aktive_bruker og så sende ut enkeltstående nøkkeloppdateringer ved å hente Next_key.
2. KryptoServer krypterer den nye nøkkelen (Next_key) med den gamle (Active_Key) og sender den til RedirectServer.
3. RedirectServer henter IP til UPN fra RoamingDB.
4. Sender melding med den nye nøkkelen til enheten.
5. Enheten oppdaterer sin nøkkel og svarer med ACK kryptert med den nye nøkkelen.
6. RedirectServer sender ACK-meldingen til KryptoServer for dekryptering. Dersom meldingen stemmer oppdateres nøkkelen fra Next_key til Active_Key.
7. Ny nøkkel hentes fra feltet Key i nøkkeltabellen (Keys) og plassereres i Next_key.
8. Siste godkjente nøkkel (Last_key) oppdateres i Aktiv_bruker.

4 PROOF OF CONCEPT

Dette kapittelet gjennomgår Proof of Concept-testingen som er blitt gjennomført. Dette kapittelet utgjør sammen med deler av forrige kapittel resultatene av oppgaven.

4.1 Testmiljø

PoC-testene har vært utført i WLAN på CARDIAC og med mobiltelefon med Telenor GSM (GPRS). Mobiltelefonen som er brukt er en Sony Ericsson P910 [90]. Initielle tester er gjort lokalt på laptop med Wireless Toolkit (WTK). WTK er en mobiltelefonemulator for PC laget for emulering og testing av J2ME-applikasjoner. Alle skjermbilder er fra WTK-emulator, men alle PoC-applikasjoner er også testet på Sony Ericsson P910. Siste versjon, noe som ligner en demoapplikasjon, er også testet i 3G-nettet med en Nokia N70[91].

Figur 36 - Sony Ericsson P910i [90]

Figur 37 - Nokia N70 [91]

4.2 Vekking av applikasjon med SMS

4.2.1 Formål

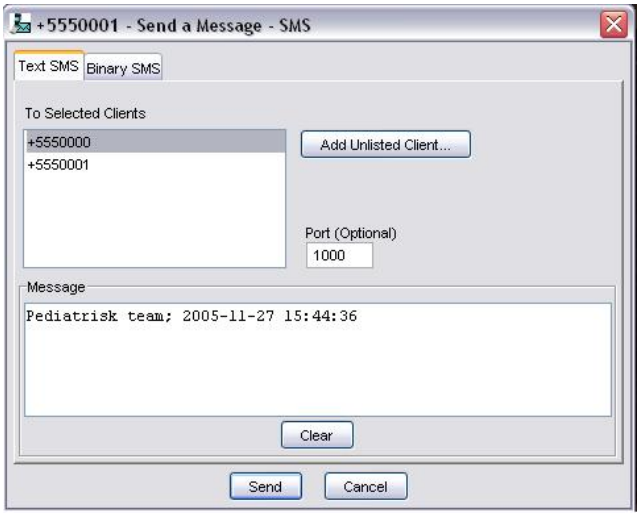
Hensikten med denne testen er å vise evnen til å vekke opp applikasjoner basert på innkommende meldinger. I MIDP 2.0 benyttes PushRegistry til dette. Grunnen til dette er at den mobile enheten benyttes til andre ting samtidig som mottakerapplikasjonen kjøres. Dette betyr at appliaksjonen må kjøres i bakgrunnen og at man samtidig kan lytte og reagere på innkommende nettverkshendelser. Ønsket er å kunne benytte dette til å generere svarmeldinger enten det er SMS eller en GPRS-melding. Meningen er deretter å benytte meldingen som kommer inn til å sende tilbake respons basert på valg i menyen.

4.2.2 Resultat

Applikasjonen er testet lokalt på maskin med WTK. WTK muliggjør sending av SMS lokalt til et spesielt portnummer. Dette er nødvendig for å få i gang PushRegistry. Applikasjonen er i tillegg testet i mobilnettet ved hjelp av en applikasjon som sender SMS til seg selv og dermed starter SMS-tolking. Grunnen til dette er at for å få til oppstart av vår applikasjon må meldingen sendes på en port som er annerledes enn standard SMS-port. Dersom ikke dette skjer vil meldingen legges i vanlig SMS-innboks. SMS-innboksen er utilgjengelig via J2ME.

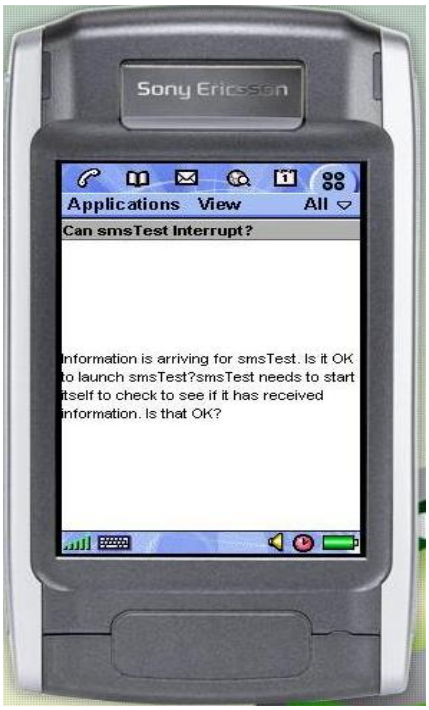
Applikasjonen baserer seg på bruk av MIDP 2.0 og WMA. PushRegistry-metoden benyttes for å lytte etter SMS på en gitt port. Applikasjonen startes hver gang en SMS kommer inn på den gitte lytterporten. Det genereres deretter en SMS som sendes tilbake til avsender basert på valgene ”Godta” og ”Avvis”. Etter dette lukker applikasjonen og går tilbake i ventemodus.

PushRegistry er en metode som benytter seg av, som navnet antyder, en Push-mekanisme. Det vil si at den håndterer informasjon asynkront og reagerer når en forhåndsdefinert ting skjer dette kan for eksempel være en innkommende SMS. Det å håndtere informasjon asynkront vil si å gjøre dette uten å være avhengig av synkronisering med motparten. PushRegistry kan starte applikasjoner på to måter. Enten ved at den lytter på en spesifikk protokoll og en gitt port eller ved hjelp av en timer.



Figur 38 - Sending av SMS i WTK-miljøet [Egen figur]

Figur 38: Her sendes en SMS til nr. +55500000 (emulert mobilklient) på port 1000. Meldingsinnholdet som ligger i Message-feltet er hentet fra en IMATIS alarm-melding.



Figur 39 - Advarsel som dukker opp ved innkommende nettverkshendelser dersom MIDleten ikke er signert [Egen figur]

Figur 40 - Meldingen vises i telefonen. [Egen figur]

Figurene over viser PushRegistry som ”vekker opp” applikasjonen etter å ha mottatt SMS på port 1000. Figur 39 viser en advarsel som må håndteres ved innkommende meldinger. Dersom ikke applikasjonen eller MIDleten er signert vil brukeren få opp denne advarselen. Disse advarslene dukker opp når bruker vil aksessere nettverksressurser og MIDleten ikke er signert.. Slike advarsler/bekreftelser vil også dukke opp når MIDleten startes for å bekrefte at bruker gir applikasjonen tilgang til å sende/motta SMS. Prosessen for å signere MIDlets er beskrevet i Vedlegg C. Figur 40 viser SMS som er sendt via WTK og vist på skjerm av applikasjonen.

Figur 41 - Visning av svarmeny på mobil [Egen figur]

Valgt svar genererer svarmelding på SMS og sender den tilbake til server. På vår testtelefon P910 fungerte ikke mottagelse av melding nummer to. Dette var imidlertid uinteressant da det var PushRegistrys evne til å vekke opp applikasjoner som skulle testes. I forhold til dette viste det seg at SMS og PushRegistry er støttet og fungerer som det skal. Problemer oppstod når det samme skulle testes med en melding over GPRS. Under testingen av PushRegistry med Socket og Datagram nektet telefonen å innstallere applikasjoner som benyttet seg av denne funksjonaliteten. Det viser seg at ingen telefoner i dag støtter dette av sikkerhetsårsaker.

Installasjon av MIDlets med PushRegistry-alarmer i form av Socket ble nektet installert på Sony Eriscsson P910. Det viser seg at denne delen av PushRegistry ikke støttes av Sony Ericsson [92]. Vi har i tillegg testet dette på en QTEK 9100 PDA med samme resultat [93].

4.3 Adressering i mobilnettet

4.3.1 Formål

Hensikten med denne testen er å etablere kommunikasjon mellom klient på mobil i GPRS-nettet og en enkel serverapplikasjon på laptop i WLAN. Dette skal gjennomføres helt enkelt i form av en enkel utveksling av REGISTER <name> og tilhørende ACK fra server. Problemet som skal belyses er at offentlige GPRS-nett bruker NAT. NAT blir gjort av både Telenor og Netcom når man kobler seg opp mot sitt abbonements APN. Med andre ord, man må finne en vei rundt dette slik at man kan nå telefonen utenfra.

4.3.2 Resultat

Systemet består av en serverapplikasjon i WLAN på laptop som kommuniserer med klientapplikasjon på mobiltelefon i GPRS. Mobiltelefonen kobler seg opp mot GPRS og sender en REGISTER-melding til server. Serveren mottar denne og sender ACK tilbake.

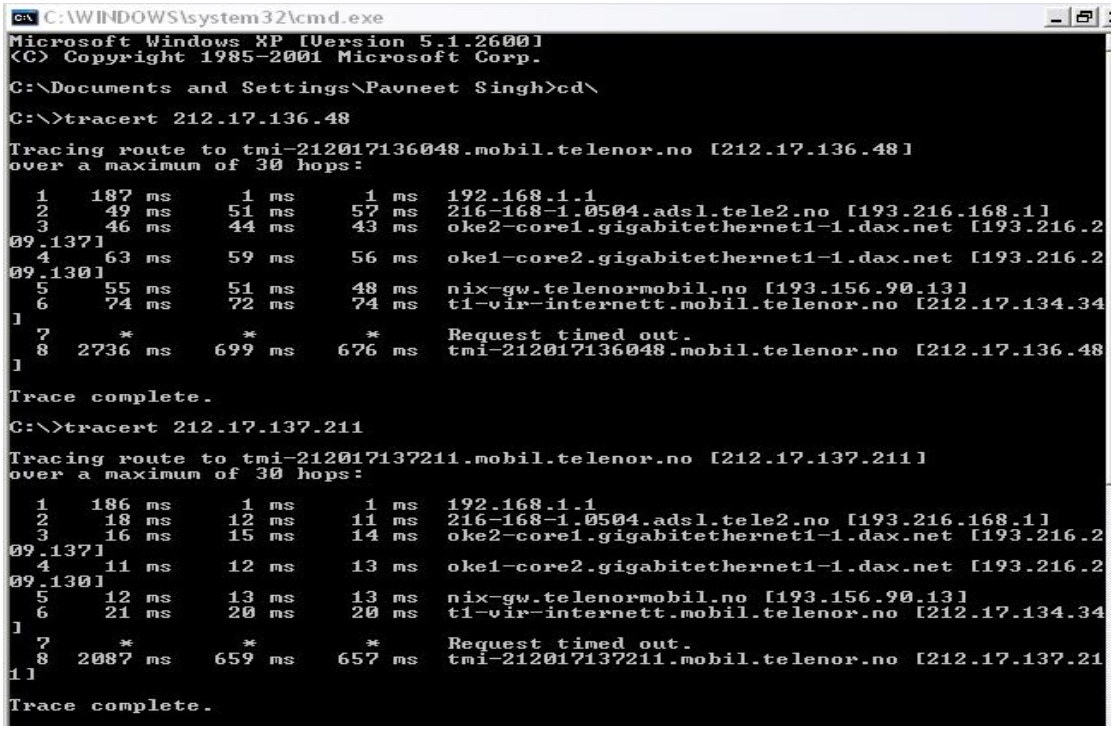
```
C:\>tracert 212.17.141.53

Tracing route to gprs-ggsn5-nat.mobil.telenor.no [212.17.141.53]
over a maximum of 30 hops:

  1   157 ms     1 ms     1 ms  192.168.1.1
  2    43 ms    44 ms    45 ms  216-168-1.0504.adsl.tele2.no [193.216.168.1]
  3    30 ms    27 ms    25 ms  oke2-core1.gigabitethernet1-1.dax.net [193.216.2
09.137]
  4    41 ms    39 ms    34 ms  oke1-core2.gigabitethernet1-1.dax.net [193.216.2
09.130]
  5   295 ms   277 ms   272 ms  nix-gw.telenormobil.no [193.156.90.13]
  6    73 ms    78 ms    72 ms  t1-vir-internett.mobil.telenor.no [212.17.134.34
]
  7    34 ms    35 ms    34 ms  gw-internett.mobil.telenor.no [212.17.134.33]
  8    51 ms    49 ms    49 ms  t1-vir-internett.mobil.telenor.no [212.17.134.34
]
  9    56 ms    54 ms     *     gw-internett.mobil.telenor.no [212.17.134.33]
 10    59 ms    43 ms    43 ms  t1-vir-internett.mobil.telenor.no [212.17.134.34
]
 11    28 ms    28 ms    30 ms  gw-internett.mobil.telenor.no [212.17.134.33]
 12    24 ms    23 ms    26 ms  t1-vir-internett.mobil.telenor.no [212.17.134.34
]
 13   314 ms   109 ms    47 ms  gw-internett.mobil.telenor.no [212.17.134.33]
 14    29 ms    28 ms    26 ms  t1-vir-internett.mobil.telenor.no [212.17.134.34
]
 15    27 ms    41 ms    31 ms  gw-internett.mobil.telenor.no [212.17.134.33]
 16    27 ms    26 ms    21 ms  t1-vir-internett.mobil.telenor.no [212.17.134.34
]
 17    16 ms    16 ms    15 ms  gw-internett.mobil.telenor.no [212.17.134.33]
 18    16 ms    18 ms    18 ms  t1-vir-internett.mobil.telenor.no [212.17.134.34
]
 19    21 ms    20 ms    18 ms  gw-internett.mobil.telenor.no [212.17.134.33]
 20    29 ms   201 ms   176 ms  t1-vir-internett.mobil.telenor.no [212.17.134.34
]
 21    17 ms    16 ms    18 ms  gw-internett.mobil.telenor.no [212.17.134.33]
 22    18 ms    38 ms    20 ms  t1-vir-internett.mobil.telenor.no [212.17.134.34
]
 23    18 ms    17 ms    17 ms  gw-internett.mobil.telenor.no [212.17.134.33]
 24    19 ms    19 ms    20 ms  t1-vir-internett.mobil.telenor.no [212.17.134.34
]
 25    26 ms    25 ms    24 ms  gw-internett.mobil.telenor.no [212.17.134.33]
 26    19 ms    18 ms     *     t1-vir-internett.mobil.telenor.no [212.17.134.34
]
 27   202 ms    19 ms   164 ms  gw-internett.mobil.telenor.no [212.17.134.33]
 28    19 ms    23 ms    19 ms  t1-vir-internett.mobil.telenor.no [212.17.134.34
]
 29    19 ms    18 ms    18 ms  gw-internett.mobil.telenor.no [212.17.134.33]
 30    20 ms    20 ms    18 ms  t1-vir-internett.mobil.telenor.no [212.17.134.34
]

Trace complete.
```

Figur 42 - Sporing av en mobiltelefon som kobler til GPRS gjennom public APN [Egen figur]

Figur 42: En sporing (trace) av en mobiltelefon koblet til en gateway (APN) som NATer på innsiden. Mobiltelefonen er koblet til med adresse *gprs-ggsn5-nat.mobil.telenor.no* med IP 212.17.141.53. Tracen av denne IPen ender opp i gatewayen med adresse *gw-internett.mobil.telenor.no* med IP 212.17.134.34. Dette medfører at REGISTER-meldingen kommer frem til server, men ACK-meldingen aldri kommer frem fordi meldingen sendes til en privat IP-adresse på innsiden av gatewayen.

Målet med applikasjonen er å nå en mobil i GPRS-nettet utenfra. Det vil si prøve ut muligheter for å komme rundt tidligere nevnte adresseringsproblem (kap 2.5.2). Gatewayen som står i mobilnettet vil i utgangspunktet operere som en ruter gjør på et vanlig WLAN. Dette medfører at enhetene som står på innsiden får IP-adresse ved hjelp av NAT slik at de ikke er tilgjengelige utenfor ruteren.

Vanligvis settes telefonene opp med disse public gatewayene:”internet” hos Telenor og ”internet.netcom.no” for Netcom. Disse operatørene har også satt opp dedikerte gatewayer for vpn-trafikk som muligjør en public IP for enhetene i gprs-nettet. Disse gatewayene finnes på ”internet.public” for Telenor og ”vpn.netcom.no” for Netcom. Ved å sette APN til ”internet.public” på telefonen oppnås det at mobiltelefonen får tildelt en public IP-adresse og dermed er tilgjengelig utenfra.



Figur 43 - Trace på en mobiltelefon som kobler til GPRS gjennom vpn-APN [Egen figur]

Figur 43: En trace av en mobil enhet koblet til public IP-APN. I utgangspunktet er IP-adressen til enheten her 212.17.136.48 og denne adressen er unik. Tracen går helt tilbake til den mobile enheten gjennom gatewayen i GPRS-nettet. Etter dette er det gjort en ny oppkobling og enheten tildeles dermed også en ny IP-adresse 212.17.137.211. Denne traces også helt tilbake til den opprinnelige mobiltelefonen. Dette gjør at når REGISTER meldingen kommer frem til server og ACK-meldingen sendes vil den nå mobiltelefonen. Dette ble også gjennomført og oppnådd med testing med en Ericsson P910i og tilkobling til ”internet.public” i Telenors mobilnett.

4.4 Forsinkelse i UMTS under meldingsutveksling

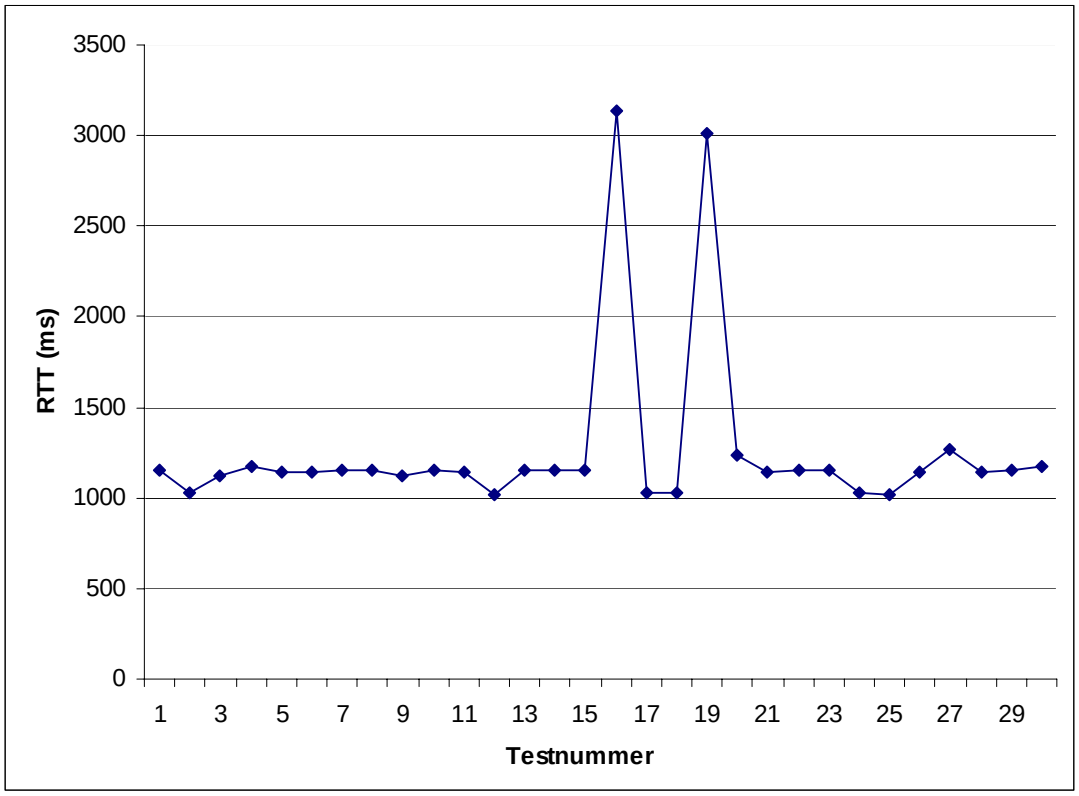
4.4.1 Formål

Formålet med disse målingene var å måle forsinkelse ved bruk av GPRS og 3G-nettet til sending av enkle datameldinger. Det som ble målt var tiden det tok fra den mobile enheten sendte sin REGISTER-melding til den mottok svarmeldingen fra serveren (ACK); RTT[10].

[10] Round Trip Time

4.4.2 Resultat GPRS

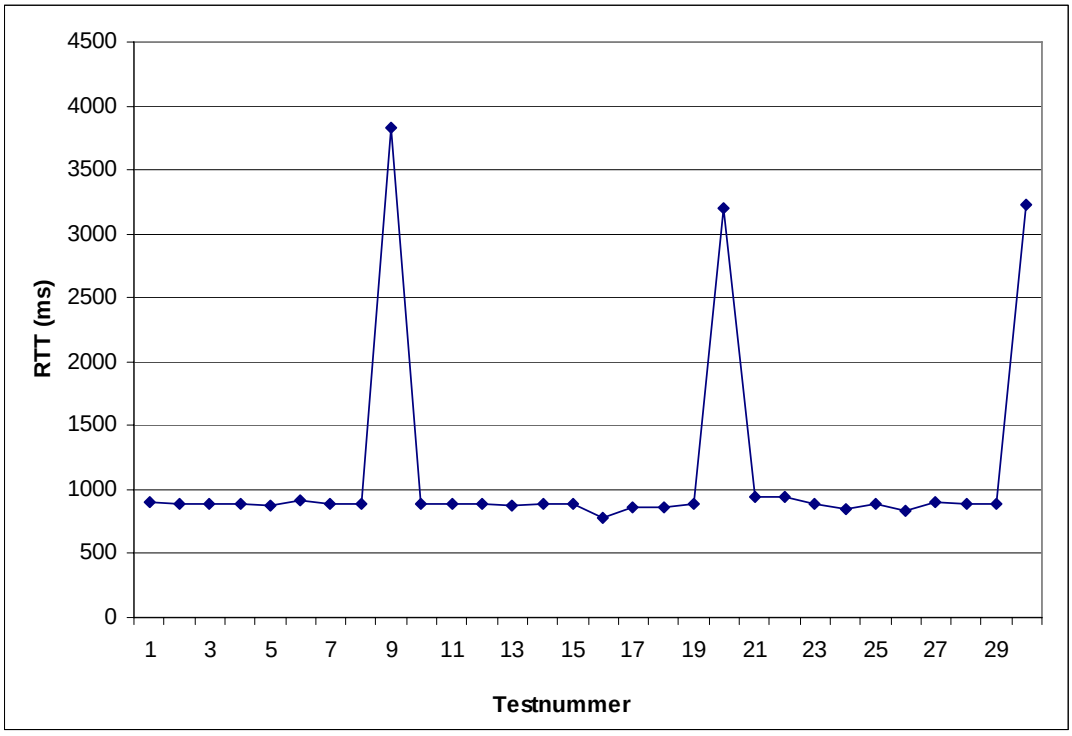
Vi har gjort to forsøk med måling av RTT. Den første målingen ble gjort i GPRS-nettet med en Sony Ericsson P910. Resultatet av målingen kan sees i Figur 44. Det ble gjort 30 stikkprøver og gjennomsnittlig forsinkelse i GPRS-nettet ble målt til 1260 millisekunder. Minimum og maksimum forsinkelse ble målt til henholdsvis 1016 ms og 3140 ms. Siden svært få resultater beveger seg bort fra antatt middelverdi som ligger på ca. 1130 ms er det rimelig å anta at de to avvikende verdiene kommer som et resultat av retransmisjoner. Dersom vi ser bort fra de to avvikende verdiene vil gjennomsittet i målingene (28 stikkprøver) ligge på 1132 ms. Vi kan også bruke disse resultatene som veiledende når det gjelder å si noe om hvor ofte retransmisjon inntreffer. I vårt forsøk inntraff retransmisjon ved 2/30 av tilfellene noe som er 6.6 %.



Figur 44 - RTT ved meldingsutveksling i GPRS

4.4.3 Resultat UMTS

Den andre målingen ble gjort i UMTS og resultatet kan sees i Figur 45. Det ble tatt 30 stikkprøver og gjennomsnittlig forsinkelse ble målt til 1138 ms. Minimum og maksimum forsinkelse ble målt til henholdsvis 782 ms og 3828 ms. Også her viste det seg at noen få verdier lå mye høyere enn det reelle gjennomsnittet for en RTT. Ved å se bort fra de tre verdiene som kommer som et resultat av retransmisjon vil gjennomsnittet for disse 27 stikkprøvene ligger på 885 ms. I dette forsøket ble det gjennomført retransmisjon ved 3/30 stikkprøver noe som betyr 10 % av tilfellene.



Figur 45 - RTT ved meldingsutveksling i UMTS

4.5 Tolking av XML-meldinger

4.5.1 Formål

Hensikten med denne testen vil være å utvide applikasjon slik at den også omfatter det å tolke IMATIS-meldingsformatet og vise riktig resultat på skjerm over GPRS som tidligere blir gjort for IP-telefoner i WLAN. I dette ligger også generering av svarmeldinger på telefonen utfra gitt svaralternativer (jf. IP-telefon og godta/avvis). I tillegg skal disse meldingene også sendes tilbake til server med svarkode.

4.5.2 Resultat

Testene ble gjort ved hjelp av utvidelser av applikasjonene omtalt i kap 4.3. Systemet består av en serverapplikasjon på laptop i WLAN som sender en XML-melding til mobiltelefonen. I tillegg utvides klientapplikasjonen på mobil enhet som mottar og viser valgte elementer av XML-meldingen på skjerm. For å håndtere XML-formatet benyttes kXML som muliggjør parsing av XML-dokumenter. Beskrivelsen av kXML under er hentet fra [94].

"The kXML project provides an XML pull parser and writer suitable for all Java platforms including the Java 2 Micro Edition (CLDC/MIDP/CDC). Because of its small footprint size, it is especially suited for Applets or Java appications running on mobile devices like Palm Pilots or MIDP enabled cell phones."

Meldingen sendes fra server til en mobil enhet koblet til GPRS-nettet. Deretter parses XML-meldingen slik at selve meldingsinnholdet kan håndteres i enheten.

Figur 46 - Skermbilde av XML-melding fra server på P910i [Egen figur]

Skjermbilde av XML-melding fra server på mobil enhet. Meldingen tolkes og alle elementene håndteres i mobilen. Meldingsheaderen vises som tittel og meldingsinnholdet vises i display. De andre elementene i den opprinnelige meldingen tas vare på og benyttes når svarmelding genereres utfra brukerens valg.

4.6 Sikkerhetsmekanismer

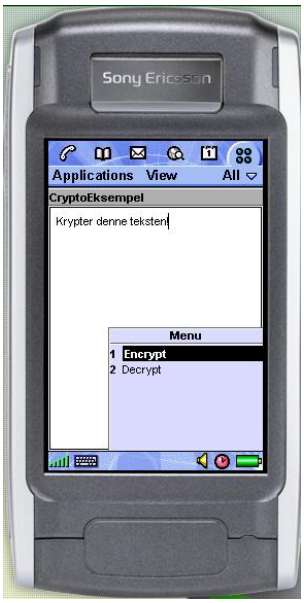
4.6.1 Formål

Formålet med denne testen er å benytte seg av valgte krypteringsalgortimer for å oppnå konfidensialitet for meldingsinnhold og generere sjekksum for å sikre dataintegriteten. Vi har valgt å benytte AES for kryptering av meldinger. For å sikre dataintegritet velger vi å benytte SHA-1 for generering av sjekksummer

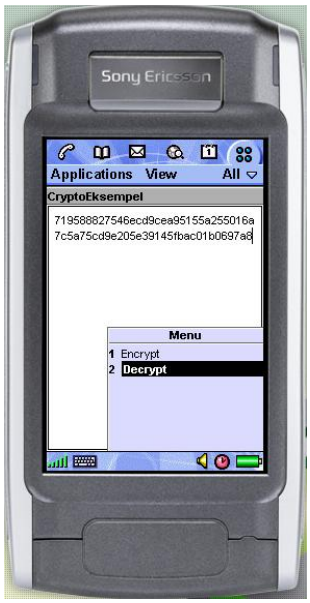
4.6.2 Resultater

For å få til tilfredstillende symmetrisk kryptering er AES blitt valgt som krypteringsalgoritme. AES er implementert i et tredjeparts klassebibliotek; BouncyCastle. BouncyCastle er blitt benyttet for kryptering og beskrivelsen under er hentet fra [95].

"The Bouncy Castle Crypto package is a Java implementation of cryptographic algorithms....The package is organised so that it contains a light-weight API suitable for use in any environment (including the newly released J2ME) with the additional infrastructure to conform the algorithms to the JCE framework."



Figur 47 - Implementering av 256-bits AES-kryptering 1 [Egen figur]

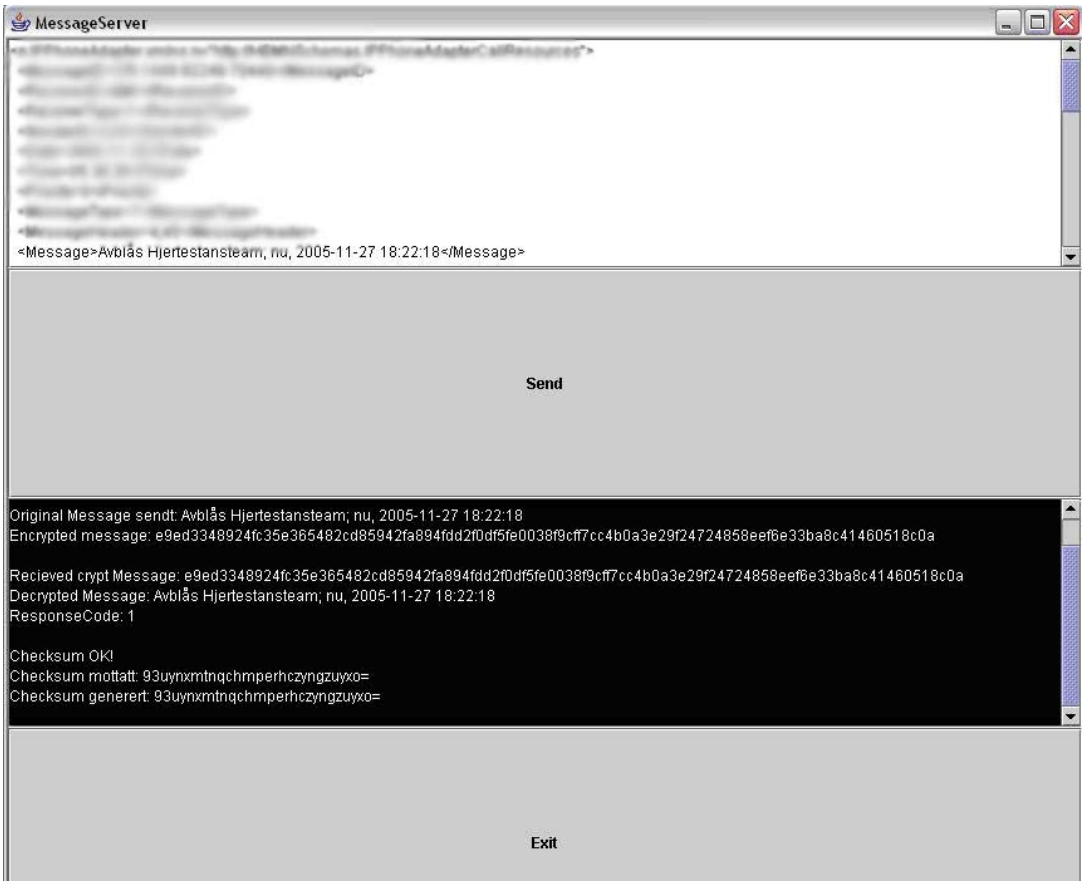


Figur 48 - Implementering av 256-bits AES-kryptering 2[Egen figur]

Figur 49 - Implementering av 256-bits AES-kryptering 3 [Egen figur]

Figur 47, Figur 48, Figur 49: Implementering av 256-bits AES-kryptering i mobil enhet. Figur 47 viser tekststrengen som skal krypteres i display (klartekst). Figur 48 viser resultatet av krypteringen (siffertekst) og figur Figur 49 viser dekryptering av gjeldende siffertekst.

Målet med denne enkle konseptuelle testen er å først kryptere og dektryptere en enkel tekststreng. Krypteringen gjøres ved hjelp av en forhåndsdefinert nøkkel på 32 tegn (256 bit). Testen viser at kryptering og dekryptering av en gitt tekststreng med samme nøkkel gir samme resultat som utgangspunktet.



Figur 50 – Sending og mottagelse av kryptert melding fra server-aplikasjon [Egen figur]

Figur 50: Bilde av sending av kryptert melding over lufta. Samt sending av tilhørende sjekksum. Bildet viser et skjermbilde av serverapplikasjonen som tar imot en svarmelding fra klienten. Den øverste delen inneholder meldingen som skal sendes til klienten samt en send-knapp. Den nederste delen inneholder et vindu som viser trafikken som går på serveren og informasjon om denne.

Meldingssekvensen over initieres av REGISTER-meldingen fra tidligere PoC-test (kap 4.3). Deretter sendes en alarm-melding som tolkes og svares på hos klienten. Selve innholdet, det vil si selve informasjonen i meldingen, krypteres med 256-AES. På klienten lagres denne nøkkelen i en XML-fil sammen med brukernavn og passord. På serversiden er nøkkelen skrevet rett inn i koden. I vinduet nederst på Figur 50 ser vi trafikken som går på serveren. Vinduet er ment som et debug-vindu og for å få ut enkel informasjon om trafikken på PoC-applikasjonen. Vi har valgt å implementere sjekksum som en egen melding for å ta vare på dataintegriteten. Meldingen starter med CHKSUM og etterfølges av generert sjekksum fra tidligere sendt XML-melding. Sjekksumen er generert med BouncyCastle og bruker SHA-1 som algoritme. Kryptering av meldingsinnhold og generering av sjekksum er testet i 3G og i GPRS-miljø med samme resultat. Under følger en beskrivelse av de forskjellige informasjonselementene i debug-vinduet.

"Original Message sent": Vises når serveren har sendt en XML-melding til klienten. Det etterfølgende er selve meldingsinnholdet i klartekst.

"Encrypted Message": Viser meldingsinnholdet kryptert med AES 256. Dette er det faktiske meldingsinnholdet som sendes til klienten.

"Recieved Crypt Message": Viser meldingsinnholdet som faktisk mottas fra klienten. Meldingsinnholdet er kryptert med AES 256 og benytter den samme nøkkelen som ligger på serveren.

"Decrypted Message": Viser det dekrypterte meldingsinnholdet som er mottatt fra klienten.

"Response Code": Response Code er et eget element i XML-meldingen. Det er denne som faktisk bestemmer hvorvidt brukeren har godtatt eller avvist meldingen. Her ser vi at Response Code er satt til 1. I vår PoC-applikasjon betyr dette at brukeren har godtatt oppdraget.

"Checksum OK!": Indikerer at mottatt sjekksum og generert sjekksum for meldingen stemmer. Disse asekksummene generert med SHA-1Man kan derfor gå ut fra at meldingen ikke er blitt forandret siden den ble sendt fra klienten. De to neste feltene viser de to sjekksummene. Sjekksum-prosessen fungerer likt ute på klientene.

5 DRØFTING

I dette kapittelet drøftes resultatene fra designkapittelet og PoC-kapittelet. Videre vil bruksområder og fremtidige muligheter bli gjennomgått.

5.1 Introduksjon

I begynnelsen av denne rapporten, introduserte vi CARDIACs IMATIS-løsning som benyttes på St.Olavs Hospital i Trondheim. Denne kommunikasjonsløsningen baserer seg på meldingsutveksling og gir helsepersonell mulighet til å bevege seg samtidig som de kan motta meldinger ved hjelp av en trådløs IP-telefon. Denne IP-telefonen henger på sykehusets WLAN og mottar meldinger når den er koblet til dette. Svakheten ved denne løsningen er at St. Olavs Hospital inneholder gammel bygningsmasse som ikke dekkes av WLANet. Dette medfører at helsepersonellet må ha med seg en mobiltelefon i tillegg til IP-telefonen. Problemet med dette er både at man må holde styr på to enheter til enhver tid samt at manuelt svar på SMS ikke er spesielt brukervennlig. Essensen i vår oppgave har vært å erstatte disse to enhetene med en enkelt enhet som gir samme tilgjengelighet og sikkerhet i gammel (GPRS/UMTS) og ny (WLAN) bygningsmasse. For å kunne gjøre dette må enheten støtte begge nettverksteknologier og også ha muligheten til å roame sømløst mellom disse.

For å løse oppgaven har vi tatt utgangspunkt i eksisterende roamingteknologier. Videre har vi gjort valg basert på disse forskjellige teknologiene og foreslått et teoretisk design for en roamingløsning. Utvalgte deler av dette designet er blitt testet ved Proof of Concept-testing. Kapittel 2.9 *Våre Valg* forteller om hvilke teknologier som er valgt i løsningen og hvordan de skal benyttes er presentert i kapittel 3 (design) og kapittel 4 (PoC). Disse valgene er gjort med tanke på tre faktorer:

- Hvilken roamingteknologi skal benyttes?
- Hvordan og i hvilken grad skal sikkerhet implementeres?
- Hva slags mobil enhet skal/kan benyttes i den videre utviklingen av oppgaveløsningen?

Den videre drøftingen av oppgaven baseres på disse spørsmålene, og hvordan de valgene vi har gjort har påvirket vår løsning.

5.2 Roaming

Vi valgte å legge vår løsning på applikasjonslaget i OSI-modellen og det finnes flere grunner til dette. For det første vil man kunne operere uavhengig av underliggende linklag. Dette medfører at så lenge IP benyttes for å overføre data vil løsningen fungere, uavhengig av underliggende kommunikasjonsteknologi. I tillegg til å operere på applikasjonslaget, har vi basert vår foreløpige kommunikasjonsprotokoll på SIP. Dette gir fordeler i forhold til fremtidig bruk av systemet. Vi har også forsøkt å basere vår valgte arkitektur på SIP-arkitekturen fordi den er bygget opp med grunnlag i en klient-tjener arkitektur og inneholder enkelte enheter som vil være nødvendig i vår løsning. Ulempen med en slik løsning vil være at applikasjonslaget medfører tregere handover

og overhead enn en tilsvarende løsning på nettverkslaget eller linklaget. I denne problemstillingen anser vi at dette likevel ikke vil være et så stort problem siden kravene til rask handover vil være små. Dette fordi det i IMATIS benyttes små xml-meldinger for kommunikasjon. Testene vi har gjort med tilhørende måling av RTT viser at forsinkelsen i mobilnettet vil være liten. Disse små xml-meldingene vil også være gunstige i forhold til pris. Prisen for datatrafikk i GPRS og UMTS er hos Telenor 10 kroner pr. MB. Dette betyr at med en meldingstørrelse på 1 kilobyte som i IMATIS kan man sende 1000 meldinger for 10 kroner.

Når det gjelder handover-delay i en SIP-basert løsning er det gjort målinger av dette i [25] der den totale forsinkelsen med forsinkelse i nettet og tiden det tar for handover i WLAN og GPRS ligger på maksimalt 13 sekunder. Dette forsøket er gjort med en løsning som ligner veldig på vår. En løsning der SIP tar seg av data-trafikken og MIP tar seg av nettverkstilkoblinger og roaming. Det skal også påpekes at dette eksperimentet er gjort med handover av en videosamtale slik at den reelle forsinkelsen i vår løsning vil være mindre. Vi mener at denne forsinkelsen ikke vil gi store problemer i forbindelse med handover for en mobil enhet. I forhold til fremtidig bruk av systemet og mulige utvidelser som IP-telefoni og andre realtime overføringer av lyd og bilde er dette resultatet noe som må legges mer vekt på.

I forhold til stor overhead i SIP-baserte løsninger mener vi at heller ikke dette gir nevneverdige konsekvenser i en slik løsning fordi selve meldingene som sendes er såpass små. Størrelsen på en alarmmelding i IMATIS ligger under 1 kb og det er dermed ikke snakk om store datamengder som skal overføres. Noe som setter til siden problemstillingene som omhandler tilstrekkelig hastighet i UMTS/GPRS. Under våre forsøk viste det seg at RTT for en enkel melding ligger henholdsvis rett over ett sekund i GPRS og rett under ett sekund i UMTS.

Enheter med støtte for både WLAN og UMTS/GPRS finnes foreløpig ikke i stort omfang på markedet. Likevel finnes det et antall tredjeparts klientapplikasjoner som er designet for å håndtere roaming i disse nettverkene. Vi velger å anbefale Birdstep SmartRoaming for å ta seg av dette på bakrunn av informasjon og samtaler med ressurspersoner fra BirdStep. Denne klienten tilbyr prekonfigurering av nettverksprofiler og tar seg av roamingen for en mobil enhet uavhengig av hva slags IP-nettverk den befinner seg i. Fordelen med dette er at man kan konfigurere enheten til å benytte WLAN dersom dette er tilgjengelig. Dersom WLAN ikke er tilgjengelig kan enheten konfigureres til å koble seg til et valgt APN.

Roamingløsningen vil også kunne fungere i GPRS-nett. Dette gir muligheten å implementere systemet i områder som enda ikke er bygd ut med UMTS basestasjoner. Ulempen vil være at man ikke vil oppnå de store hastighetene dersom det skal overføres annet enn XML-meldingene. Det vil likevel være nødvendig å bruke en 3G-enhet siden GPRS-enheter blokkerer all andre kommunikasjon ved oppkobling. Dette forhindrer innkommende/utgående samtaler.

I designet for roamingløsning ble det først vurdert en felles RedirectServer som skulle ta seg av all kommunikasjon. Dette betyr at alle meldingene går til RedirectServer før de sendes ut på UMTS eller WLAN. For at dette skulle implementeres, ville det vært nødvendig å endre nøkkelelementer i det eksisterende IMATIS-systemet. Siden denne

roamingutvidelsen bare er en liten del av hva IMATIS tilbyr, ville denne implementeringen vært tungvint. Valget falt da på å utvikle en modulbasert løsning som kan operere uten store endringer i IMATIS. Med denne løsning, vil IMATIS fungere akkurat slik som før, men hvis en enhet er tilkoblet via UMTS, vil meldingene gå om tilleggsmodulen.

5.3 Sikkerhet

Vi har valgt sikkerhetsmekanismer utfra tilgjengelighet og nytteverdi. Konfidensialitet beskyttes ved hjelp av AES-kryptering. Integritet beskyttes ved hjelp av sjekksum for hver melding som går mellom den mobile enheten og IMATIS. Autentisering av enheter gjøres ved hjelp av brukernavn (UPN) og passord. Dette kunne vært gjort ved hjelp av et digitalt sertifikat og en trusted tredjepart som Verisign. Vi velger å benytte kun brukernavn med etterfølgende kryptert passord og mener dette gir god nok autentisering av bruker. Grunnen til dette er både ressurser i den mobile enheten samt de ekstra kostnadene det måtte medføre å holde disse sertifikatene (et ett års SSL-sertifikat hos Verisign koster 995 dollar [96]).

Når det gjelder selve krypteringsalgoritmene som benyttes er dette AES og SHA-1, AES er nåtidens mest brukte og sikre symmetriske kryptering. SHA-1 er brutt som algoritme men benyttes bl.a for å beskytte Microsofts XBOX-konsoll og i SSL-protokollen. Vi anser derfor at SHA-1 kan benyttes som algoritme for å generere sjekksummer i meldingene. PoC-applikasjonene tar for seg utvalgte sikkerhetsmekanismer. Meldingsutvekslingen er gjort med kryptering og dekryptering av meldingsinnhold. I tillegg etterfølges hver melding av en sjekksum-melding. Bouncycastle har vist seg som en kraftig, men samtidig enkel API å benytte seg av. Vi har valgt å fokusere på å få benyttet de to kryteringsalgortimene i den mest sentrale biten der de skal benyttes, nemlig meldingsutvekslingen. Kryptering og tilhørende verifisering av brukernavn og passord er foreløpig ikke implementert.

En sentral del av systemet og av sikkerhet generelt er *trust*. Trust omhandler det å ”stole på” enkelte deler av systemet. Vi har valgt å truste meldinger fra systemet ut til den mobile enheten. Dette medfører at autentiseringsprosessen kun omhandler den mobile enheten. Dette gjøres både for å spare tid og for å spare den mobile enheten for prosessering.

5.4 Mobil håndholdt enhet

Det første som måtte vurderes var hvilke enheter med WLAN/3G-støtte som er på markedet i dag, og hvilke som er på vei inn i den nærmeste fremtiden. Dette ga grunnlaget for flere mulige veier å gå. Valget for hvilken plattform vi skulle benytte oss av for den videre utviklingen av PoC stod mellom Windows Mobile 5.0 og Symbian OS. Symbian er det mest etablerte operativsystemet for mobile enheter i verden i dag. Det betyr at det finnes bred støtte for utviklingsmuligheter for OSet. Denne brede støtten er samtidig det største problemet for Symbian. Symbian er bygget opp av en JVM sammen med en konfigurasjon og en profil. I tillegg til dette finnes valgfrie pakker

som kan implementeres i enhetene. Mulighetene er mange i den vanligste konfigurasjonen (CDLC) og profilen (MIDP).

Noe av problemet er at mobilprodusentene fritt også benytter seg av muligheten til å begrense visse deler av funksjonaliteten. Et eksempel på dette er *PushRegistry* som i MIDP 2.0 inneholder støtte for oppstart av applikasjoner basert på innkommende nettverkshendelser. I oppgaven er det påpekt at roamingløsningen skal fungere uten brukerinteraksjon. Dette betyr at applikasjonen skal automatisk bytte nettverk i bakgrunnen og kun varsle brukeren når det kommer inn et oppdrag fra IMATIS. Det viser seg at alle telefoner vi har testet med støtter *PushRegistry* for SMS og MMS, men ikke Socket. Dette er noe mobilprodusentene har fjernet med tanke på sikkerhet.

5.5 Bruksområder

Pr. idag vil nok ikke løsningen med en enkelt enhet innføres på St. Olavs Hospital. Grunnen til dette er både kostnader ved innkjøp av nye enheter samt tilgangen på enheter som støtter både WLAN og UMTS. Vi velger derfor å se på bruksområder for forenkling av SMS-løsningen samt bruksområder for løsningen med en enkelt enhet.

5.5.1 Bruksområder med to enheter:

I både ”Portøroppdrag” og ”NurseCall” benyttes idag manuelt svar på SMS dersom personellet befinner seg på utsiden av det trådløse nettverket. Dette medfører problemer for personell som utfører oppgaver samtidig som innkommende oppdrag tikker inn. En portør som mottar et oppdrag samtidig som de flytter en pasient vil bruke lang tid på å taste inn sin svarkode samtidig som man utfører et oppdrag. Et eksempel på dette kan være en portør som flytter en seng med en pasient og samtidig skal kvittere for mottatt oppdrag. Dette vil både gå utover selve utførelsen av oppdrag samt koste mer tid. Med en løsning som implementerer et enkelt menyvalg ved innkommende SMS vil portøren være i stand til å opprettholde god kommunikasjon med pasienten og spare tid. Det viktige i portørtjenesten vil være tidsbruken i det øyeblikket man mottar meldingen. Med dette menes responstiden på et innkommende oppdrag. Forsinkelser i forhold til dette vil medføre uønskede forsinkelser i systemet. Når oppdraget er utført vil derimot portøren ha tid til å skrive en kvittering på at oppdraget er utført. Det viktige vil til enhver tid være den umiddelbare responsen som sier noe om hvorvidt oppdraget er godtatt eller ikke. Med vårt forslag til SMS-løsning som demonstrert i kap 4.2 vil slike svar bli autogenerert med grunnlag i et menyvalg. I tillegg vil man være sikret raskere responstid ved NurseCall.

5.5.2 Bruksområde for en samlet enhet:

Vår løsning baseres direkte på casene ’NurseCall’ og ’Portøroppdrag’ som allerede finnes i IMATIS ved St. Olavs. Ved å implementere denne løsningen, vil helsepersonell kunne benytte seg av en håndholdt enhet og samtidig være i kontakt med sykehusnettverket selv om de befinner seg utenfor sykehusets eget trådløse nett. En enkelt enhet medfører forenkling for sykehuspersonell ved at de slipper å forholde seg

til både IP-telefon og mobiltelefon samtidig. Dette gir en forenkling av informasjonsflyt og fjerner den omfattende bruken av SMS.

I tillegg vil det gjøre arbeidet til operatører lettere som da kun har posisjon og rolle og forholde seg til når oppdrag skal tildeles. I dag gjøres det en del ekstraarbeid i forbindelse med tildeling av oppdrag på SMS. I enkelte tilfeller vil det også være gunstig i forhold til at personell kan ta med seg telefonen hjem og fremdeles få informasjon om gitte hendelser. Dette kan gjelde i tilfeller der personell sitter på spesialkompetanse om gitte pasienter o.l.

Innføring av posisjonering i UMTS-nettet vil åpne mange flere muligheter i forhold til flåtestyring av hjemmesykepleiere og folk på hjemmevakt. Dette åpner også for andre markeder og muligheter i forhold til flåtestyring. Dette gjelder all industri der sentrale enheter har behov for å nå og spore personell/kjøretøyer.

5.6 Fremtidig muligheter

Lokasjon i UMTS kan håndteres ved hjelp av innleide tredjeparter. I den kommende E-serien til Nokia, finnes det en egen lokasjon API. Dette vil mest sannsynlig gjør det mulig for enheten å sende sin posisjon i nettverket til en sentral server. Det vil da være mulig å ha lokasjoner til enheter også mens de er på UMTS, uten å gå gjennom en tredjepart. I tillegg til dette leverer Map Solutions løsninger for posisjonering i GSM-nettet til både Telenor og Netcom. Løsningen kan benyttes sammen med ferdige kartløsninger fra Map Solutions. Selve mobilposisjoneringen leveres som en Web Service og kan derfor konfigureres til å oppdatere på ønskede tidspunkter eller i gitte intervaller. Dette gjør at løsningen kan implementeres ved hjelp av vanlige Web Service-kall og dermed kan integreres i systemet.

Nettverksprofilene på den mobile enheten må konfigureres for å håndtere selve roamingen på den mobile enheten. Dette kan gjøres med Birdsteps SmartRoaming-klient. Dersom en slik type klient ikke kan benyttes må dette løses på en annen måte. J2ME gir ikke tilgang til å operere på nettverkshodene. For å gjøre dette må man ned på et lavere programmeringsnivå som f.eks. C. Dette er støttet veldig variabelt i aktuelle mobiltelefoner. Dersom man velger en ferdig klient som SmartRoaming må det implementeres en eller annen form for kommunikasjon til denne internt i telefonen slik at IMATIS-klienten på telefonen oppdager når mobilen har skiftet nettverk og dermed kan sende REGISTER.

For å benytte seg av nettverksressurser på mobilen må applikasjonen signeres. Dette gjøres ved hjelp av en trusted tredjepart som VeriSign [96]. Beskrivelse av hvordan man signerer en MIDlet finnes i Vedlegg C.

En videre utvikling av denne løsningen på SIP-nivå vil gi muligheter for overføring av flere typer medier til og fra den mobile enheten siden SIP-protokollen støtter realtime overføring av lyd og video. Dette kan utnyttes på flere måter; det kan være en live-stream av pasientmålinger, eller video-stream fra et pasientrom. Det kan også være implementasjon av VoIP slik at all kommunikasjon, inkludert ”vanlige” samtaler, foregår over IP-nettverket.

SIP er kjernen i IMS arkitekturen som implementeres i UMTS R5. Ved å implementere en SIP-basert struktur på meldingsutvekslingen allerede nå, vil systemet være klar for å implementeres i en roamingløsning som støttes i selve mobilnettverket. Da vil det ikke lenger være nødvendig med egne lokasjonsservere som håndterer adresseoppdateringen og bytte av nett for den mobile enheten siden dette vil tilbys av mobiloperatør. Adressering vil bli enklere dersom IPv6 blir implementert som den aksepterte standarden innen implementeringen av UMTS R5. Slik det er nå, er det mange som nøler og tviholder på IPv4. Det er dermed ennå tvil om IPv6 blir den neste standarden eller om det kommer en ny standard som vil tilby enklere muligheter oppgradering av IPv4.

6 KONKLUSJON

I denne oppgaven har det blitt designet en roamingløsning mellom WLAN og UMTS ved å benytte SIP-basert meldingsutveksling og MIP-basert nettverkshåndtering. SIPs muligheter i kommende UMTS-nett gjør at denne løsningen vil være fremtidsrettet og klar for mulige utvidelser. Birdsteps løsning basert på MIP er en av flere produkter som kan håndtere nettverksforbindelser

Vårt utgangspunkt for oppgaven har vært å gjøre færrest mulig endringer i det eksisterende systemet og designe en utenforliggende modul. Designet av roamingløsningen vil bli et eventuelt tillegg til IMATIS Meldingstjener. Den foreslåtte løsningen vil effektivisere gjeldende løsning på St. Olavs Hospital. De ansatte vil spare tid i forhold til det å måtte håndtere flere enheter og svare manuelt på SMS. Omfattende bruke av SMS benyttes i dag på sykehuset grunnet gammel bygningsmasse uten trådløs dekning. Dermed vil SMS-løsning for generering av svarmeldinger spare helsepersonell for tid. Dette gjør at de oppnår bedre pasientkontakt under oppdrag.

Det er vist at det er mulig for en mobil enhet å kommunisere med en server via UMTS og GPRS. Det er også vist at det er mulig å lese og endre innkommende XML-meldinger på en mobil enhet, samt sende svarmelding tilbake til server på et WLAN. Bevaring av datasikkerhet er vist gjennom AES-kryptering av meldingsinnholdet og SHA-1 sjekksum på meldinger. Tester gjort av RTT ved meldingsutveskling viser at transportforsinkelsen for disse meldingene er overkommelig. Forsøk med adressering i mobilnett har vist at det er mulig å implementere dette på en mobil enhet og samtidig være tilgjengelig (offentlig IP-adresse) til enhver tid

Det vil være nødvendig å ta hensyn til hva som tilbys av kommende mobile enheter. De forskjellige operativsystemene som eksisterer og brukes, samt de forskjellige implementeringene av disse, gjør det umulig å produsere en generisk løsning som vil fungere på alle enheter. På grunn av dette vil valget av mobil enhet være avgjørende for videre arbeid.

Dette er ennå et ungt marked med mange muligheter etter hvert som teknologi utvikles. Vi føler vi har klart å gi oversikt over mulighetene som finnes for å løse problemstillingen i den opprinnelige oppgaveteksten. Systemet er ikke beregnet for kommersielt bruk foreløpig, men vi føler at oppgaven gir et godt utgangspunkt for videre arbeid med et eventuelt salgbart produkt.

7 REFERANSER

[1] Moderniseringsdepartementet, ”IT for en enklere helse- og omsorgssektor”, Mai 2006
http://odin.dep.no/fad/modernisering/handlingsplan/forenkling/050001-140021/dok-bn.html

[2] Wikipedia, “2G”, Januar 2006,
http://en.wikipedia.org/wiki/2G

[3] Wikipedia, “2.5G”, Januar 2006,
http://en.wikipedia.org/wiki/2.5G

[4] Wikipedia, “2.75G”, Januar 2006,
http://en.wikipedia.org/wiki/2.75G

[5] Wikipedia, “3G”, Januar 2006,
http://en.wikipedia.org/wiki/3G

[6] CellularOnline, “GSM Data Speed Evolution to UMTS”, April 2006,
http://www.cellular.co.za/data_speed_evolution.htm

[7] Wikipedia, “IMT2000”, April 2006,
http://en.wikipedia.org/wiki/IMT2000

[8] Tektronix, UMTS Protocols and Protocol Testing
http://www.rfpeople.com/docs/umts.pdf

[9] Frode Sørensen, Moderne IP-Nett, ISBN: 82-7772-279-6

[10] Wikipedia, “2G”, Januar 2006,
http://en.wikipedia.org/wiki/2G

[11] Geir M. Køien, “An Introduction to Access Security in UMTS”, Januar 2006,

[12] Wikipedia, “WLAN”, Januar 2006,
http://en.wikipedia.org/wiki/WLAN

[13] Arild Haglund, “Forelesning WLAN, Fag IKT 501”, Januar 2006,

[14] Tutorial-Reports, “Wireless LAN (Wifi) Tutorial”, Januar 2006,
http://www.tutorial-reports.com/wireless/wlanwifi/wifi_architecture.php

[15] Wikipedia, “802.11”, Januar 2006,
http://en.wikipedia.org/wiki/802.11

[16] Wikipedia, “Service Set Identifier”, juni 2006
http://en.wikipedia.org/wiki/SSID

[17] Wikipedia, “Ad-Hoc”, Januar 2006,
http://en.wikipedia.org/wiki/Adhoc

[18] UMA Technology, ”UMA Overview”, Februar 2006
http://www.umatechnology.org/overview

[19] Wikipedia, ”Generic Access Network”, Februar 2006
http://en.wikipedia.org/wiki/Unlicensed_Mobile_Access

[20] Motorola, ”Motorola Seamless Mobility Solutions”
http://www.motorola.com/mot/doc/5/5550_MotDoc.pdf

[21] Tech-Invite, “UMTS Network Architecture -- Evolution: from 2G-GSM to 3G—UMTS Release 5”, April 2006
http://www.tech-invite.com/Ti-ims-releases.html

[22] Wikipedia, “IP Multimedia Subsystem”, April 2006
http://en.wikipedia.org/wiki/IP_Multimedia_Subsystem

[23] Siemens, “Siemens IP Multimedia Subsystem (IMS)
http://www.siemens.com/Daten/siecom/Germany/COM/Internet/Mobile_Networks/WORKAREA/com_mnde/templatedata/Deutsch/file/binary/WP_IMS_1306653.pdf

[24] Alcatel, ”Fixed Mobile Convergence”
http://www.itcss16.ua.ac.be/presentations/lieve_bos.pdf

[25] RFC 2002, C. E. Perkins, ”IP mobility support”, oktober 1996,
http://www.ietf.org/rfc/rfc2002.txt

[26] C. E. Perkings ”Mobile networking through mobile IP”, 1998
http://portal.acm.org/citation.cfm?id=613224

[27] Lucent Technologies, W.A. Romjin, D. Plas, D. Bijwaard, E. Meeuwissen, G. Van Oijen, ”Mobility managment for SIP sessions in a heterogeneous network enviroment”, 2004
http://www.interscience.wiley.com

[28] A.G. Valko, ”Cellular IP: A new approach to Internet host mobility”, januar 1999,
http://portal.acm.org/citation.cfm?id=505758

[29] Birdstep Technologies, ”Birstep intelligent mobile IP client”, 2002
http://www.birdstep.com/wireless_infrastructure/mobile_ip.php3

[30] E. Wedlund, H. Schulzrinne, ”Mobility support using SIP”, 1999
http://www.cs.columbia.edu/~hgs/papers/Wedl9908_Mobility.pdf

[31] C.E. Perkins, D. Johnson, ”Route optimization in mobile IP”, februar 1999
http://mosquitonet.stanford.edu/mip/draft-ietf-mobileip-optim-08.txt

[32] RFC 2003, C.E. Perkins, ”IP encapsulation within IP”, oktober 1996
http://www.ietf.org/rfc/rfc2003.txt

[33] X. Zhao, C. Castellucia, M. Baker, ”Flexible network support for mobility”, oktober 1998,
http://portal.acm.org/citation.cfm?id=288274

[34] RFC 3261, J. Rosenberg et al,, ”SIP: Session Initiation Protocol”, Juni 2002,
http://www.rfc-archive.org/getrfc.php?rfc=3261

[35] KK Tan, HL Gob, ”Session Initiation Protocol”, 2002

[36] K. Arabshian, H. Schulzrinne, ”A SIP-based medical event monitoring system”, juni 2003
http://ieeexplore.ieee.org/search/srchabstract.jsp?arnumber=1218720&isnumber=27395&punumber=8645&k2dockey=1218720@ieeecnfs&query=%28+sip+based+medical+event+%3Cin%3Emetadata+%29&pos=0

[37] Lucent Technologies, W.A. Romjin, D. Plas, D. Bijwaard, E. Meeuwissen, G. Van Oijen, ”Mobility managment for SIP sessions in a heterogeneous network enviroment”, 2004
http://www.interscience.wiley.com

[38] Wikipedia, “Internett Protocol”, Mars 2006,
http://en.wikipedia.org/wiki/Internet_Protocol

[39] Wikipedia, “IPv4”, Mars 2006,
http://en.wikipedia.org/wiki/IPv4

[40] Wikipedia, “IPv6”, Mars 2006,
http://en.wikipedia.org/wiki/IPv6

[41] Wikipedia, “DHCP”, Mars 2006,
http://en.wikipedia.org/wiki/DHCP

[42] Wikipedia, “NAT”, Mars 2006,
http://en.wikipedia.org/wiki/Networlk_Address_Translation

[43] Wikipedia, “APN”, Mars 2006,
http://en.wikipedia.org/wiki/Access_Point_Name

[44] Rune Karlsen, Netcom

[45] Wikipedia, ”Evolution to 3G”, April 2006
http://en.wikipedia.org/wiki/Evolution_to_3G

[46] H. Kremling, “B3G and 4G: What is it good for?”, presented at 14th IST Mobile & Wireless Communications summit, Dresden, Germany, 2005,
http://www.mobilesummit2005.org/session.php?session=103

[47] Wikipedia, ”EDGE”, Mai 2006
http://en.wikipedia.org/wiki/Enhanced_Data_Rates_for_GSM_Evolution

[48] Wikipedia, ”UMTS”, Mai 2006-04-28
http://en.wikipedia.org/wiki/Umts

[49] Wikipedia, ” High-Speed Downlink Packet Access”, Mai 2006
http://en.wikipedia.org/wiki/HSDPA

[50] Wikipedia, “High-Speed Uplink Packet Access”, Mai 2006
http://en.wikipedia.org/wiki/High-Speed_Uplink_Packet_Access

[51] Forrester Research, Consumer Fixed-Mobile Substitution Persists, September 2004

[52] ABI Research, Broadband Wireless, Last Mile solutions, 1Q 2004

[53] IDC EMEA, WLAN Tracker Summary, 3Q2004

[54] Digi.no, “Stadig flere ringer over WLAN”, Februar 2006
http://digi.no/php/art.php?id=289777

[55] Michael Yuan, “Securing your J2ME/MIDP apps”, Juni 2002
http://www-128.ibm.com/developerworks/library/j-midpds.html

[56] Matt Bishop, “Computer Security, Art and Science”, ISBN: 0-201-44099-7

[57] Mark Stamp, “Information Security, Principals and Practice”, ISBN: 0-471-73848-4

[58] Wikipedia, “Symetric-key algorithm”, April 2006
http://en.wikipedia.org/wiki/Symmetric_key_algorithm

[59] Wikipedia, "RC4", April 2006
http://en.wikipedia.org/wiki/RC4

[60] Wikipedia, "DES", April 2006
http://en.wikipedia.org/wiki/DES

[61] Wikipedia, "AES", April 2006
http://en.wikipedia.org/wiki/AES

[62] Wikipedia, "Public-key Crypography", April 2006
http://en.wikipedia.org/wiki/Public-key_cryptography

[63] Wikipedia, "Diffie-Hellman", April 2006
http://en.wikipedia.org/wiki/Diffie-Hellman

[64] Wikipedia, "RSA", April 2006
http://en.wikipedia.org/wiki/RSA

[65] Wikipedia, "Hash Function", April 2006
http://en.wikipedia.org/wiki/Hash_function

[66] Symbian, "Symbian OS"
http://www.symbian.com

[67] Todd Sundsted, "J2ME grows up"
http://www-128.ibm.com/developerworks/library/j-j2me/

[68] Wikipedia, "J2ME" Mars 2006
http://en.wikipedia.org/wiki/J2me

[69] Sun Microsystems, Java Map, Mars 2006
http://java.sun.com/developer/onlineTraining/new2java/javamap/index.html

[70] Sun Microsystems, "Mobile Information Device Profile", Mars 2006
http://java.sun.com/products/midp/midp-ds.pdf

[71] Sun Microsystems, "Connected Limited Device Configuration Hotspot Implementation", Mars 2006 http://java.sun.com/j2me/docs/pdf/CLDC_HI112.pdf

[72] Sun Microsystems, "The Java Hotspot Virtual Machine"
http://java.sun.com/products/hotspot/docs/whitepaper/Java_HotSpot_WP_Final_4_30_01.htm

[73] Michael Juntao Yuan, "Data security in mobile Java applications"
http://www.javaworld.com/javaworld/jw-12-2002/jw-1220-wireless.html

[74] Mobile Review, "Preview of the operating system Windows Mobile 2005", Mars 2006
http://www.mobile-review.com/pda/articles/wm2005-magneto-en.shtml

[75] MSDN, "Why Target Windows Mobile 5.0?", Mars 2006
http://msdn.microsoft.com/mobility/windowsmobile/howto/windowsmobile5/why/default.aspx

[76] Windows Mobile, "What is Windows Mobile? FAQ", Mars 2006
http://www.microsoft.com/windowsmobile/about/faq.mspx

[77] Wikipedia, "Windows Mobile", Mars 2006
http://en.wikipedia.org/wiki/Windows_Mobile

[78] MSmobiles , "Rumors confirmed : next major Windows Mobile version is codenamed PHOTON and it will merge smartphone and Pocket PC", Mars 2006

http://msmobiles.com/news.php/4660.html

[79] ComputerWeekly, "Microsoft's improved Windows Mobile 5.0 will boost applications choice for users", Mars 2006
http://www.computerweekly.com/Articles/2006/02/21/214278/Microsoft'simprovedWindowsMobile50willboostapplicationschoiceforusers.htm

[80] Nokia Phones, "Nokia E60"
http://www.nokia.no/phones/e60/

[81] Nokia Phones, "Nokia E61"
http://www.nokia.no/phones/e61/

[82] Nokia Phones, "Nokia E80"
http://www.nokia.no/phones/e80/

[83] Sony Ericsson, "Sony Ericsson P990i"
http://www.sonyericsson.com/spg.jsp?cc=no&lc=no&ver=4000&template=pp1_loader&php=php1_10336&zone=pp&lm=pp1&pid=10336

[84] Qtek, "Qtek 9000", Mars 2006
http://www.qtek.se/sweden/produkter/9000.aspx

[85] HP, HP iPAQ h6340 Pocket PC – Overview
http://h10010.www1.hp.com/wwpc/uk/en/sm/WF05a/21675-21679-21679-21679-297609-4873205.html

[86] Espen Klovning, Birdstep

[87] St. Olavs Hospital, Kundesenter

[88] Harald Sverre Mælum, Vidar Undem, " Utvikling av metoder og løsninger for lokasjonsbaserte tjenester i sykehus", Masteroppgave HiA 2005

[89] CARDIAC, IMATIS-dokumentasjon. Oppgis ved forespørsel

[90] Sony Ericsson, "Sony Ericsson P910i"
http://www.sonyericsson.com/spg.jsp?cc=no&lc=no&ver=4000&template=pp1_loader&php=php1_10183&zone=pp&lm=pp1&pid=10183

[91] Nokia Phones, "Nokia N70"
http://www.nokia.no/phones/n70/

[92] Sony Ericsson Developer Forums, "Thread: PushRegistry with socket", Mars 2006
http://developer.sonyericsson.com/thread.jspa?threadID=24003&tstart=15

[93] Andreas Häber, Doktorgradsstipendiat ved Fakultet for Teknologi, Mobile Communication Group, Agder Mobility Lab, Agder University College

[94] kXML, http://kxml.objectweb.org/project/aboutProject/index.html

[95] BountyCastle, "BountyCastle documentation"
http://www.bouncycastle.org/documentation.html

[96] Verisign, Verisign
http://www.verisign.com/

[97] Map Solutions
http://www.mapsolutions.no/posisjonering.html

[98] Telenor, Telenor Bedrift
http://www.telenor.no/bedrift/produkter/mobil/gprs_priser.html

8 ORDLISTE

2G	2. Generasjons mobiltelefonnettverk - GSM / GPRS
3G	3. Generasjons mobiltelefonnettverk - UMTS
3GPP	Third Generation Partner Project
4G	4. Generasjons mobiltelefonnettverk
802.11x	Standard for WLAN teknologier, der x = bokstav fra a-z
ACK	Acknowledgment
AES	Advanced Encryption Standard
AP	Aksesspunkt
API	Application Program Interface
APN	Acces Point Name
ASK	Amplitude-shift Keying.
BSS	Basic Service Set
CARDIAC	Computer Aided, Research, Development, Instrumentation And Control
CDMA	Code Division Multiple Access
CoA	Care of Address
DES	Data Encryption Standard
DHCP	Dynamic Host Configuration Protocol
DS	Distribution System
DSSS	Direct Sequencing Spread Spectrum
EDGE	Enhanced Datasets for Global Evolution
ESS	Extended Service Set
FA	Foreign Agent
FHSS	Frequency Hop Spread Spectrum
FSK	Frequency Shift Keying
GAN	Generic Access Network
GFSK	Gaussian Frequency Shift Keying
GPRS	General Packet Radio Service
GPS	Global Positioning System
GSM	Global System for Mobile Communication
HA	Home Agent
IETF	Internet Engineering Taske Force
IM	Instant Messagning
IMATIS	Integrert Modulbasert Administrativt Teknisk Informasjons System
IMS	IP Multimedia Subsystem
IMT-2000	International Mobile Telecommunications-2000
IP	Internet Protocol
LAN	Local Area Network
MAC	Media Access Control
MIP	Mobile IP
NAT	Network Address Translation
Node B	Base Station in UMTS/UTRAN
OFDM	Orthogonal Frequency Division Multiplexing

ORL	Olivetti and Oracle Research Laboratory
OSI-Modell	Open Systems Interconnect Modell
PAN	Personal Area Network
PDA	Personal Digital Assistant
PSK	Phase Shift Keying
QoS	Quality of Service
RC4	Rivest Cipher 4
RFID	Radio Frequency Identification
RTT	Round-Trip Time
SIP	Session Initiation Protocol
SQL	Structured Query Language
SSID	Service Set Identifier
TDD	Time Division Duplex
TDMA	Time Division Multiple Access
UE	User Equipment
UMA	Unlicensed Mobile Access, nå kjent som GAN
UMTS	Universal Mobile Telecommunications System
UTRAN	Universal Terrestrial Radio Access Network
VoIP	Voice over IP
VPN	Virtual Private Network
WAP	Wireless Application Protocoll
WCDMA	Wide-Band CDMA
WEP	Wired Enquivalency Privacy
Wi-Fi	Wireless Fidelity
WLAN	Wireless Local Area Network.
WPA	Wi-Fi Protected Access
XML	eXtensible Markup Language

A. DETALJERT UMTS UTRAN INFRASTRUKTUR

UTRAN

Det er introdusert et nytt radioaksessnett i UMTS, kalt UTRAN (UMTS Terrestrial Radio Access Network). Radioteknologien som blir benyttet er CDMA, som erstatter TDMA i GSM-nett. UTRAN er koblet på samme kjernenettverk (CN) som GSM/GPRS. Det er to nye elementer i UTRAN: Node B og RNC [4].

MSC, SGSN og HLR kan utvides til å tilfredsstille UMTS-krav, men det må investeres i nytt utstyr for Node B, RNC og nye mobile enheter

For at kjernenettverket skal kunne kommunisere med UTRAN, må de oppgraderes med nye grensesnitt for påkopling. Dette er nødvendig siden UTRAN er ATM-basert[11]. Man kan sende pakke og linjesvitsjet data over samme kabel, men bruker en annen sendeformat enn f.eks. IP over Ethernet [8]]. I UMTS aksessnettet, er det kommunikasjonsgrensesnitt mellom kontrollere, noe som gjør at en UE kan kommunisere med flere basestasjoner samtidig. Fordelen med dette, er at UE kan kombinere samme informasjon med forskjellig signalstyrke fra flere basestasjoner, og minske sannsynligheten for fading. Dette gjør også at handoverprosedyren blir bedre, da UE allerede har kontakt med neste basestasjon før den kobler ned på den forrige (”soft handover”)[8].

Node B

Node B er den fysiske enheten som tar seg av radiokommunikasjon til UE. Den støtter både FDD og TDD moduser og en leverandør kan spare på kostnader ved å implementere Node B sammen med tilsvarende i GSM-nettet (BTS – Base Transciever Station) [4].

Kommunikasjon mellom Node B og UE foregår over W-CDMA UU-grensesnittet. Mellom Node B og RNC, foregår kommunikasjonene over IUB-grensesnittet (ATM-basert). Det er siste leddet i ATM-linken.

Hovedoppgaven til Node B er å oversetter data til og fra UU-radiogrensesnittet, og foreta diverse QoS-tjenster på signalet. Node B måler signalstyrken til UE og sender denne dataen videre til RNC som igjen bruker det til handover, micro diversity og andre tjenester. Node B kan justere strømforbruket på UE på grunnlag av signalstyrken, som fører til lengre levetid på enheten.

RNC

RNC er en ATM svitsj som samler pakke- og linjesvitsjet data fra en bruker sammen. Den tar seg av all kommunikasjon mellom UTRAN-nettet og kjernenettet. En stor forbedring over GSM-nettet er at RNCer har grensesnitt til hverandre (IUR-grensesnittet), noe som gjør at RNCene er selvstyrende. Kommunikasjon mellom flere RNC og kjernenettet foregår gjennom en såkalt SRNC (Serving RNC). Denne behandler radiosignalene fra andre RNC, over IUR-grensesnittet, samler de og sender de videre til kjernenettet [4].

[11] Asynchronous Transfer Mode – Linklagsteknologi som egner seg til sending av store datamengder raskt.

CN - Core Network

UMTS bruker samme kjernenett som GSM. Dette medfører til gjenbruk av den eksisterende infrastrukturen. For at UTRAN skal få kommunisert direkte med kjernenettet, må det inn med nye kommunikasjonsprotokoller samt litt oppgradering.

Kjernenettet består av en linjesvitsjet del (telefoni) og en pakkesvitsjet del (data). Den linjesvitsjet delen består av en MSC – GMSC, som da sender videre til ISDN-nettverket. Den pakkesvitsjet delen består av en SGSN (Service GPRS Support Node) – GGSN (Gateway GPRS Support Node), som da sender videre til Internett [4],[6].

For håndtering av brukere og å tilknytte enheter til den respektive abonnementet, finnes det noen felles støtteservere for det linje- og pakkesvitsjet nettverket, bl.a HLR (Home Location Register) og VLR (Visitor Location Register)

B. MODULASJONSTEKNIKKER

Informasjonen er hentet fra rapporten til Harald Sverre Mælum og Vidar Undem; ”Utvikling av metoder og løsninger for lokasjonsbaserte tjenester i sykehus” [88].

FHSS

Acronym for frequency-hopping spread spectrum. FHSS is one of two types of spread spectrum radio, the other being direct-sequence spread spectrum. FHSS is a transmission technology used in LAWN transmissions where the data signal is modulated with a narrowband carrier signal that "hops" in a random but predictable sequence from frequency to frequency as a function of time over a wide band of frequencies. The signal energy is spread in time domain rather than chopping each bit into small pieces in the frequency domain. This technique reduces interference because a signal from a narrowband system will only affect the spread spectrum signal if both are transmitting at the same frequency at the same time. If synchronized properly, a single logical channel is maintained.
The transmission frequencies are determined by a spreading, or hopping, code. The receiver must be set to the same hopping code and must listen to the incoming signal at the right time and correct frequency in order to properly receive the signal. Current FCC regulations require manufacturers to use 75 or more frequencies per transmission channel with a maximum dwell time (the time spent at a particular frequency during any single hop) of 400 ms.

DSSS

Acronym for direct-sequence spread spectrum. DSSS is one of two types of spread spectrum radio, the other being frequency-hopping spread spectrum. DSSS is a transmission technology used in LAWN transmissions where a data signal at the sending station is combined with a higher data rate bit sequence, or chipping code, that divides the user data according to a spreading ratio. The chipping code is a redundant bit pattern for each bit that is transmitted, which increases the signal's resistance to interference. If one or more bits in the pattern are damaged during transmission, the original data can be recovered due to the redundancy of the transmission.

OFDM

Short for Orthogonal Frequency Division Multiplexing, an FDM modulation technique for transmitting large amounts of digital data over a radio wave. OFDM works by splitting the radio signal into multiple smaller sub-signals that are then transmitted simultaneously at different frequencies to the receiver. OFDM reduces the amount of crosstalk in signal transmissions. 802.11a WLAN, 802.16 and WiMAX technologies use OFDM.

FDM

Short for frequency division multiplexing, a multiplexing technique that uses different frequencies to combine multiple streams of data for transmission over a communications medium. FDM assigns a discrete carrier frequency to each data stream and then combines many modulated carrier frequencies for transmission. For example, television transmitters use FDM to broadcast several channels at once.

PSK

Short for phase-shift keying, a modulation technique used by modems in which different phase angles in the carrier signal are used to represent the binary states of 0 and 1.
The simplest method of PSK, also called biphase modulation, uses two signal phases - 0 degrees and 180 degrees. The digital signal is broken up according to time into binary digits and the state (1 or 0) of each bit is determined according the state of the bit that preceded it. If the phase of the bit does not change then the state of the signals stays the same. If the phase of the signal changes by 180 degrees, then the signal state changes (from 0 to 1, or 1 to 0).
There are more complex forms of PSK that rely on four or eight phases to transmit data at a faster rate.

C. SIGNERING AV MIDLETS

Verifisering av applikasjonen skjer i to steg. En preverifikasjon hos utvikler og en verifikasjon i terminalen. I tillegg til dette kjører Midleten i en sandbox som begrenser tilgang til visse APIer som nettverk, filtilgang osv. Adgangen til disse bestemmer av om MIDleten er trusted eller ikke. Denne tilgangen settes da i en domain-policy.

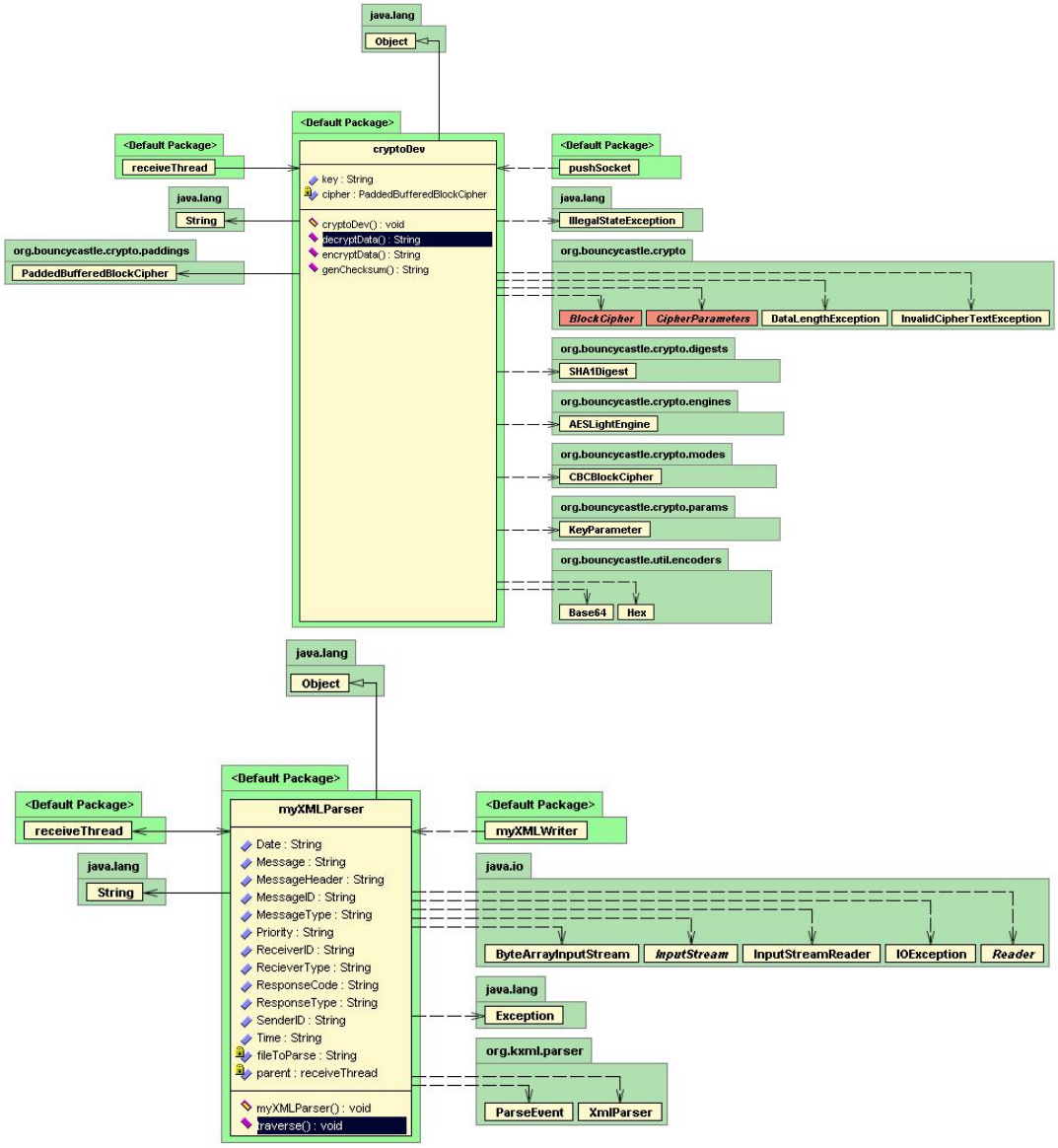
Protection domains er et sett av permissions og interaksjonsmoduser. Disse permissions kan bli gitt automatisk eller av bruker ved godkjennelse. Når en midlet innstalleres blir den gitt et protection domain og får sine permissions og interaksjonsmoduser. Permissions settes i JAD-fila eller i manifest-fila. Tre interaksjonsmoduser eksisterer; blanket (en bekreftelse), session (en bekreftelse pr. sesjon) og oneshot (bekreftelse hver gang).

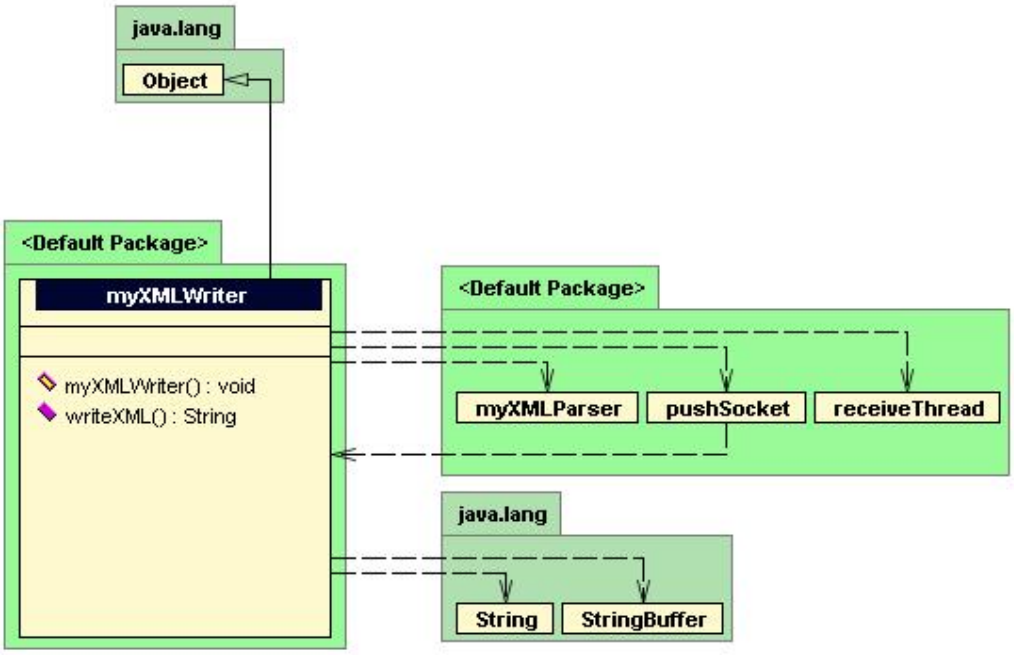
Steg for å signe en MIDlet. Det vil si gjøre en MIDlet trusted ved hjelp av en trusted tredjepart.

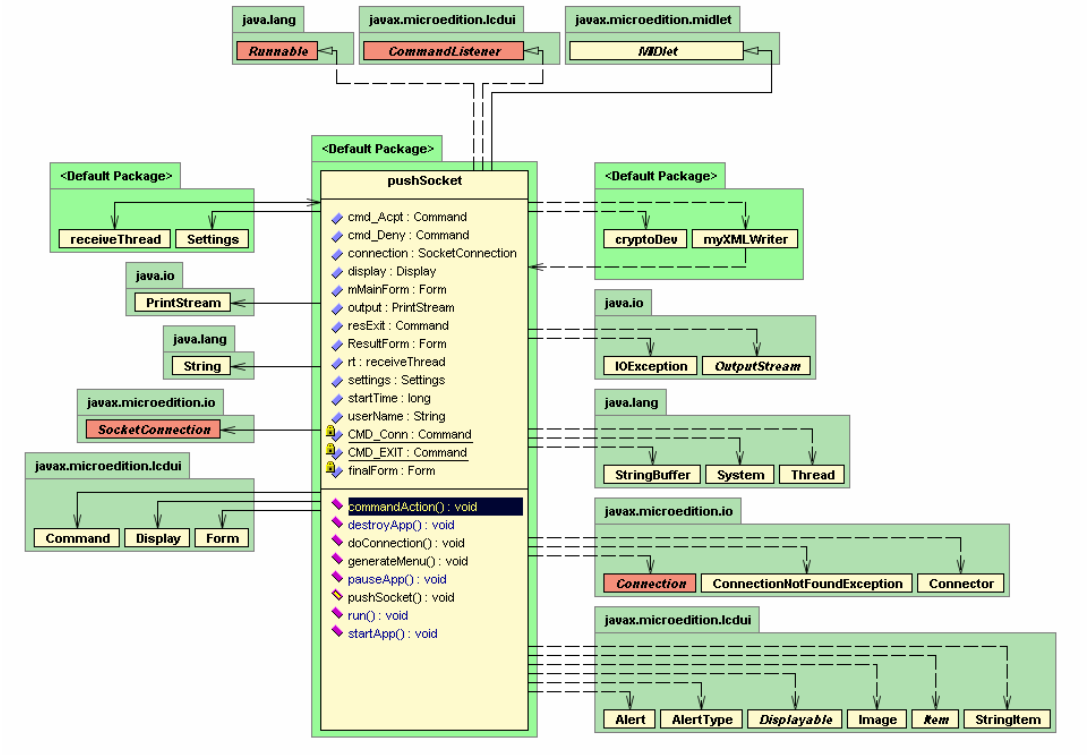
1. Sett opp liste over permissions i JAD eller Manifest-fil.
2. Konstruer et nøkkelpar (public/private) for signering.
3. Signer MIDlet før deployment.
4. Registrer sertifikat med trusted tredjepart, eksempelvis VeriSign eller andre kjente Certificate Authorities (CA).

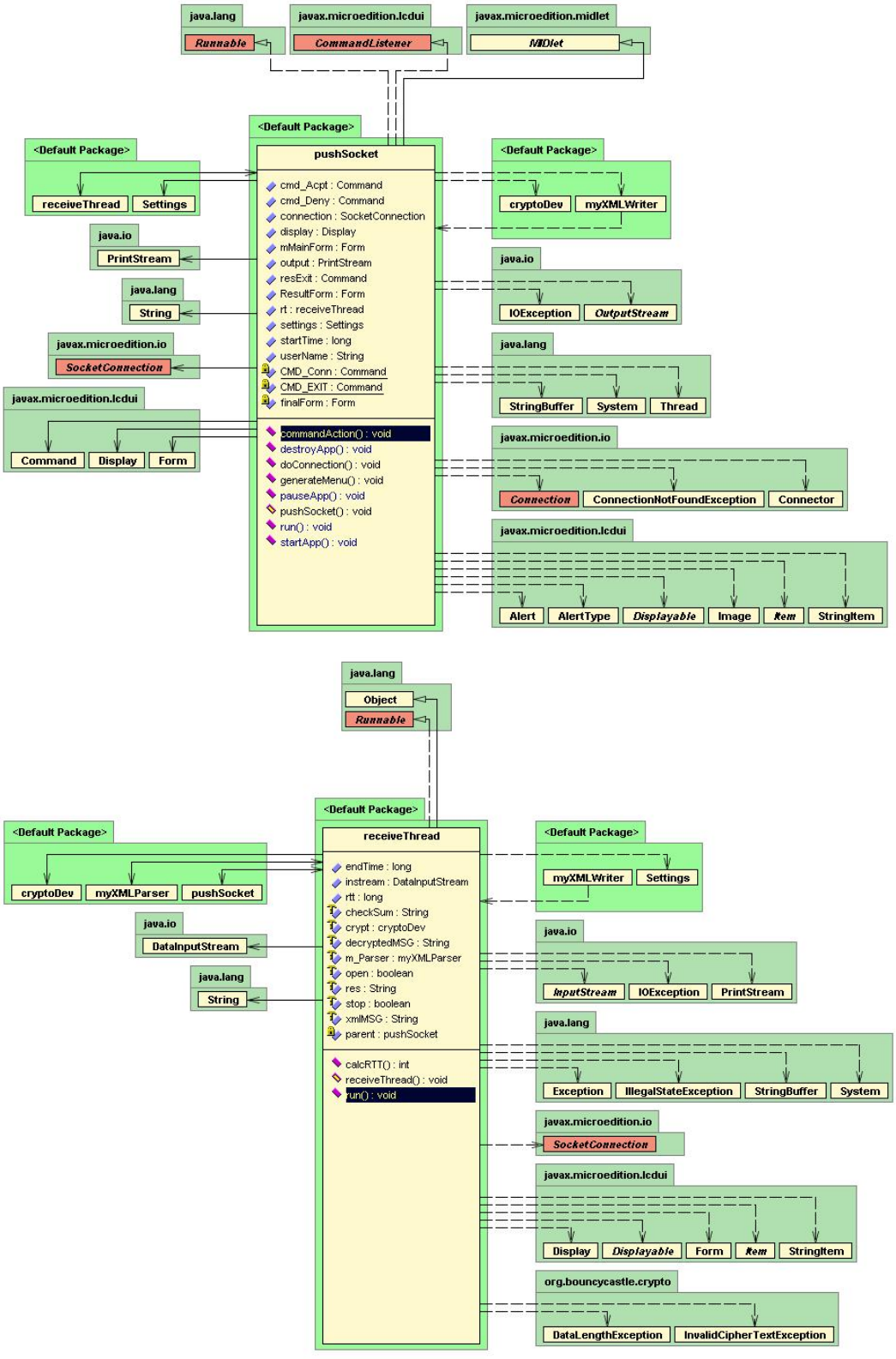
D. UML DIAGRAMMER

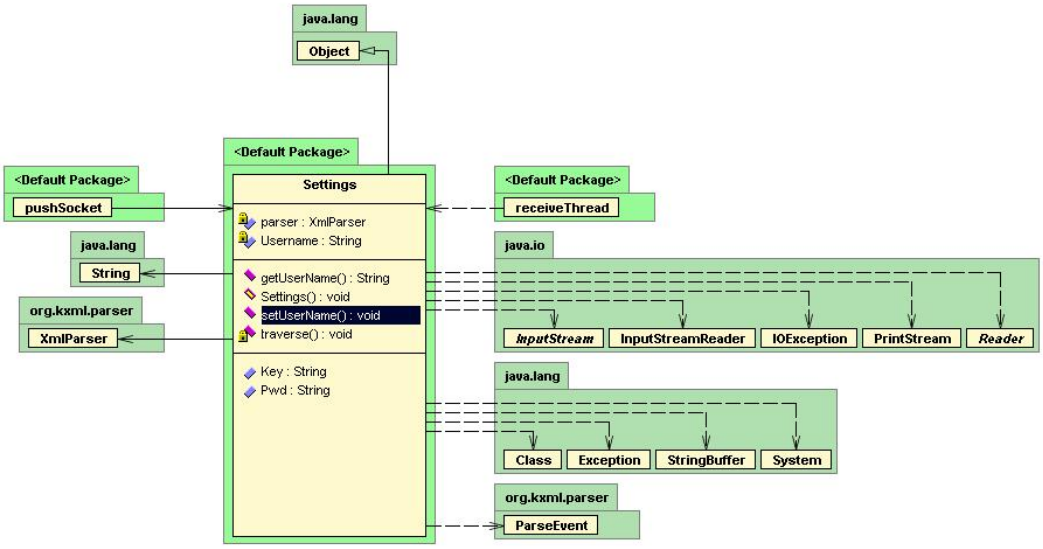
Vedlagt følger en UML-oversikt over klassene som benyttes ved den mest omfattende PoC-testingen.




java.lang
Object
<Default Package>
myXMLWriter
myXMLWriter() : void
writeXML() : String
<Default Package>
myXMLParser
pushSocket
receiveThread
java.lang
String
StringBuffer


java.lang
Runnable
javax.microedition.lcdui
CommandListener
javax.microedition.midlet
MIDlet
<Default Package>
pushSocket
cmd_Acpt : Command
cmd_Deny : Command
connection : SocketConnection
display : Display
mMainForm : Form
output : PrintStream
resExit : Command
ResultForm : Form
rt : receiveThread
settings : Settings
startTime : long
userName : String
CMD_Conn : Command
CMD_EXIT : Command
finalForm : Form
commandAction() : void
destroyApp() : void
doConnection() : void
generateMenu() : void
pauseApp() : void
pushSocket() : void
run() : void
startApp() : void
<Default Package>
receiveThread
Settings
java.io
PrintStream
java.lang
String
javax.microedition.io
SocketConnection
javax.microedition.lcdui
Command
Display
Form
<Default Package>
cryptoDev
myXMLWriter
java.io
IOException
OutputStream
java.lang
StringBuffer
System
Thread
javax.microedition.io
Connection
ConnectionNotFoundException
Connector
javax.microedition.lcdui
Alert
AlertType
Displayable
Image
Item
StringItem


java.lang
Runnable
javax.microedition.lcdui
CommandListener
javax.microedition.midlet
MIDlet
<Default Package>
pushSocket
cmd_Acpt : Command
cmd_Deny : Command
connection : SocketConnection
display : Display
mMainForm : Form
output : PrintStream
resExit : Command
ResultForm : Form
rt : receiveThread
settings : Settings
startTime : long
userName : String
CMD_Conn : Command
CMD_EXIT : Command
finalForm : Form
commandAction() : void
destroyApp() : void
doConnection() : void
generateMenu() : void
pauseApp() : void
pushSocket() : void
run() : void
startApp() : void
receiveThread
Settings
java.io
PrintStream
String
javax.microedition.io
SocketConnection
Command
Display
Form
cryptoDev
myXMLWriter
IOException
OutputStream
StringBuffer
System
Thread
Connection
ConnectionNotFoundException
Connector
Alert
AlertType
Displayable
Image
Item
StringItem
Object
receiveThread
endTime : long
instream : DataInputStream
rtt : long
checkSum : String
crypt : cryptoDev
decryptedMSG : String
m_Parser : myXMLParser
open : boolean
res : String
stop : boolean
xmlMSG : String
parent : pushSocket
calcRTT() : int
receiveThread() : void
run() : void
myXMLParser
pushSocket
DataInputStream
InputStream
Exception
IllegalStateException
org.bouncycastle.crypto
DataLengthException
InvalidCipherTextException


java.lang
Object
<Default Package>
Settings
parser : XmlParser
Username : String
getUserName() : String
Settings() : void
setUserName() : void
traverse() : void
Key : String
Pwd : String
<Default Package>
pushSocket
java.lang
String
org.kxml.parser
XmlParser
<Default Package>
receiveThread
java.io
InputStream
InputStreamReader
IOException
PrintStream
Reader
java.lang
Class
Exception
StringBuffer
System
org.kxml.parser
ParseEvent