



GLOBETEAM



POLSAG:
Performance review

17. november 2011
(omredigeret og afsluttet pr. 16.
januar 2012)

Jakob Røjel, Jesper Hvid &
Morten Strunge Nielsen

Ledelsesresumé

Performance review'et er igangsat af Rigspolitiet på baggrund af observationerne i det tekniske review af POLSAG, som blev gennemført i forbindelse med POLSAG-undersøgelsen, som Boston Consulting Group (BCG) udførte for Rigspolitiet, Justitsministeriet og Finansministeriet i foråret 2011.

Performance reviewet er gennemført af Globeteam med støtte fra Devoteam i perioden juli til november 2011. I december 2011 blev der tilføjet to ekstra afsnit ("Test i produktionsmiljøet" og "Analyse af anvendelse af randsystemer i de enkelte test cases"), som er baseret på de sidste aftestninger i november måned samt analyserne af samme.

Der blev udført et antal præciseringer i formuleringerne primo januar 2012. Medio januar blev der endvidere gennemført en ombearbejdning af ledelsesresumé og der blev tilføjet et ekstra appendix ("Appendix E. Supplerende information") med henblik på at indarbejde de seneste informationer omkring leverandørens indsats for at optimere POLSAGs performance for at kunne bestå den kontraktuelle svartidsprøve i andet halvår 2011, således at rapporten er samlet i et enkelt dokument.

Det præciseres at reviewet, formelt set er udført på bestilling af Rigspolitiet med Devoteam som leverandør med Globeteam som underleverandør. Efter aftale med Rigspolitiet har Devoteam ikke været involveret i udarbejdelsen og kvalitetssikringen af ombearbejdningen af rapporten medio januar 2012.

Formål og afgrænsning af opgaven

Det helt overordnede formål med performance review'et er at tilvejebringe et kvalificeret svar på, om POLSAG-løsningen, rent performancemæssigt, er i stand til at håndtere en udrulning til Midt & Vestjylland samt de efterfølgende politikredse.

Selve performance review'et har til opgave at gennemføre en dybdegående analyse af POLSAGs ydeevne og skalerbarhed.

Performance review'et var som udgangspunkt defineret til at indeholde:

- *Etablering af et testmiljø og en metode til at foretage kvantitative performancemålinger på POLSAG. Med kvantitative målinger dokumenteres, hvordan systemet opfører sig under højere belastning. Eftersom målingerne afvikles i et testmiljø vil svartider ikke være repræsentative for driftsmiljøet.*
- *Dokumentation af POLSAGs performancekarakteristika baseret på mellem 8 og 10 anvendelsesscenarier som er ofte anvendt i systemet, herunder afgive anbefalinger for opdateringer, der vil forbedre den af brugeren oplevede performance. Scenarierne forventes at dække mellem 25 og 30 skærbilleder. Scenarierne er udpeget af Rigspolitiet.*
- *Afdækning af belastning af eksterne systemer for de højfrekvente scenarier.*
- *Udførelse af en load/stress-test i samarbejde med CSC/Scanjour, der skal afdække systemets opførsel under hård belastning.*

Det har ikke været en del af performance reviewet's opgave at udføre rettelser i POLSAG, da dette henhører under leverandørens (Alliancens) ansvarsområde.

Den primære analyse og dermed det bærende grundlag for performance reviewet er gennemført i Rigspolitiets POLSAG uddannelsesmiljø (benævnes herefter **uddannelsesmiljø**). Uddannelsesmiljøet blev valgt, da det på tidspunktet for performance reviewets start, var det miljø, som bedst afspejlede bestyrelsen af produktionsmiljøet. Dette valg skal også ses i lyset af at Rigspolitiet skulle beslutte om den nuværende POLSAG leverance kunne udvides til også at blive anvendt i Midt- og Vestjylland.

Parallelt med performance reviewet har leverandøren henover andet halvår 2011 udført performanceoptimering med henblik på at kunne bestå den kontraktuelt fastsatte svartidsprøve. Denne performanceoptimering er gennemført i Rigspolitiets performancetestmiljø (benævnes herefter **performancetestmiljø**).

Globeteam har i samarbejde med Rigspolitiet og leverandøren i december 2011 foretaget en indledende vurdering af de af leverandørens opgivne resultater (benævnes herefter **opfølgende performanceanalyse**), samt disses eventuelle implikationer for konklusionerne og anbefalingerne afledt af analysen i uddannelsesmiljøet. Disse informationer er samlet i Appendix E og afspejlet i ledelsesresumeeet. Den resterende del af rapporten afspejler fortsat udelukkende det arbejde, der er udført af Globeteam med støtte fra Devoteam i perioden juli til november 2011.

Medio december har Rigspolitiet revurderet belastningsmodellen, hvilket har resulteret i en ny model, hvor brugen af POLSAG har vist sig at være mere jævnt fordelt end antaget i den oprindelige model. Målt i "rå" requests ligger belastningen i den opdaterede model på 40-50% af den model, som danner udgangspunkt for aftestningen i uddannelsesmiljøet.

Det er dog vigtigt at bemærke at aftestningen i uddannelsesmiljøet dokumenterer, at der er meget stor forskel på, hvor meget de forskellige requests belaster POLSAG. Der er altså ingen sikkerhed for at "alt andet lige" er korrekt i denne sammenhæng.

Dette dokument udgør den endelige afrapportering fra performance reviewet og konsoliderer alle de essentielle informationer, data, delkonklusioner og overordnede konklusioner fra analysen.

Performance review'et har endvidere genereret en række detaljerede målinger og en del forløbsdokumentation, som *ikke* er en del af rapporten.

Konklusion og anbefalinger

Performance Review'et i uddannelsesmiljøet bekræfter fuldt ud Rigspolitiets formodning om, at POLSAG-løsningen lider under alvorlige problemer for såvidt angår systemets ydeevne og skalerbarhed. Reviewet har endvidere demonstreret væsentlige problemer med systemets stabilitet i uddannelsesmiljøet, eftersom det har været svært at gennemføre større testforløb, fordi miljøet ofte bliver ustabilt eller bliver ramt af fatale fejl, når det sættes under pres.

POLSAG-løsningen i uddannelsesmiljøet er således meget langt fra at kunne imødekomme Politiets behov og den tilsigtede og aftalte anvendelse af POLSAG. Imødekommelse af Politiets behov for den fremtidige tilsigtede anvendelse af POLSAG forudsætter væsentlige forbedringer i hver enkelt servers ydelse samt i systemets evne til at skalere ved tilføjelse af flere servere. Det vil herunder særligt være nødvendigt at demonstrere, at POLSAG-løsningen kan bringes til at skalere tilnærmelsesvist lineært henover mindst 16 servere.

Testresultaterne fra uddannelsesmiljøet indebærer, at Globeteam må anbefale Rigspolitiet at fastholde deres nuværende beslutning om *ikke* at rulle POLSAG ud til Midt/Vest før:

1. Leverandøren kan vise, at POLSAG-løsningen kan skalere lineært op til mindst 16 servere.
2. Leverandøren mindst har femdoblet POLSAG-løsningens ydeevne (dvs. effektivisering af applikationen eller driftsplatformens ydeevne), idet servermiljøet ellers vurderes at skulle være væsentligt større end planlagt.
3. Leverandøren kan demonstrere, at POLSAG-løsningen kan klare en meget høj spidsbelastning uden at blive ustabil eller helt eller delvist gå ned.

Desuden er det Globeteams anbefaling, at:

4. POLSAG tilpasses, således at generering af døgnrapporter ikke er i stand til at belaste hele systemet på en sådan måde, at det fører til væsentligt forøgede svartider eller nedsat brugbarhed.
5. det bliver undersøgt, hvilken betydning stigende datamængder vil have for POLSAG-løsningens ydelse.

Den opfølgende performanceanalyse i performancetestmiljøet indikerer, at der er sket en væsentlig forbedring i svartider og ydeevne. Leverandøren har oplyst, at problemerne i forhold til skalerbarhed og stabilitet også skulle være løst samt at der er gennemført optimering i forhold til døgnrapporten. Der vurderes dog at være en høj grad af usikkerhed forbundet med hvilke resultater leverandøren reelt har opnået, samt om de er tilvejebragt på en måde, som kan overføres til Rigspolitiets produktionsmiljøer. Det tilrådes derfor at anlægge en høj grad af forsigtighed i forhold hvad der konkluderes på denne baggrund, indtil denne usikkerhed måtte være elimineret¹.

Hvis alle leverandørens rapporterede resultater kan valideres og overføres til Rigspolitiets produktionsmiljøer vil der være behov for at lave nedennævnte justeringer af konklusionerne (angivet i kursiv).

Anbefaling 2

Ændring af belastningsmodellen reducerer antallet af "rå" requests med ca. en faktor 2.5. Dette gør at anbefalingen alt andet lige vil skulle ændres til en faktor 1,7-2,45.

Målinger på performancetestmiljøet indikerer at antallet af "rå" requests er ca. halveret. Der er indikationer på at performancetestmiljøet er i stand til at behandle requests væsentligt hurtigere end det er tilfældet i uddannelsesmiljøet.

Det må derfor regnes for sandsynligt at POLSAG kan håndtere det nødvendige antal requests. Dette skal valideres i praksis.

Anbefaling 3

Leverandøren har rapporteret at have rettet to alvorlige fejl i standard ScanJour Captia, som de mener er årsagen til de rapporterede fejl i uddannelsesmiljøet. Scanjour påstår at have testet POLSAG ved væsentligt højere belastning end det er sket i uddannelsesmiljøet uden at opleve

¹ Appendix E indeholder en kortfattet gennemgang af de supplerende oplysninger og ændringer, der er indtruffet efter eller sideløbende med aftarstningen af Uddannelsesmiljøet, som ligger til grund for rapportens hovedafsnit. Det må på den baggrund anbefales at nærlæse appendiks E, da især resultaterne fra det opdaterede performancemiljø og de deraf følgende konklusioner ikke er verificeret af andre parter end leverandøren selv.

stabilitetsproblemer. Der angives samtidig at være løst et alvorligt problem med frigivelse af databaseressourcer med tilhørende memory leaks.

Det vurderes som værende positivt, at leverandøren har afviklet en stresstest og angiveligt har fundet løsninger på de rapporterede problemer. Anbefalingen om at dette skal valideres i praksis gælder naturligvis fortsat. Der gøres opmærksom på, at det ikke var alle de oplevede problemer i uddannelsesmiljøet, der var af en art og type der kunne indrapporteres.

Anbefaling 4

Leverandøren har rapporteret, at der er foretaget væsentlige performanceoptimeringer i døgnrapporten. Der foreligger imidlertid ingen formel dokumentation eller resultater. Dette skal valideres i praksis.

I det følgende gennemgås kort problemstillingerne omkring dårlig performance, manglende skalerbarhed og ustabil drift som er afdækket i uddannelsesmiljøet, valideret i produktionsmiljøet og forsøgt adresseret af leverandøren i performancetestmiljøet. Derefter beskrives mulighederne for udbedring af problemer.

Performance

Dimensioneringen af POLSAGs servermiljø er – lettere forsimplet – baseret på en forudsætning om, at hver enkelt server (en såkaldt "simpel server"²) er i stand til at dække behovene i en kreds.

Resultaterne af performance review-aftestningen i uddannelsesmiljøet angiver, at POLSAG-servermiljøet er meget kraftigt underdimensioneret relativt til forventningen om at en server omtrent vil være i stand til at håndtere belastningen fra en kreds. Det formodes at hver server som minimum vil skulle have fem gange højere ydeevne end den nuværende server for at være i stand til at håndtere den belastning, som genereres i en politikreds.

Performance review'et har afdækket, at selv når serverne opgraderes til såkaldte komplekse servere og udvides til at bestå af fire komplekse servere pr. kreds, er det fortsat ikke muligt at nå igennem de mix af aktiviteter, som Rigspolitiet har defineret som værende typiske for en gennemsnitlig kreds, indenfor den stipulerede tidsperiode. Dette vurderes dog også at hænge sammen med den manglende skaleringssevne, eftersom systemet holder op med at skalere lineært efter den anden server.

Resultaterne fra Kriminalforsorgens driftsmiljø giver en formodning om, at det vil være muligt for leverandøren at øge servernes ydeevne ganske væsentligt og dermed undgå en flerdobling af antallet af webservere i installationen. Testresultaterne fra Kriminalforsorgen³ demonstrerer således at man opnår svartider, der er ca. 2½ gange bedre i dette miljø relativt til POLSAG-miljøet.

Den opfølgende performanceanalyse indikerer, at POLSAG i performancetestmiljøet performer betydeligt bedre end uddannelsesmiljøet for så vidt angår requestmængder og servicetider. Det er dog fortsat vores vurdering at blandt andet antallet af requests giver anledning til bekymring, da det vurderes at ligge relativt højt – ikke mindst under indtryk af at applikationen skal afvikles henover

² POLSAG-miljøet er p.t. konfigureret med servere, der er af en dimensionering, som leverandøren betegner "simple servere". Leverandøren opererer også med såkaldt "komplekse servere", som indeholder mere RAM og flere processorkerner.

³ Jvf afsnittet "Test hos Kriminalforsorgen" i rapporten.

Rigspolitiets WAN. Det vurderes derfor at være afgørende, at der bliver gennemført en opdateret test i et produktionslignende miljø med realistiske netværksforsinkelser.

Da Rigspolitiets efterfølgende revurdering af belastningsprofilen samtidig viser, at load for normal belastningen alt andet lige vil skulle halveres, er der opstået en forventning om, at performancetestmiljøet har den ønskede ydeevne. I den forbindelse er det ligeledes positivt, at Rigspolitiet også har konkluderet, at den i belastningsprofilen angivne normale belastning er udtryk for en 1/12-del og ikke en 1/16-del af det samlede load. Derfor må de belastningsafledte problemer alt andet lige forventes at være mindre udtalte. Dette er baggrunden for vurderingen, at kravet om en faktor 5 forbedring i uddannelsesmiljøet formentlig kan reduceres til en faktor 1,7-2,45 forbedring i performancetestmiljøet.

Skalerbarhed

Aftestningen i uddannelsesmiljøet har valideret, at POLSAG-servermiljøet på det nærmeste ikke skalerer, idet den tilnærmelsesvis lineære skalering ophører allerede efter den anden server. Det er af afgørende vigtighed, at systemet som minimum kan skalere tilnærmelsesvist lineært over det antal servere, som POLSAG-miljøet består af. Miljøets evne til at skalere falder meget kraftigt fra dette punkt, og det vil derfor kun have begrænset effekt at tilføje yderligere servere.

POLSAGs produktionsmiljø består pt. af 16 servere, og servermiljøet er – lidt forsimplet betragtet – dimensioneret i forventning om, at én såkaldt "simpel server" kan dække behovene i en politikreds. Performance review'et viser imidlertid, at selv når serverne opgraderes til såkaldte "komplekse servere" og udvides til at bestå af fire komplekse servere pr. kreds, så er det fortsat ikke muligt at opnå en performance, så man kan nå at gennemføre de aktiviteter, som Rigspolitiet har defineret som værende typiske for en gennemsnitlig kreds, indenfor den aftalte tidsperiode.

Leverandøren har i forbindelse med den opfølgende performanceanalyse påstået, at skaleringsproblemerne nu skulle være løst. Leverandøren har på nuværende tidspunkt ikke dokumenteret, hvad det oprindelige problem var og har endnu ikke demonstreret i praksis, at deres påstand er korrekt.

Ustabilitet under pres

Aftestningen i uddannelsesmiljøet har afdækket, at POLSAG ikke opfører sig hensigtsmæssigt under hård belastning. Her risikerer webserverne at gå helt eller delvist ned, ligesom databasen i værste fald kan risikere ikke at være tilgængelig for webserverne, hvilket medfører at systemet vil være helt eller delvist uanvendeligt for Rigspolitiet indtil problemet er løst.

Performance Review'et i uddannelsesmiljøet har afdækket, at systemet bliver ustabil, når det sættes under pres, hvilket har medført at de gennemførte tests i mange tilfælde har måttet udføres flere gange, fordi systemet gik helt eller delvist ned under testene. I enkelte tilfælde har det ikke været muligt at gennemføre testene fuldstændigt.

Det er på nuværende tidspunkt uklart, hvilke underliggende årsager der ligger bag ustabiliteten i uddannelsesmiljøet. Dette skyldes, dels at det ikke var muligt at etablere et separat testmiljø med on-line adgang til kildekoden, dels at Globeteam ikke har haft adgang til servermiljøet. Leverandøren har i øvrigt ikke afgivet nogen uddybende informationer omkring nedbruddene.

Leverandøren har efterfølgende oplyst, at der er fundet og løst stabilitetsproblemer med standard Captia i ScanJours eget performancetestmiljø. Leverandøren forventer, at disse rettelser løser de

observerede problemer, som blev identificeret i uddannelsesmiljøet. Dette er endnu ikke bekræftet i en test.

De gennemførte test afspejler "best case"

De udførte tests i uddannelsesmiljøet tog udgangspunkt i en model for den største normale belastning af POLSAG, som Rigspolitiet i forbindelse med performance review'ets start vurderede var forbundet med den daglige produktion. Testene er baseret på belastningen mandag morgen henholdsvis mandag formiddag.

Der skal i den sammenhæng gøres opmærksom på at de tests, der er blevet gennemført i uddannelsesmiljøet som en del af performance review'et, ikke afspejler forventningen til den fulde systembelastning i produktionsmiljøet:

- Testen tager ikke højde for nogen af alle de andre operationer på POLSAG ud over 13 test cases – I performance reviewet er POLSAG-miljøet kun blevet belastet med de 12 ud af de 13 test cases, der har kunnet automatiseres. De gennemførte tests udgør således ikke et fuldt realistisk billede af belastningen, idet POLSAG jo også bliver brugt til at udføre andre arbejdsgange end dem, der er repræsenteret i de udvalgte test cases.
- Belastningen i testen er fuldstændig udjævnet og vil som sådan lægge den mindst mulige kontinuerte belastning på POLSAG-servermiljøet. – De gennemførte tests har været spredt helt ensartet og mest muligt ud over den tid, der har været til rådighed for testen. I den daglige anvendelse kan belastningen ikke forventes at være jævnt fordelt, men derimod udvise store udsving med deraf følgende forøget belastning på hele servermiljøet.
- POLSAG udfører ikke nogen audit og sikkerhedslogning – Testene er gennemført i uddannelsesmiljøet, hvor POLSAG ikke udfører nogen audit og sikkerhedslogning. Det er kendt fra produktionsmiljøet, at POLSAGs logningsdel udløser et større antal SQL-kald og dermed en forøget belastning på databasen. CSC har tidligere oplyst, at deres egne målinger viser, at der kan opnås en 15-20 % reduktion af svartiderne, hvis al sikkerhedslogning fjernes. Det er forventningen, at logningen udelukkende udløser meget simple databasekald, hvorfor dette forhold ikke vurderes at virke ødelæggende for hele analysen.

Med andre ord tegner analysen et "best case"-billede af situationen på det tidspunkt i arbejdsugen, hvor systemet forventes at være udsat for den største normale belastning⁴.

Testene i uddannelsesmiljøet er som sådan gennemført på et niveau, der er mindre end den belastning, der forventes i normal produktion og dækker ikke spidsbelastninger. Globeteam anbefaler derfor, at Rigspolitiet opfatter resultaterne fra performance review'et som de absolutte minimumskrav, der kan og bør stilles til POLSAG.

Rigspolitiet har efter performance review'ets afslutning revurderet den belastningsmodel, der er anvendt i forbindelse med testen i uddannelsesmiljøet. Dette har resulteret i en opdateret model, som i forhold til antal udløste requests udgør ca. en halvering af den belastning POLSAG bør

⁴ Eneste detalje, der ikke er best case er, at test case 0 ikke tager højde for Internet Explorers eventuelle caching af de mest statiske ressourcer. Dvs. alle kørsler af test case 0 er baseret på, at det er første gang klienten indlæser forsiden, hvilket vurderes at være en acceptabel tilnærmelse af den stund klienterne vitterlig vil indlæse forsiden fuldt og helt mandag morgen og dermed giver testen et "fair" bud på den forventede belastning.

udsættes for, for at simulere normal produktion. Dette betyder alt andet lige, at POLSAG må forventes at have færre og mindre udtalte problemer i forhold til ydeevne og stabilitet, end dem der er konstateret i uddannelsesmiljøet

Muligheder for forbedringer

Der er ikke indenfor rammerne af Performance Review'et i uddannelsesmiljøet blevet afdækket et samlet sæt af løsninger, der umiddelbart formodes at ville udbedre de konstaterede problemer.

På det foreliggende grundlag er det vores umiddelbare vurdering, at der ikke findes nogen hurtige veje til at udbedre problemerne. Dette skyldes, at en væsentlig del af de konstaterede problemer indtil videre må formodes at relatere sig til POLSAG-løsningens grundlæggende karakteristika:

- POLSAG-klienten er meget "snakkende" – Udførelsen af hver enkelt operation i POLSAG-klienten kræver relativt mange requests mellem web-browseren og serveren. Dette er en vigtig kilde til, at brugeren oplever systemet som langsomt under belastning. Selv hvis disse requests forløber optimalt relativt til svartiderne fra POLSAG-servermiljøet og transmissionstiden på netværket, skal der således ikke meget til, før systemet ikke opleves som tilstrækkeligt hurtigt af brugerne.
- POLSAG belaster serverinfrastrukturen hårdt – POLSAG-klienten lægger en væsentlig belastning på webserverne på grund af det store antal af requests. Webserverne belaster ligeledes databaseserveren tilsvarende hårdt.
- POLSAG-klientens operationer fører til mange databasekald – Der er konstateret en væsentlig belastning af den bagvedliggende database, hvilket blandt andet skyldes at almindelige operationer i klientens brugerflade udløser relativt mange kald mod databasen.
- POLSAG-klienten trækker data direkte fra randsystemer – I forbindelse med flere af operationerne udfører POLSAG-klienten (via POLSAG-servermiljøet) direkte kald til randsystemerne for at hente data. Det er i flere tilfælde konstateret, at randsystemerne svarer meget langsomt, hvilket er med til at skærpe oplevelsen af, at klienten er langsom.
- POLSAG-klienten er svær at optimere – Det er Globeteams vurdering at det vil være svært at udføre en gennemgribende optimering af POLSAG uden en større koderevision eller en egentlig omskrivning af koden. Denne vurdering er blandt andet baseret på at klienten dels bliver afviklet i JavaScript på en web-browser, at den grundlæggende opererer som en client/server-løsning samt at der forekommer interne kald mellem flere forskellige lag på serversiden. Dertil kommer, at JavaScript-delen af koden har en størrelsesorden og kompleksitet, der gør det vanskeligt og tidskrævende at sætte sig ind i.

I forbindelse med performance review'et er der dog blevet afdækket to helt oplagt forbedringsmuligheder. Den ene forbedring består i at sørge for at web browseren cacher websidens meget statiske ressourcer i en hel del dage, hvilket har potentialet til at reducere den nuværende belastning ved forsideload til omkring halvdelen. Den anden forbedring handler om at eliminere alle fejlene i relation til caching (http 304). Konsekvensen af dette vil være, at der sker en reduktion på ca. 20% af den http-kommunikation, som forekommer i performance review'et! Fejlen er tilsyneladende ikke relateret til POLSAG-specifik kode, men må være noget der hidrører fra Captia, eftersom det også ses i vores afestning hos Kriminalforsorgen.

Samlet set er Globeteam ikke i tvivl om, at der er muligheder for at forbedre POLSAG-løsningen ganske væsentligt, relativt til resultaterne fra uddannelsesmiljøet, for så vidt angår såvel performance, skalering og stabilitet. Vi vurderer at de største forbedringer går via ændringer i POLSAG-applikationen, hvilket formodes at kræve en væsentlig ressourceindsats (såvel i timer

som kalendertid) til både udvikling og slutaftestning. Globeteam henviser i øvrigt til BCG-rapporten og det senere kommende code review for yderligere afdækning af detaljerne omkring applikationen.

Leverandøren har gennemført en række tiltag i forbindelse med leverandørens arbejde med at bestå den kontraktuelle svartidsprøve i performancetestmiljøet. Leverandøren anfører, at disse tiltag samlet set indebærer en væsentlig forbedring i forhold til svartider, ydeevne og stabilitet. Af afgørende tiltag peger leverandøren primært på forbedring i servicetider gennem anden bestykning af servere, udvidet brug af indeksering og tuning af "tunge" SQL'er i databasen, samt reduktion i antallet af genererede requests gennem forbedret caching på klienterne og brug af anden sikkerhedsmodel (Kerberos).

Det er dog, som tidligere anført, forbundet med en del usikkerhed, hvad leverandøren har udført i performancetestmiljøet, da der kun i meget begrænset omfang foreligger dokumentation for leverandørens tiltag. Det har endvidere ikke været muligt konkret at besigtige miljø og testopstillingen, ligesom der hersker usikkerhed om den faktisk afledte effekt af tiltagene, idet Globeteam kun i et enkelt tilfælde har haft mulighed for at gennemføre en simpel måling i performancetestmiljøet, hvor tidsvinduet var begrænset til en pc og 3 arbejdstimer.

Det er således nødvendigt at der

1. Gennemføres en opfølgende performancetest i performancetestmiljøet eller andet miljø, hvori leverandørens ændringer er implementeret, med brug af den reviderede belastningsmodel.
2. At usikkerheden omkring de af leverandøren gennemførte tiltag og deres reproducerbarhed imødegås ved, at performanceforbedringerne kræves dokumenteret og overført til uddannelsesmiljøet, som herefter underlægges en opfølgende performancetest. Dette kan samtidig tjene som forberedelse til udrulning i produktionsmiljøet.

Forudsætninger for læsning af rapporten

For at få det fulde udbytte af rapporten forudsættes læseren at være i besiddelse af en grundlæggende datalogisk baggrund samt en generel viden om POLSAG-systemet, herunder de forskellige miljøer, der er etableret i forbindelse med POLSAG:

- Scanjourns udviklingsmiljø – ScanJourns interne udviklingsmiljø, som afvikles på en enkelt server og ikke på nogen områder ligner Rigspolitiets produktionsmiljø.
- Brugertestmiljø (IV3) – Rigspolitiets brugertestmiljø, som er bygget til funktionel test af POLSAG-løsningen. Brugertestmiljøet er således opbygget som en fuldt funktionelt ækvivalent kopi af produktionsmiljøet med henblik på brugeraftestning og indeholder også integration til randsystemerne samt en større og varieret mængde af sagsdata.
- Performancetestmiljø (AP3) – Rigspolitiets performancetestmiljø, som er opbygget med henblik på performanceaftestning af selve POLSAG-systemet. Der er lagt vægt på at dimensionere performancetestmiljøet som produktionsmiljøet (dog kun med 1/8 af produktionsmiljøets performance). AP3-miljøet indeholder ikke randsystemintegration, hvorfor en række af de beskrevne test cases ikke kan udføres på miljøet.
- Produktionsmiljø – Rigspolitiets driftsmiljø. Indtil videre er det kun Bornholms politikreds, der har ibrugtaget POLSAG.
- Uddannelsesmiljø (UDD) – Rigspolitiets uddannelsesmiljø er, som navnet antyder bygget til uddannelsesformål. Uddannelsesmiljøet afvikles på samme hardware som produktionsmiljøet og indeholder fuld randsystemintegration. De endelige målinger, der ligger til grund for performance review'et, er foretaget i uddannelsesmiljøet.

Der henvises til leverandørens dokumentation for evt. yderligere informationer omkring hardware, software og konfigurerings af POLSAG-systemet.

Ved læsningen af rapporten er det endvidere vigtigt at være opmærksom på, at det undervejs i forløbet generelt ikke har været muligt at udføre nogen yderligere undersøgelser af de underliggende årsager til de problemer i uddannelsesmiljøet, der er opsummeret i dette afsnit, og dermed hvor svære de er at udbedre. Dette gælder ikke mindst ydeevne og skalerbarhed. Det er vores formodning, at de konstaterede problemer er symptomer på mere grundlæggende problemstillinger, som i skrivende stund ikke er identificeret.

Årsagen til dette er, at uddannelsesmiljøet i mangt og meget tager form af en black box for Globeteam, eftersom Globeteam ikke har haft on-line adgang til servermiljøet, men driftsleverandøren blot har afleveret udvalgte logfiler for de tests der er blevet afviklet dagen forinden.

Det har således ikke været muligt for Globeteam at følge servernes reaktioner på testene og dermed tage evt. problemer i opløbet etc. Dette forhold har haft betydning for testens kvalitet og effektivitet, fordi det ikke var muligt at følge med i, hvornår systemet begynder at blive ustabil. Det har som sådan medført, at afestningerne i mange tilfælde har måttet udføres flere gange, fordi systemet gik helt eller delvist ned under afestningen, og i enkelte tilfælde har måttet opgives helt eller delvist.

Af samme årsag er der af og til blevet konstateret "forkerte" udsving på enkelte af testresultaterne samt mindre betydende overlap af tests (dvs. det viste sig efterfølgende at den tidligere test ikke var kørt helt færdig). Det har ikke altid været muligt at udbedre dette ved gentagelse af tests.

Uddybning af hovedkonklusioner og anbefalinger

Globeteam vurderer at hovedårsagen til de i uddannelsesmiljøet oplevede performanceproblemer med POLSAG er, at:

- Klienten genererer mange http requests – Antallet af http requests og antallet af klienter kan relativt hurtigt føre til forøgede svartider eller egentlig overbelastning af webserveren.
- Serveren genererer for mange databasekald – Antallet af kald til databasen fra webserverlaget stiller endog meget store krav til Oracle-databasen og kan som sådan føre til overbelastning af samme.
- Ved øgede datamængder vokser resourceforbruget tilsyneladende per behandlet række – Udover at en større datamængde typisk vil betyde, at der skal behandles flere rækker, indikerer afestningen, at behandlingstiden per række også vokser med datamængden. Et skalerende system bør kun forøge ressourceforbruget med antallet af transaktioner og størrelsen af transaktionerne, hvilket p.t. ikke ser ud til at være tilfældet.

De vigtigste observationer fra de gennemførte afestninger er:

Observation	Har konsekvenser for
Antallet af requests mellem klienten og webserveren	Brugerens svartider Belastningen af webserverne
Antallet af queries mellem webserveren og databasen	Belastningen på databasen Belastningen på webserverne

POLSAG-klienten trækker data direkte fra randsystemerne (via en proxy i POLSAG)	Brugerens svartider
Manglende optimering af database queries	Belastningen på databasen Brugerens svartider
Skrivninger belaster systemet meget hårdt	Belastningen på databasen Brugerens svartider
Døgnrapporter kan belaste systemet meget hårdt	Belastningen på databasen Brugerens svartider

I det følgende uddybes observationerne omkring http requests og databasekald.

For mange http requests

En overslagsberegning over antallet af requests i uddannelsesmiljøet angiver, at der mindst vil blive kørt 7.872 millioner requests pr. time i løbet af mandag formiddag (jvf belastningstesten afvikles der 656.000 requests pr. kredse pr. time, hvilket bliver til 7.872.000 requests for 12 kredse pr. time).

Dette betyder, at webserverne i uddannelsesmiljøet vil modtage 2.187 requests pr. sekund, hvilket medfører at hver webserver skal kunne håndtere 137 requests pr. sekund.

Resultaterne fra belastningstesten i uddannelsesmiljøet indikerer, at en kompleks server i gennemsnit kan levere 50 requests pr. sekund ved 100% belastning (tallet afhænger meget snævert af hvilke requests, der er tale om, hvorfor det må betragtes som en grov generalisering). Såfremt dette er korrekt vil der altså være behov for mindst 44 servere blot for at kunne imødekomme den beskrevne belastning.

Det er vigtigt at bemærke, at dette tal ikke tager højde for alle de operationer, der må formodes at blive udført i POLSAG, som ikke er dækket af de 13 test cases, der er indeholdt i belastningstesten.

Den indledende måling i performancetestmiljøet indikerer, at antallet af requests er reduceret. Det er dog fortsat vores vurdering, at antallet af requests giver anledning til bekymring, da det vurderes at ligge relativt højt – ikke mindst under indtryk af at applikationen skal afvikles henover Rigspolitiets WAN.

Det vurderes derfor at være afgørende at der bliver gennemført en opdateret test i et produktionslignende miljø med realistiske netværksforsinkelser.

For mange SQL queries

Den "løse" alt andet lige beregning af antallet af requests i uddannelsesmiljøet på 7.872.000 requests pr. time (12 kredse) vil betyde at Oracle skal eksekvere 212,5 millioner SQL-sætninger pr. time eller 59.040 SQL-sætninger pr. sekund, såfremt hvert http request i gennemsnit kræver 27 SQL-sætninger⁵.

⁵ Tallet er baseret på database trace i forbindelse med performancetesten (som består af 4.198 http requests, hvilket resulterer i intet mindre end 103.000 SQL-sætninger eller 183.000, når man medregner rekursive SQL-sætninger med). Disse tal vurderes umiddelbart at være i overensstemmelse med CSCs tal fra AP3, som indikerer at der bliver udført 30-40 "rene" databasekald pr. request (dvs. excl. logningen), hvilket er benyttet i beregningen.

Dette bliver til ca. 100.000 SQL-sætninger pr. sekund, hvis man medregner de interne SQL-kald. Dette er et virkelig stort tal nærmest uanset hvor stor en Oracle-installation der er tale om. Relativt til applikationen er det i øvrigt også et alt for stort tal, som det bør være teknisk muligt for leverandøren at reducere.

Givet erfaringerne fra den kontraktmæssige test i performancetestmiljøet og fra performance review'et skønner Globeteam, at Oracle-serveren vil skulle bestykses med mellem 60 og 128 CPU'er for at kunne levere denne ydelse. Det nuværende Oracle-miljø vil med andre ord være markant underdimensioneret. En eventuel tilføjelse af flere CPU'er vil indebære væsentligt forøgede omkostninger til såvel it-drift som licensvederlag til Oracle. Igen må det forventes at være teknisk muligt for leverandøren at reducere behovet for CPU'er, blandet andet ved at optimere de tungeste SQL-sætninger.

Det er vigtigt at bemærke at dette tal ikke tager højde for bidraget fra logning (som jævnt før CSCs tests vil mere end fordoble antallet af SQL-sætninger) og heller ikke tager højde for alle de operationer, der må formodes at blive udført i POLSAG, som ikke er dækket af de 13 test cases, der er indeholdt i belastningstesten.

Umiddelbare forbedringsmuligheder

Under aftestningen har Globeteam identificeret følgende umiddelbare forbedringsmuligheder på databasesiden:

- Det ser ud til at der kan laves betydelige optimeringer ved at kigge på opdateringsprocesser og minimere de nødvendige antal forespørgsler til databasen. Der er flere forespørgsler, der tager mellem 1-10 sekunder.
- Applikationen anvender ikke konsekvent bind variables, hvilket betyder dårligere skalering. Oracle database traces indeholder en stor andel såkaldte "Hard parse", hvilket betyder at parsningen af forespørgslen ikke bliver cachet. Dette betyder ligeledes dårligere skalering. "Hard parse"-problemet er i øvrigt allerede rapporteret til leverandøren.
- Der er set indikationer på, at databasen vil få problemer med at skalere ved øgede datamængder og brugere. Skaleringsevnen kan forbedres ved at optimere og tune de oftest forekommende samt de tungeste databasekald.

Globeteam er endvidere stødt ind i følgende umiddelbare forbedringsmuligheder på webserversiden:

- Konfigurering af længerevarende caching af websidens mest statiske ressourcer – Internet Explorer er i stand til at cache de mest statiske ressourcer og på den måde reducere båndbreddeforbruget på netværket samt webservernes belastning. Caching af disse ressourcer er i dag konfigureret til kun at vare i 24 timer (86.400 sekunder), hvilket bedømt på klienternes opførsel oven i købet ignoreres. Globeteam anbefaler, at opsætningen udbedres og tidsperioden forøges ganske dramatisk for at opnå de beskrevne fordele, som ikke mindst giver meget i relation til forsiden, hvor antallet af requests kan reduceres med ca. 50% (mængden af overførte data forventes endda at gå fra 5,5 Mb til ca. 2 Mb).
- Eliminering af http 304 (caching) – Det vil helt oplagt være en mærkbar fordel for POLSAG, hvis alle fejlene i relation til caching (http 304) bliver elimineret. Konsekvensen af dette vil

være, at der sker en reduktion på ca. 20% af den http-kommunikation, som genereres under belastningstesten!

- Sætte ind overfor systemets hukommelsesforbrug – Globeteam har i forbindelse med aftestningerne bemærket, at systemet tilsyneladende ikke frigiver RAM-hukommelsen, når først den er blevet allokeret. Dette kan blandt andet dække over memory leaks, som leder til risiko for ustabilitet og/eller forværret performance.
- Fjerne http 401-fejlene (reauthentication) – Det var allerede forud for performance review'et velkendt for leverandøren såvel som Rigspolitiet, at POLSAG er karakteriseret ved et stort antal genautentifikationer (http 401). Det anbefales at dette problem udbedres, således at denne kilde til belastning af webservere og netværket elimineres.
- Aftestning af http compression – Selvom IE6 har haft problemer med at tumle gzip, så vurderer vi umiddelbart, at det også kan være relevant at implementere http compression i POLSAG-installationen efter at have oplevet hos Kriminalforsorgen, hvilken besparelse det kan give på dataoverførslerne . Det er dog noget, der kræver en tilbundsgående aftestning, idet http compression indebærer en forøget belastning af webserverne.
- Tilpasning af Word-integrationskomponenten, så klienten kun kommunikerer med en server – POLSAGs Word-integration holder kontakt og forsøger at checke for nye versioner af dokumentintegrationen på alle de servere, den har haft kontakt med (disse servere står angivet under nøglen HKEY_CURRENT_USER\Software\Scan·Jour\Net server). Hvis klienterne har haft fat i flere servere vil den angiveligt kontakte alle disse servere for at sikre sig at den er i den nyeste version i forhold til alle serverne og derefter give besked til brugeren om at der findes en nyere version. Det er ikke undersøgt, hvor mange forbedringer en tilretning af denne funktionalitet vil bibringe.

Globeteam anbefaler i øvrigt at der sker en tilpasning af døgnrapporter-funktionaliteten (dvs. søgning samt tilføjelse og visning af døgnrapport), således at en brugere ikke uforvarende risikerer at lægge en væsentlig belastning på systemet, hvilket kan risikere at betyde at POLSAG bliver væsentligt sløvet eller ikke kan benyttes af Rigspolitiets organisation i en længere periode. Der findes flere veje til dette mål, fx:

- Isolere rapporteringsdata i et data warehouse
- Brugere får leveret rapporter asynkront (samt at man evt. er i stand til at abonnere på rapporter)
- Alle de gængse rapporter automatisk bliver genereret og ligger direkte tilgængelige for POLSAG-brugere
- Omfanget og fleksibilitet i bygningen af rapporter begrænses

Det kræver en detaljeret evaluering af forretningskravene og de tekniske muligheder at grave dybere i denne opgave.

Nødvendige yderligere aktiviteter

Det er som tidligere nævnt ikke praktisk muligt at udføre en "skrivebordsvurdering" af, hvor stort et potentiale de af leverandøren udførte tiltag i performancetestmiljøet reelt har for at forbedre POLSAG, når det kommer under belastning. På grund af manglen på data samt manglende muligheder for at afvikle den opbyggede test suite på performancetestmiljøet har Globeteam således ikke på nuværende tidspunkt nogen kvalificeret holdning til leverandørens anbringender.

Det er derfor under afsnittet "Muligheder for forbedringer" anbefalet, at der foretages en genkørsel af belastningstesten for at Rigspolitiets kan få et opdateret billede af POLSAGs nuværende status.

Det vil klart være at foretrække, hvis testen foretages på et opdateret uddannelsesmiljø, da dette vil gøre testene væsentligt mere sammenlignelige, ligesom det vil validere at det er muligt at overføre tiltagene i performancetestmiljøet til produktionsmiljøerne. Alternativt kan testen udføres i performancetestmiljøet, hvilket dog i sagens natur må ske under et forbehold for at de udførte tiltag kan og vil blive overført til produktionsmiljøerne samt at POLSAG i performancetestmiljøet giver et retvisende billede af løsningen, også i relation til de faktorer, som ikke er indeholdt i performancetestmiljøet.

Det vurderes at være vigtigt, at testmiljøet bestykses sådan at det som minimum kan sandsynliggøres gennem aftestning, at POLSAG besidder de lovede skaleringssegenskaber og er i stand til at håndtere Rigspolitiets behov.

Indhold	
Ledelsesresumé	2
Formål og afgrænsning af opgaven	2
Konklusion og anbefalinger	3
Performance	5
Skalerbarhed.....	6
Ustabilitet under pres.....	6
De gennemførte test afspejler "best case"	7
Muligheder for forbedringer	8
Forudsætninger for læsning af rapporten	9
Uddybning af hovedkonklusioner og anbefalinger	10
For mange http requests	11
For mange SQL queries.....	11
Umiddelbare forbedringsmuligheder	12
Nødvendige yderligere aktiviteter	13
Introduktion.....	23
Beskrivelse af de indledende målinger.....	24
Forudsætninger for læsning af rapporten	25
Baggrund for dokumentet	25
Analyse	27
Model	28
Terminologi.....	28
Metode	29
Beskrivelse af testløsningen	30
Beskrivelse af POLSAG Load Tool-applikationen	31
Tekniske detaljer omkring POLSAG Load Tool.....	34
Testforløbet	35
Forudsætninger	36
Risici.....	37
Evt. fravalg og udeståender	40
Introduktion til testene	41
Test cases, test case-planen og de opstillede modeller	41

De tre tests	42
Afviklingen af testene	43
Testmiljøet.....	45
Hurtigt overblik over de enkelte test cases.....	46
Test case 0: Åbn forside.....	46
Test case A: Grundregistrering	47
Test case B: Opret dokument	47
Test case C: Sagsplacering	48
Test case D: Sigtelse	48
Test case E: Sikret genstand	48
Test case F: Køretøj.....	49
Test case G: Søg	49
Test case H: Dan sagsstyring.....	49
Test case I: Afslut sag.....	50
Test case J: Træk døgnrapport	50
Test case K: Fremsøg pakkedokumenter.....	50
Observationer omkring forskellene mellem performance review-testen og den kontraktuelle test i AP3	51
Testresultater	53
Indledning.....	53
Generelle testsammenligninger	54
Baselining – Performancetests	57
Test: Performancetest mod 1 kompleks server	57
Test: Performancetest mod NLB	58
Belastningstests	59
Test: Belastningstest mod NLB	59
Analyse af anvendelse af randsystemer i de enkelte test cases	62
Belastningstest mod NLB – Test Case 0.....	66
Belastningstest mod NLB – Test Case A.....	67
Belastningstest mod NLB – Test Case B.....	68
Belastningstest mod NLB – Test Case C.....	69
Belastningstest mod NLB – Test Case D	70

Belastningstest mod NLB – Test Case E	71
Belastningstest mod NLB – Test Case F	72
Belastningstest mod NLB – Test Case G	73
Belastningstest mod NLB – Test Case H	74
Belastningstest mod NLB – Test Case I	75
Belastningstest mod NLB – Test Case J	76
Belastningstest mod NLB – Test Case K	77
Belastningstest mod NLB – Serverperformance	78
Belastningstest mod 1 simpel server	84
Belastningstest mod 1 simpel server – Test case 0	86
Belastningstest mod 1 simpel server – Test case A	87
Belastningstest mod 1 simpel server – Test case B	88
Belastningstest mod 1 simpel server – Test case C	89
Belastningstest mod 1 simpel server – Test case D	90
Belastningstest mod 1 simpel server – Test case E	91
Belastningstest mod 1 simpel server – Test case F	92
Belastningstest mod 1 simpel server – Test case G	93
Belastningstest mod 1 simpel server – Test case H	94
Belastningstest mod 1 simpel server – Test case I	95
Belastningstest mod 1 simpel server – Test case J	96
Belastningstest mod 1 simpel server – Test case K	97
Belastningstest mod 1 simpel server – Server performance	98
Belastningstest mod 1 kompleks server	100
Belastningstest mod 1 kompleks server – Test case 0	101
Belastningstest mod 1 kompleks server – Test case A	101
Belastningstest mod 1 kompleks server – Test case B	102
Belastningstest mod 1 kompleks server – Test case C	102
Belastningstest mod 1 kompleks server – Test case D	103
Belastningstest mod 1 kompleks server – Test case E	103
Belastningstest mod 1 kompleks server – Test case F	103
Belastningstest mod 1 kompleks server – Test case G	104
Belastningstest mod 1 kompleks server – Test case H	104

Belastningstest mod 1 kompleks server – Test case I	105
Belastningstest mod 1 kompleks server – Test case K	105
Belastningstest mod 1 kompleks server – Server performance.....	106
Belastningstests - Problemløst	107
Døgnrapporter mod NLB	109
8.00-8.15 tests	116
8.00-8.15 test mod 1 simpel server.....	117
8.00-8.15 test mod 1 kompleks server	118
8.00-8.15 test mod NLB	122
Stresstests.....	127
Stresstest med test case 0 (1 til 8 Tråde) mod NLB.....	127
Stresstest med Case 0 (9 og 10 Tråde) mod NLB.....	129
Stresstest med Case A (4 til 7 Tråde) mod NLB	131
Skalerbarhedstest	135
8.00 - 8.15 test mod 1, 2, 3 og 4 komplekse server.....	135
Belastningstest mod 2 til 4 komplekse servere	141
Belastningstest mod 2-4 komplekse servere – Test case 0	143
Belastningstest mod 2-4 komplekse servere – Test case A.....	144
Belastningstest mod 2-4 komplekse servere – Test case B	145
Belastningstest mod 2-4 komplekse servere – Test case C.....	146
Belastningstest mod 2-4 komplekse servere – Test case D.....	147
Belastningstest mod 2-4 komplekse servere – Test case E	147
Belastningstest mod 2-4 komplekse servere – Test case F	148
Belastningstest mod 2-4 komplekse servere – Test case G.....	149
Belastningstest mod 2-4 komplekse servere – Test case H.....	150
Belastningstest mod 2-4 komplekse servere – Test case I	151
Belastningstest mod 2-4 komplekse servere – Test case J.....	151
Belastningstest mod 2-4 komplekse servere – Test case K.....	152
Forøget datamængde-test	154
Andre tests og resultater	156
Databaseanalyse.....	156
Trace fra 23/9 2011	156

Trace fra 6/10 2011	158
Massivt antal databaseforespørgsler	159
Uddybende information om udvalgte databaseforespørgsler.....	160
Test hos Kriminalforsorgen.....	169
Captia-miljøet	169
Testforløbet	169
Observationer	170
Andre tests.....	171
Andre observationer.....	172
Test af web browsere	173
Evaluering af JavaScript	174
Udfordringer med http caching (304)	174
Uddybende information om de enkelte http requests	176
Klassedeling af service request.....	176
Kommentarer til antallet af requests	176
Kald til BPHandler	177
Fordeling af de forskellige test cases.....	178
Case 'O' Load af forside.....	178
Case 'A' Grundregistrering	178
Case 'B' Opret dokument	179
Case 'C' Ny sagsplacering.....	179
Case 'D' Sigtelse	179
Case 'E' Sikret genstand.....	180
Case 'F' Køretøj.....	180
Case 'G' Søg	180
Case 'H' Dan sagsstyring	181
Case 'I' Afslut sag	181
Case 'J' Træk døgnrapport.....	181
Case 'K' Fremsøg dokument pakke.....	182
Tidsforbrug for de enkelte requests.....	182
Request under 300 ms.....	182
Request over 300 ms	183

Afviklingstid fordelt på requestklasser	183
Test i produktionsmiljøet	189
Stresstest med test case 0 (10 tråde) på produktionsmiljøet	190
Stresstest med test case 0 og J på produktionsmiljøet (uden sager)	192
8.00-8.15 stresstest med test case 0 og J på produktionsmiljøet med 10 klienter	195
8.00-8.15 stresstest med test case 0 og J på produktionsmiljøet: 10, 20 og 30 klienter	198
Appendix A. De 12 test cases.....	209
0 Opstart af klienten	210
Arbejdsgang	210
Skærbilleder	210
A Grundregistrering	211
Arbejdsgang	211
Skærbilleder	212
B Opret dokument	221
Arbejdsgang	221
Skærbilleder	221
C Ny sagsplacering	241
Arbejdsgang	241
Skærbilleder	241
D Sigtelse	256
Arbejdsgang	256
Skærbilleder	256
E Sikret genstand	276
Arbejdsgang	276
Skærbilleder	276
F Køretøj	301
Arbejdsgang	301
Skærbilleder	301
G Søg	325
Arbejdsgang	325
Skærbilleder	325
H Dan sagsstyring	330

Arbejdsgang.....	330
Skærbilleder	330
I afslut sag.....	357
Arbejdsgang.....	357
Skærbilleder	357
J Træk døgnrapport	391
Arbejdsgang.....	391
Skærbilleder	391
K Fremsøg pakke dokumenter.....	400
Arbejdsgang.....	400
Skærbilleder	400
L Hent bødesag.....	432
Arbejdsgang.....	432
Skærbilleder	432
Appendix B. Test case-plan.....	433
Appendix C. Beskrivelse af testene, der indgår i rapporten	435
Performancetesten.....	435
Belastningstesten	436
Stresstesten	439
Rampup testen	440
8.00-8.15 testen	441
Døgnrapporter.....	443
Appendix D. Opsamling og forklaring på fejl oplevet på POLSAG under aftestningerne.....	444
Kategori 1 fejl.....	444
Oracle sessionsfejl	444
Fejltet hovedoplysning:file_key er ikke udfyldt	444
Oracle loginfejl.....	444
Scanjour intern fejl	444
Feltet nameKey er ikke udfyldt.....	445
Låsningsfejl ved sagsoprettelser.....	445
Forbindelse til databasearkivet er nede.....	445
Fejl fra arkivdriver.....	445

Systemkey findes ikke i registeret	445
Alle forespørgsler mod POLSAG giver HTTP 503 (System Unavailable) statuskode i response	445
Fejl i døgnrapporterne under load	446
Kategori 2 fejl.....	447
Kategori 3 fejl.....	447
Personen behandles i en anden sagsstyringssag.....	447
Appendix E. Supplerende information	448
Revurdering af belastningsmodel.....	448
Opsummering	448
Ændringer til belastningsmodellen.....	448
AP3-observationer	449
Opsummering	449
Overordnede konklusioner	450
Afgrænsninger og risici	451
Testforløbet	451
Resultater	452
Test case-dokumentation	453
Testcase 0	453
Testcase A.....	453
Testcase B.....	454
Testcase C.....	454
Testcase D.....	454
Testcase E	454
Testcase F	454
Testcase G.....	454
Testcase H.....	454
Testcase I	455
Testcase J.....	455
Testcase K	455
Leverandørens påstande fra teknisk underspor.....	455
Opsamling på afrapportering fra svartidsprøve	456

Introduktion

Formålet med performance review'et er at gennemføre en dybdegående analyse af POLSAGs ydeevne og skalerbarhed.

Performance review'et var som udgangspunkt defineret til at indeholde:

- *Etablering af et testmiljø og en metode til at foretage kvantitative performancemålinger på POLSAG. Med kvantitative målinger dokumenteres, hvordan systemet opfører sig under højere belastning. Eftersom målingerne afvikles i et testmiljø vil svartider ikke være repræsentative for driftsmiljøet.*
- *Dokumentation af POLSAGs performancekarakteristika baseret på mellem 8 og 10 anvendesscenarier som er ofte anvendt i systemet, herunder afgive anbefalinger for opdateringer, der vil forbedre den af brugeren oplevede performance. Scenarierne forventes at dække mellem 25 og 30 skærbilleder.*
- *Afdækning af belastning af eksterne systemer for de højfrekvente scenarier.*
- *Udførelse af en load/stress-test i samarbejde med CSC/Scanjour, der skal afdække systemets opførsel under hård belastning.*

Hele opgaveforløbet i performance reviewet er blevet afviklet på en iterativ facon, hvorfor der løbende er sket kurs- og indholdsjusteringer i takt med fremkomsten af forskellige informationer og data samt de deraf følgende erkendelser.

Det var eksempelvis oprindelig forventningen at der skulle bygges et fuldt testmiljø, der indbefattede adgang til kildekoden for at det ville være muligt at udføre eksperimentelle ændringer i det omfang det måtte være relevant for at underbygge de afgivne anbefalinger. Med direkte adgang til kildekoden ville det være muligt med det samme at afdække, hvordan observationerne af, hvad der sker ude ved brugeren på "ydersiden af systemet" hænger sammen med de reelle kodelinjer og dermed blandt andet med sikkerhed være i stand til at stedfæste den underliggende årsag til de forskellige observationer samt ikke mindst, hvad der er umiddelbart muligt og hvad der ikke er.

Denne metodik måtte dog opgives efter det stod klart for Rigspolitiet at det ville tage mindst 6 uger alene at opbygge et sådant miljø hos CSC.

Derudover er der sket en væsentlig udvidelse af antallet af test cases og tests, ligesom performance review'et har gennemført aftestninger på flere forskellige miljøer. Til gengæld blev omfanget af afdækningen af belastningen af eksterne systemer nedprioriteret.

I forbindelse med foranalysen blev der foretaget målinger i følgende miljøer:

Scanjourns udviklingsmiljø	Foranalysen startede med at afprøve forskellige målemetoder i et udviklingsmiljø, som ScanJour stillede til rådighed. Udviklingsmiljøet er karakteriseret ved at det højst er brugbart i forbindelse med opbygningen af målinger samt validering af disse, da det ikke på nogen områder ligner Rigspolitiets produktionsmiljø.
IV3 (brugertestmiljø)	IV3-miljøet er karakteriseret ved følgende: <ul style="list-style-type: none">• Produktionslignende datamængder• Integration med randsystemer• Ikke bygget med henblik på performance (men

	derimod funktionel test) <ul style="list-style-type: none"> • Under foranalysen var den konfigureret med en ældre version af POLSAG
AP3 (performancetestmiljø)	AP3-miljøet er bygget som en 1/8 af produktionsmiljøet og er karakteriseret ved <ul style="list-style-type: none"> • Ikke at have integration til randsystemerne • Kun at indeholder et meget behersket antal sager • Have et andet teknisk setup end produktion (f.eks. anden strategi for load balancing og en anderledes opsætning mod Active Directory)

Det blev imidlertid konkluderet, at ingen af disse miljøer giver tilstrækkelig sikkerhed for analyseresultaterne er retvisende relativt til produktionsmiljøet, hvorfor det blev besluttet at henlægge performance review-aftestningerne til uddannelsesmiljøet.

Beskrivelse af de indledende målinger

Globeteam har foretaget indledende performancemålinger i følgende POLSAG-miljøer for at få afklaret, hvilket af de eksisterende miljøer, performance review'et skulle henlægges til:

- IV3-miljøet
- AP3-miljøet
- Produktionsmiljøet (læsemålinger, hvor Rigspolitiet førte kontrollen med systemet)

På IV3-miljøet blev Rigspolitiets test cases gennemført pr. håndkraft (dvs. via manuel indtastning i browseren). I forbindelse med målingerne blev der indsamlet informationer med følgende værktøjer:

- Fiddler traces af al http-trafik mellem klient og server.
- IIS logs, der viser hvilke forespørgsler, der ramte serveren under testene.
- DotTrace-målinger på serveren, der viser detaljer omkring hvad systemet har lavet inde i koden.
- Performance counters på serveren, der viser, hvordan serveren har været belastet, rent hardwaremæssigt.

Der blev testet på IV3-miljøet den 27. og 28. juli.

På AP3-miljøet blev Rigspolitiets test cases også gennemført pr. håndkraft og tider blev sammenlignet med tiderne fra IV3-miljøet. Der blev indsamlet informationer under målingerne med følgende værktøjer:

- Fiddler traces af http-trafik mellem klient og server.
- IIS logs, der viser hvilke forespørgsler der ramte serveren under testene.

En række test cases viste sig ikke at kunne afvikles på AP3 på grund af den manglende integration til randsystemer (dvs. blandt andet index2 og KR). På AP3-miljøet blev der testet både med og uden Network Load Balancing. Dvs. både hvor en enkelt server blev ramt samt hvor loadet blev fordelt ud på flere servere.

Der blev testet på AP3-miljøet den 2. og 4. august.

Målingerne på produktionsmiljøet var ikke direkte sammenlignelige med de øvrige aftenninger, idet der kun var adgang til at køre et begrænset udpluk af dem på produktionsdata. Der blev indsamlet informationer under målingerne med følgende værktøjer:

- Fiddler traces af http-trafik mellem klient og server.

Der blev testet på produktionsmiljøet den 15. og 16. august.

Resultaterne fra produktionsmiljøet har efterfølgende indgået i sparring mellem Globeteam og Scanjour omkring bl.a.

- Særligt langsomme http-forespørgsler mellem klient og server.
- Mange logins mellem klient og server.
- Mange cachede forespørgsler (http 304-problemet), der bruger meget tid.

På baggrund af de indsamlede informationer samt måleresultater fra de forskellige miljøer blev der i dialog med CSC besluttet at performance review-aftestningerne gennemføres i uddannelsesmiljøet. Den primære årsag til at valget faldt på uddannelsesmiljøet er at det rummer randsystemintegration samt at CSC har påpeget at det er det miljø, der ligner produktionsmiljøet mest.

Forudsætninger for læsning af rapporten

For at få det fulde udbytte af rapporten forudsættes læseren at være i besiddelse af en grundlæggende datalogisk baggrund samt en generel viden om POLSAG-systemet, herunder de forskellige miljøer, der er etableret i forbindelse med POLSAG:

- Scanjourns udviklingsmiljø – ScanJourns interne udviklingsmiljø, som afvikles på en enkelt server og ikke på nogen områder ligner Rigspolitiets produktionsmiljø.
- Brugertestmiljø (IV3) – Rigspolitiets brugertestmiljø, som er bygget til funktionel test af POLSAG-løsningen. Brugertestmiljøet er således opbygget som en fuldt funktionelt ækvivalent kopi af produktionsmiljøet med henblik på brugeraftestning og indeholder også integration til randsystemerne samt en større og varieret mængde af sagsdata.
- Performancetestmiljø (AP3) – Rigspolitiets performancetestmiljø, som er opbygget med henblik på performanceaftestning af selve POLSAG-systemet. Der er lagt vægt på at dimensionere performancetestmiljøet som produktionsmiljøet (dog kun med 1/8 af produktionsmiljøets performance). AP3-miljøet indeholder ikke randsystemintegration, hvorfor en række af de ønskes test cases ikke kan udføres på miljøet.
- Produktionsmiljø – Rigspolitiets driftsmiljø. Indtil videre er det kun Bornholms politikreds, der har ibrugtaget POLSAG.
- Uddannelsesmiljø (UDD) – Rigspolitiets uddannelsesmiljø er som navnet antyder bygget til uddannelsesformål. Uddannelsesmiljøet afvikles på samme hardware som produktionsmiljøet og indeholder fuld randsystemintegration. De endelige målinger, der ligger til grund for performance review'et, er foretaget i uddannelsesmiljøet.

Der henvises til leverandørens dokumentation for evt. yderligere informationer omkring hardware, software og konfiguration af POLSAG-systemet.

Baggrund for dokumentet

Dette dokumentet har til formål at konsolidere alle de essentielle informationer, data, delkonklusioner og overordnede konklusioner fra performance review'et. Det udgør som sådan den endelige afrapportering for selve performanceanalysen.

Performance review'et har endvidere genereret en række detaljerede målinger og en del forløbsdokumentation, som ikke er en del af rapporten.

Efter aftestningerne blev afsluttet har Rigspolitiet opdateret den belastningsmodel, der ligger til grund for hovedparten af aftestningerne. Læseren skal således være klar over at målingerne i rapporten er baseret på en højere belastning end det som i dag forventes at være "travleste time". Det henvises til ledelsesresumeeet for en umiddelbar vurdering af konsekvenserne.

Rapportens målinger er baseret på de miljøer som Globeteam har fået stillet til rådighed til aftestning i perioden (uddannelsesmiljøet og produktionsmiljøet) og begrænser sig derfor generelt til at udtale sig om de faktuelle forhold, der er konstateret for POLSAG i de benævnte miljøer. Dette skal også ses i lyset af at Rigspolitiets primære behov bestod i at få undersøgt om POLSAG i den leverede version kunne rulles videre til Midt og Vestjylland.

De ændringer som leverandøren har foretaget i performancetestmiljøet AP3 er udelukkende indarbejdet i Appendix E og Ledelsesresumeeet.

Analyse

Performance review'et har undersøgt følgende:

- POLSAG performance – POLSAG-løsningens ydeevne belyses fra forskellige vinkler ved hjælp af nogle virkelighedsnære tests, der er baseret på et antal repræsentative test cases udarbejdet af Rigspolitiet.
 - Performance – Undersøgelse af POLSAG-løsningens performance relativt til de forskellige test cases.
 - Belastning – Undersøgelse af den forventede ydeevne i POLSAG er baseret på de opstillede forventninger til kredsenes behov. Måling af ydeevne er baseret på to tests, som simulerer belastningen en mandag i en typisk kreds.
 - Stress – Undersøgelse af hvordan POLSAG reagerer, når det bliver sat under pres. Denne test tester først og fremmest hvordan systemet håndterer høje spidsbelastninger, men afdækker samtidig også vigtige informationer i relation til POLSAGs skaleringssevne.
 - Skalbarhed – Undersøgelse af, hvilke muligheder der findes for at forøge POLSAGs performance ved at tilføje yderligere webservere. Dvs. hvordan er POLSAG i stand til at skalere med antallet af webservere.
- Ovenstående undersøgelser af ydeevnen vil samtidig levere noget information om hvor stor en spidsbelastning POLSAG forventes at kunne klare. Der bliver dog ikke udført en udtømmende undersøgelse af systemets ydeevne.
- Performancemæssige konsekvenser af tilvækst i sagsdatabasen – Undersøgelse af, hvordan POLSAG ser ud til at reagere på den løbende forøgelse af antallet af sager og sagsdata i databasen, som vil opstå hen over tid.
- POLSAG performancekarakteristika relativt til en anden Captia-installation – Denne test har primært til formål at undersøge om POLSAGs performancekarakteristika modsvarer det man oplever i andre Captia-installationer baseret på et behersket antal aftestninger. Testen formodes samtidig at kunne bidrage til forståelsen af, hvor meget eller hvor lidt POLSAG ligner Captia med henblik på at substantiere formodninger omkring hvordan eventuelle performanceforbedringer af standard-Captia vil influere på POLSAG.

I forbindelse med performanceaftestningerne skete der også en indledende undersøgelse af hvilken indvirkning et skift af web-browser versionen kunne have, idet dette i forløbet blev fremhævet af POLSAG-leverandøren som en helt åbenbar og markant mulighed for at forbedre systemets brugeroplevelse. Undersøgelsen bestod i at Globeteam udførte en indledende aftestning af de performancemæssige konsekvenser af en opgradering fra den nuværende IE6-browser til IE8 eller en forbedret JavaScript-fortolker.

POLSAG-leverandøren har i øvrigt senere fremhævet at en opgradering fra Windows Server 2003 til Windows Server 2008 R2 også ville give en markant forbedring af systemets brugeroplevelse. Dette er dog ikke blevet undersøgt af Globeteam. Dette skete blandt andet under indtryk af at den version af ScanJour Captia som POLSAG p.t. er baseret på ikke understøtter Windows Server 2008 R2 og leverandøren selv tog aktive skridt til at undersøge denne formodning. Givet vores nuværende viden om POLSAG-systemet deler Globeteam i øvrigt ikke forhåbningen om at et skift af serverstyresystem vil medføre en mærkbare eller afgørende ændring af systemets brugeroplevelse.

Model

I forbindelse med performance review'et har Rigspolitiet defineret følgende 12 funktionelle test cases:

Test case navn
A Grundregistrering
B Opret dokument
C Ny sagsplacering
D Sigtelse
E Sikret genstand
F Køretøj
G Søg
H Dan sagsstyring
I afslut sag
J Træk døgnrapport
K Fremsøg pakke dokumenter
L Hent bødesag (manuel test case)

Dertil kommer basistesten "0 Initial load", som består i opstarten af POLSAG-systemet i web-browseren, da Rigspolitiet har konstateret at denne operation kan tage lang tid.

De 12 funktionelle test cases og basistesten er beskrevet detaljeret i Appendix A.

De funktionelle test cases er udvalgt baseret på at det skal være en test case, der vurderes som værende en del af systemets kritiske vej (dvs. noget som bliver udført ofte eller noget, der er vitalt for Politiets arbejde og forventede almindelige anvendelse af systemet). Derudover er der tilføjet en test, der angiver systemets opstartstider (dvs. adgangen til startbilledet), da dette også vurderes at være vigtigt for brugeroplevelsen og med baggrund i feltstudier ofte viser sig at forekomme flere gange om dagen hos hver medarbejdere.

Disse 12 funktionelle test cases danner grundlag for de forskellige tests, der afvikles som en del af performance review'et. Baggrunden for testene, som performance review'et har været fokuseret mod at simulere, er den belastning som man forventes at opleve i en typisk politikreds om mandagen. Dette er beskrevet i nærmere detaljer i Appendix B.

Det går således som en rød tråd gennem de tests, der indgår i performance review'et, at det tilstræber at lægge en belastning på systemet, som giver et realistisk billede af, hvordan POLSAG vil blive belastet mandag formiddag i en typisk politikreds. Der er altså ikke tale om en kontraktmæssig test eller en teoretisk test, men derimod om en test, der har til formål at ramme så tæt som teknisk muligt på Rigspolitiets konservative forventninger i relation til den almindelige og påkrævede belastning på POLSAG-systemet relativt til de 13 test cases.

Med det sagt er det vigtigt at understrege at modellen er afgrænset til de 13 test cases, som kun repræsenterer en del af aktiviteterne i systemet. Testene dækker således ikke den øvrige belastning som POLSAG ventes at blive udsat for i den virkelige verden.

Terminologi

Rapporten opererer med følgende termer.

Modeller og test cases

- Test cases: Alle tests er bygget op omkring 13 test cases, som er defineret i appendix A.
- Test case-plan – Rigspolitiet har udarbejdet en plan, som man vurderer, er repræsentativ for den belastning på POLSAG der forventes at være i løbet af en typisk mandag i hver politikreds i relation til hver af de 13 test cases. Test case-planen er blandt andet baseret på erfaringerne fra POLSAS-systemet.
- Belastningsmodel – Baseret på test case-planen er der opstillet et mål for, hvor meget POLSAG skal kunne levere af ydelse på en time indenfor hver af de 13 test cases. Modellen er baseret på den belastning af POLSAG, som test case-planen angiver for mandag formiddag i en typisk kreds.
- 8.00-8.15 model – Baseret på test case-planen er der opstillet et mål for, hvor meget POLSAG skal kunne levere af ydelse i det første kvarter indenfor hver af de 13 test cases. Modellen er baseret på den belastning af POLSAG, som test case-planen angiver for mandag morgen i en typisk kreds.

Egentlige tests (såvel manuelle og automatiske)

- Performancetest – Består af de 12 test cases, som er automatiseret. Den sidste test case (bødesager) udføres manuelt, da denne involverer ældre systemer, der ikke med rimelig indsats kan automatiseres.
- Belastningstest – Består af et nærmere defineret antal kørsler af hver af de 12 automatiserede test cases henover en nærmere defineret tidsperiode. Antallet af kørsler er baseret på en "syntetisering" af Rigspolitiets estimater på belastningen, som POLSAG-systemet vil være udsat for i løbet af en time mandag formiddag i en typisk kreds jævnfør test case-planen.
- 8-8.15 testen – Består af 200 kørsler af test case 0 (forside) og 100 kørsler af test case J (døgnrapport), som skal afvikles henover 15 minutter. Antallet af kørsler er baseret på Rigspolitiets estimater på belastningen som POLSAG-systemet vil være udsat for i løbet af de første 15 minutter af arbejdsdagen mandag formiddag i en typisk kreds jævnfør test case-planen. Det oversættes til opstart af POLSAG-systemet på 200 pc'er og visning af 100 døgnrapporter.

Metode

I forbindelse med foranalysen blev der udført forskellige indledende undersøgelser af mulighederne for at udføre en retvisende aftestning af POLSAGs ydeevne.

På baggrund af forskellige forsøg med klienten (Internet Explorer 6) var det Globeteams oplevelse at:

- Systemets klientside er skrøbelig – Hvis man fx taster for tidligt, så risikerer man at web-browser'en ikke svarer eller at der opstår fejlen på indtastningsbillederne.
- Systemets klientside er udviklet med mange synkrone kald, hvilket medfører en del venten i systemet, hvor brugeren er henvist til at vente på at systemet svarer.
- Det er generelt ikke et hurtigt system og det varierer en del i svartider på tværs af de forskellige miljøer baseret på sammenligninger af tiderne fra de indledende aftestninger i de forskellige miljøer.

Globeteams konkluderede på den baggrund at det ikke ville være praktisk muligt at udføre performancetest ved hjælp af automatisering helt fra klienten (Internet Explorer) uden en meget stor og konstant ressourceindsats til tuning af testene, ligesom det ville indebære et væsentligt merarbejde at flytte testene på tværs af miljøerne. Ydermere var det vores opfattelse at der

eksisterede en risiko for at klienten ville vise sig at opføre sig teknisk ustabil, og der derved ikke ville kunne opnås en tilstrækkelig sikkerhed for at testene ville blive håndterbare og retvisende. Såvidt Globeteam forstår stemmer vores vurderinger overens med de erfaringer, der er blevet indhøstet, i forbindelse med leverandørernes udvikling af de QTP-baserede tests (som netop bliver afviklet helt ude fra klienten). Disse tests har således givet mange bryderier undervejs.

I stedet blev det besluttet at Globeteam ville udføre performancetesten baseret på afvikling af optagede request direkte mod POLSAG-serveren fra en testapplikation med løbende opsamling af alle requests og responses. Globeteam har bemærket at denne fremgangsmåde siden hen også er blevet adopteret af Scanjour i forbindelse med deres egne aftestninger.

Det var kombinationen af følgende nedenstående behov som førte til at Globeteam byggede en load test-applikation i stedet for at benytte en off-the-shelf applikation:

- For at kunne foretage den korrekte autentificering af klienten.
- For at kunne substituere værdier i html'en dynamisk. Dette benyttes blandt andet til at opsamle de nødvendige systemnøgler, der bliver genereret af Captia, og benytte disse senere i testen.
- For at kunne oprette sagerne på forhånd og henvise til disse i forbindelse med testen. Klienten laver således selv sit "arbejdsområde" inden den påbegynder testene for at testene, således at disse målinger ikke belastes med etableringen af de sager, der anvendes i testen.
- For nemt og hurtigt at kunne imødekomme de unikke behov for styring af forskellige detaljer i relation til testene, der opstod i projektførelsen.
- For at sikre at Rigspolitiet i fremtiden har mulighed for at genkøre testene op mod POLSAG-miljøet samt evt. genbruge løsningen i andre miljøer og til andre tests.
- For at kunne levere en konfigurerbar model til afvikling af tests, der blandt andet kan simulere
 - Mange samtidige brugere
 - Gentagelser af tests
 - Ventetider mellem tests
 - Simulering af brugerventetider i tests

Brugt i kombination med Windows Server's indbyggede funktionaliteter Performance Counters og IIS-logning vurderes dette at være fuldt tilstrækkeligt til at afteste, dokumentere og evaluere POLSAG ydeevne samt dens opførelse under load.

Beskrivelse af testløsningen

Alle tests er som sagt blevet afviklet fra en Windows-applikation ved navn POLSAG Load Tool, der er skræddersyet til de specielle forhold i POLSAG-aftestningen.

Applikationen anvender programbiblioteker udstillet af Fiddler-værktøjet (se <http://www.fiddler2.com/fiddler2/>), som er et anerkendt værktøj til web debugging på Windows-området.

En af de vigtigste pointer i forbindelse med brugen af Fiddler er at det er i stand til at optage al kommunikation mellem klienten og webserveren. Dette betyder at der efterfølgende bliver genereret en fuldstændig dokumentation af hvilke tests, der er udført helt ned til hvert enkelt http

request således at der aldrig vil kunne opstå berettiget tvivl efterfølgende om en given test er korrekt afviklet eller ej. Dette har i øvrigt vist sig særdeles værdifuldt i relation til POLSAG, idet det hurtigt har kunne afdækkes, hvis hele eller dele af POLSAG-servermiljøet er holdt op med at svare korrekt under en test og resultaterne således ikke er valide.

Netop det forhold at Fiddler bliver benyttet indebærer en væsentlig lettelse af arbejdet med at sikre at testen er fuldstændig korrekt relativt til det, der sker på web-browseren, når brugeren udfører tingene i virkeligheden. Det er jo altafgørende for testens validitet at alle http requests er korrekte relativt til, hvordan den pågældende web browser (i POLSAGs tilfælde IE6) opfører sig. Med Fiddler har vi direkte adgang til at sammenligne hvert eneste request i test case'n med "originalen", der blev optaget ved at gennemføre den pågældende test case manuelt i web-browseren.

Det betyder samtidig også at det er væsentligt lettere at genindspille nye eller reviderede test cases samt ikke mindst tilsvarende test cases i andre miljøer, såfremt dette bliver relevant. Dette har undervejs i testforløbet var udnyttet til at gennemføre revisioner af testene på baggrund af Rigspolitiets input til de 12 test cases.

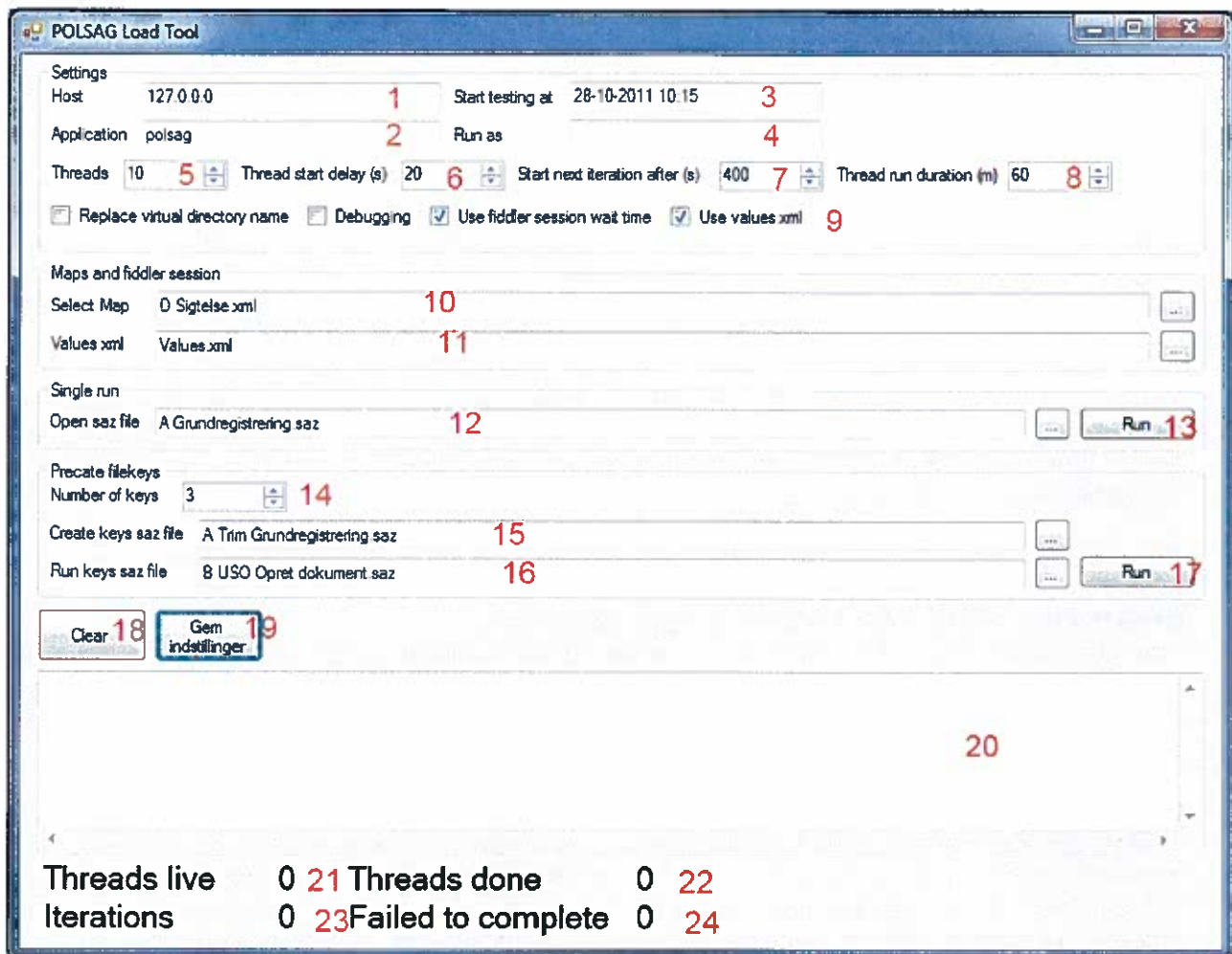
Det er samtidig vigtigt at forstå at Fiddler udelukkende bliver brugt til at optage requests.

Beskrivelse af POLSAG Load Tool-applikationen

Eftersom Fiddler "kun" er beregnet til at studere og evt. manipulere http requests rummer det ikke nogen muligheder for at benytte det til at sende flere requests samtidig, dvs. udføre egentlig load testing, ligesom det heller ikke "out of the box" kan håndtere de variable, som er nødvendige for at udføre de ønskede tests i POLSAG regi.

Denne begrænsning er udbedret ved hjælp af POLSAG Load Tool, som er et specialudviklet værktøj der læser de optagelser, der er optaget med Fiddler og kan manipulere og gensende disse mod webserveren uden brug af browserklienten. POLSAG Load Tool gør det muligt at køre de testscenarier, der er optaget med Fiddler fra mange klienter, der hver simulerer flere samtidige brugere, på samme tid for at simulere det ønskede antal klienter. Eftersom Rigspolitiet anvender Windows XP, som er begrænset til 10 udgående forbindelser, er POLSAG Load Tool konfigureret til at køre 10 tråde på hver pc-klient, hvilket således betyder at hver pc-klient⁶ er i stand til at simulere samme belastning som 10 brugere.

⁶ Hver instans af POLSAG Load Tool kører under samme brugerkonto. Dvs. alle tråde vil komme fra samme bruger set fra POLSAG-miljøet synsvinkel. Eftersom Politiet forudsætter at der vil være mange eksempler på at brugerne åbner flere sessioner mod POLSAG betragtes det som en korrekt og retvisende brug af POLSAG, at testen åbner op til ti samtidige sessioner med samme bruger.



Ovenstående skærmgrab fra POLSAG Load Tool'et er tilføjet med tal ud for værktøjets funktioner:

- 1) Angiver hvilken server den pågældende instans af værktøjet er konfigureret til at sende dets HTTP-forespørgsler til.
- 2) Angiver det virtuelle bibliotek på webserveren der kommunikerer med, feks. rplSAGu
- 3) Testens starttidspunkt. Anvendes til at koordinere starttidspunkt af kørslen (evt. forhåndsoprettelser starter straks) på tværs af flere testklienter. Klienten kan startes inden starttidspunktet og den vil derefter afvente det pågældende tidspunkt inden det påbegynder aftestningen.
- 4) Angiver navnet på den sagsbehandler, der simuleres i kørsler mod systemet. Dvs. alle testkørsler på pc'en sker med denne brugerkonto.
- 5) Antallet af tråde der startes op. Hver tråd vil køre den konfigurerede Fiddler-optagelse.
- 6) Angiver en forsinkelsen i opstarten af den næste tråd. Anvendes til at simulere at alle trådene ikke starter samtidig. I dette eksempel starter den første tråd med det samme og de næste 9 startes hver gang, der er gået 20 sekunder
- 7) Angiver et interval i sekunder, som hver iteration af den pågældende Fiddler-optagelse forventes at tage relativt til den pågældende test case. I dette tilfælde vil værktøjet vente frem til der er gået 400 sekunder, såfremt værktøjet er i stand til at køre hele optagelsen igennem inden de 400 sekunder er gået. Indstillingen bruges til at sikre værktøjet ikke kører

- "for hurtigt" ift. Rigspolitiet måltal for belastningsmodellen. Der startes ikke en ny test på en tråd før den tidligere test er afsluttet.
- 8) Angiver hvor lang tid værktøjet kører. I dette tidsrum vil hver tråd køre så mange iterationer den kan nå indenfor de konfigurerede parametre.
 - 9) Yderligere konfigurationsparametre til hhv.
 - a. Ved afkrydsning bliver det virtuelle bibliotek i Fiddler-optagelsen erstattet af det virtuelle bibliotek, der er angivet i pkt 2. Skal benyttes, hvis Fiddler-optagelserne er gjort på en anden server.
 - b. Ved afkrydsning bliver der tilføjet debugging-informationer, hvilket medfører at der skrives printer detaljerede informationer om hvert request værktøjet udfører i tekstboksen i skærmbilledet. I gældende kørsler er denne ikke afkrydset af for at maksimere værktøjets ydeevne.
 - c. Ved afkrydsning vil værktøjet benytte de ventetider, der er indeholdt i Fiddler-optagelserne (dvs. de naturlige ophold, der opstår, når systemet bliver betjent af et menneske). Dette giver et mere retvisende billede af brugen af systemet, idet systemet vil vente den tid, der blev ventet da Fiddler-optagelsen blev udført. Herved sikres, at hver tråd ikke kører hurtigere end da den manuelle optagelse blev foretaget.
 - d. Ved afkrydsning angiver det at der skal vælges identer på sagen ud fra en liste af CPR-numre. Denne sikrer, at alle sager ikke bliver oprettet på en og samme ident. I stedet vælges identerne ud fra en konfigureret liste af CPR-numre
 - 10) Valg af mapningsfilen. Mapningsfilen bruges til at konfigurere tekniske parametre i de http-forespørgsler, der sendes mod POLSAG systemet. De tekniske parametre afgør, hvordan systemet indsætter systemnøgler, CPR-numre og lignende i http-forespørgslerne for at sikre der bliver oprettet unikke sager med unikke identer hver gang.
 - 11) Valg af Values-filen. Values filen indeholder lister af CPR-numre som der vælges identer ud fra.
 - 12) Valg af den Fiddler-optagelse, der skal benyttes til testen. I dette skærmbillede er der fx valgt "A Grundregistrering"-optagelsen (test case A), som opretter en sag med en sagsdisposition og en anmelder påhæftet.
 - 13) Betyder at testen påbegyndes uden forhåndsoprettelse af sager. Der anvendes ikke forhåndsoprettelse af sager i følgende test cases: 0 Load af forside, A Grundregistrering, G Søg, J Træk døgnrapport. I test case A er sagsoprettelsen en del af selve test casen. I de øvrige test cases oprettes der ikke en sag først, da de blot udgør læsninger i systemet.
 - 14) Angiver antallet af sager, der skal forhåndsoprettes inden testen kan påbegyndes. Denne anvendes ved de test cases, der kræver forhåndsoprettelse af sager, hvilket er følgende test cases: B Opret dokument, C Ny sagsplacering, D Sigtelse, E Sikret genstand, F Køretøj, H Dan sagsstyringssag, I Afslut sag og K Fremsøg pakke dokumenter.
 - 15) Angiver den Fiddler-optagelse, der anvendes til at forhåndsoprette sagerne. Dette består på nuværende tidspunkt altid en "trimmet" version af A Grundregistrering test case'n, som kun sender de http-forespørgsler, der er absolut nødvendige for at få POLSAG til at oprette en sag.
 - 16) Valg af den Fiddler-optagelse, der benyttes til selve testen. Her genbruges de sagsnumre, der er blevet forhåndsoprettet forud for kørslen af selve testen. Det er således vitalt at Fiddler-optagelsen ikke indeholder en sagsoprettelse.
 - 17) Betyder at testen påbegyndes med forhåndsoprettelse af sager, hvorefter den evt. venter med at køre selve testen til det i punkt 3 angivne tidspunkt.
 - 18) Nulstiller værktøjets interne log.
 - 19) Gemmer de valgte konfigurationsparametre i en fil.

- 20) Viser loggen for den aktive test kørsel.
- 21) Viser hvor mange tråde, der er aktive (kun aktiv under testafvikling).
- 22) Viser hvor mange tråde, der er afsluttet (kun aktiv under testafvikling).
- 23) Viser, hvor mange iterationer af den konfigurerede Fiddler-optagelse værktøjet er nået igennem.
- 24) Viser hvor mange af de udførte iterationer af Fiddler optagelsen, som ikke opfylder de konfigurerede tidsparametre. Hvis der fx er sat 400 sekunder af i punkt 7, så må hver iteration ikke tage længere end 400 sekunder, da systemet så bliver for langsomt til at kunne opfylde måltallene for den pågældende test case og dermed vil Rigspolitiets tal for, hvad systemet skal kunne nå indenfor en time ikke kunne nås.

Tekniske detaljer omkring POLSAG Load Tool

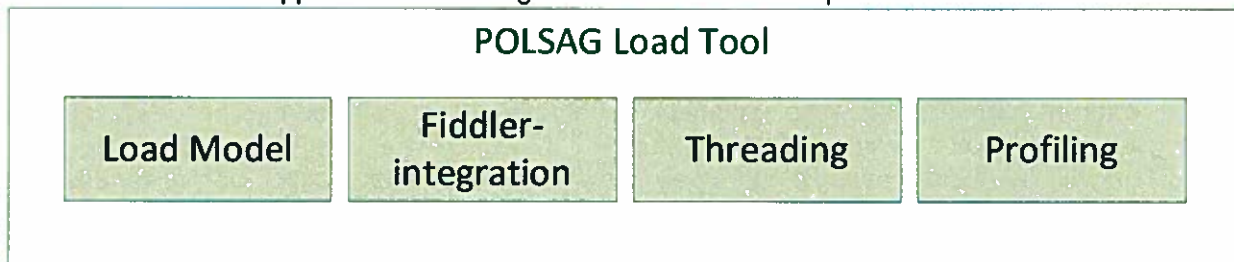
POLSAG Load Tool er udviklet specifikt til POLSAG Performance Review, og genbruger nogle af Globeteams eksisterende performance-relaterede komponenter.

Totalt set består løsningen af følgende komponenter.

- 1) Selve POLSAG Load Tool-værktøjet, som tager form af en Windows-applikation.
- 2) Globeteam Fiddler-based Wire Traffic Recorder, som bruges til at gemme den afviklede test i Fiddler-format. Det er således ikke nødvendigt at bruge Fiddler-applikationen til at optage testafviklingen, hvilket eliminerer en kilde til testusikkerhed.

POLSAG Load Tool-applikationen er udviklet til Microsoft .NET 3.5 Service Pack 1, i programmeringssproget C#.

POLSAG Load Tool-applikationen har følgende overordnede komponentarkitektur:



Load Model indeholder de basale konfigurationsparametre for POLSAG Load Tool, der gør det muligt at konfigurere i overensstemmelse med alle Rigspolitiets ønsker til tests af systemet. Dette indbefatter også metoden til at udskifte udvalgte dele af http-forespørgslerne for at sikre at der oprettes unikke sager fordelt på flere identier og flere forskellige sagsbehandlere.

Fiddler-integration anvender Fiddlers API til at læse Fiddler-optagelser, som anvendes til at genafspille http-forespørgsler i forbindelse med testen.

Threading indeholder standardbiblioteker til afvikling af flere "tråde" i systemet, som simulerer flere samtidige brugere af systemet.

Profiling indeholder standardbiblioteker til tidtagning i forbindelse med de forskellige operationer i systemet for at sikre at Load Tool'et ikke risikerer at blive flaskehalsen i forbindelse med testkørslen. Undervejs i testkørslerne måler Load Tool'et detaljeret hvor lang tid det er om at

udføre interne operationer således disse holdes udenfor de gældende målinger på POLSAG systemet.

Globeteam Fiddler-based Wire Traffic Recorder er udviklet til Microsoft .NET 3.5 Service Pack 1 i programmeringssprogt C#. Denne anvender Fiddlers API til at optage http-forespørgsler sendt af Load Tool for sidenhen at kunne sammenligne disse med de manuelle optagelser af samme test case, validere at alting er forløbet korrekt, udføre statistik på kørslerne, undersøge udvalgte detaljer i http-forespørgslerne osv.

Testforløbet

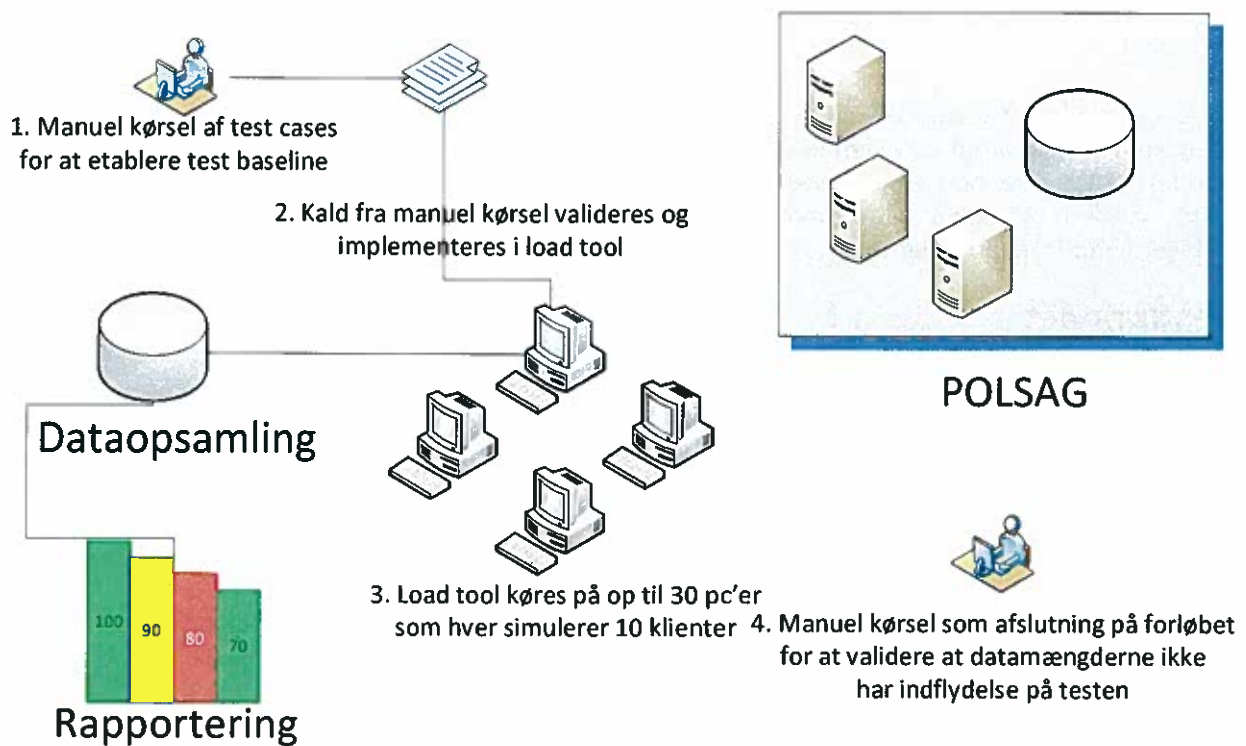
Forløbet op til afviklingen af testene er som følger:

1. Den pågældende test case specificeres (se appendix A).
2. På basis af dokumentationen af test case'n udføres der en manuel testkørsel fra web-browseren på en repræsentativ pc-klient op mod det pågældende servermiljø, hvor brugeren sørger for at arbejde i det tempo som forventes at modsvare en habil bruger. Under den manuelle kørsel optager Fiddler-værktøjet så alle http requests som sendes afsted fra browserklienten mod serveren og tilbage fra serveren til web-browseren bliver indsamlet. Efter endt testforløb gemmes denne fil.
3. Der udføres en række aftestninger med den optagede Fiddler-fil fra POLSAG Load Tool (det er vigtigt at bemærke at Fiddler-filen fra den oprindelige manuelle kørsel ikke bliver ændret undervejs i dette forløb). I dette forløb bliver hvert enkelt http request og svaret på samme minutøst sammenlignet med de tilsvarende http request/responses fra den manuelle testkørsel. Når alle requests er korrekte frigives Fiddler-filen til brug.

Selve aftestningen foregår på følgende vis:

0. POLSAG Load Tool installeres på det nødvendige antal pc-klienter.
 1. De relevante testfiler kopieres ind på pc-klienterne.
 2. POLSAG Load Tool konfigureres baseret på den ønskede test (se afsnittet "Beskrivelse af POLSAG Load Tool" for en nærmere beskrivelse af konfigureringsmulighederne).
 3. Testen afvikles.
 4. Alle de resulterende Fiddler optagelser gemmes.
 5. Det valideres at alle de resulterende filer er korrekte relativt til den oprindelige manuelle testkørsel, hvorefter testen erklæres korrekt gennemført. Hvis der har vist sig at være problemer med afviklingen startes der fra punkt 2 igen, medmindre der er tale om en fejl i testen.

Aftestningsforløbet er illustreret på figuren nedenfor.



Forudsætninger

Testene og de afledte resultater, analyser og konklusioner i performance review'et bygger på følgende forudsætninger:

- Uddannelsesmiljøet er repræsentativt for produktionsmiljøet – I processen der ledte frem til beslutningen om at benytte uddannelsesmiljøet til performance review'et blev det konkluderet, at indenfor den mulige tidsramme var uddannelsesmiljøet den bedst farbare vej givet at etableringen af et nyt miljø ville tage for lang tid samt at leverandøren vurderede at miljøet er repræsentativt relativt til produktionsmiljøet.
- De 13 test cases og Rigspolitiets test case-plan giver et nuanceret og retvisende billede af systemets "all round"-ydeevne. Dvs. det lægges til grund at de udvalgte test cases er repræsentative for belastningen på POLSAG og test case-planen stemmer nogenlunde overens med den belastning, der vil være på POLSAG om mandagen i relation til de udvalgte test cases.
- Systemet vil ikke blive udsat for en markant større belastning end angivet af de 13 test cases fra test case-planen. Hvis der enten er en væsentlig belastning fra aktiviteter, der ikke er indeholdt i de 13 test cases, eller antallet af kørsler for en eller flere af test case'ne er undervurderet vil systemet blive belastet væsentligt hårdere end det er tilfældet i performance review'et.
- Alle testkørsler er lagt så jævnt henover den givne testperiode, som der er adgang til uden at der opstår overlap imellem disse. Såfremt brugeraktiviteten på POLSAG er klumpet betydeligt mere sammen i tid vil systemet således blive belastet væsentligt hårdere end i performance review'et.
- Leverandøren udfører ikke ændringer i uddannelsesmiljøet i testperioden – Rigspolitiet har informeret leverandøren om at der ikke må udføres ændringer i uddannelsesmiljøet i testperioden for at sikre at alle testresultater er sammenlignelige.

- Web-browseren indeholder ikke selvstændig funktionalitet, der betyder at aftestningen af performance ved afspilning af http requests ikke giver et korrekt resultat – Risikoen for at der er forskel mellem manuel afvikling i web-browseren og ved hjælp af automatiseret afspilning af http requests er blevet afdækket ved at gennemføre compliance tests, tilføje pauser i trafikken som modsvarer hvad en habil bruger ville gøre samt ikke mindst kompensere for manglende caching.
- Leverandøren vil være i stand til at eliminere de mange 401-fejl, eftersom der er sket en manuel efterbehandling af testresultaterne, hvor alle 401-fejl er filtreret bort for at gøre tallene sammenlignelige. Mao. er der ikke nogen 401-fejl i testresultaterne og testresultaterne ville således være dårligere i virkeligheden, hvis ikke leverandøren er i stand til at få elimineret 401-fejlene i POLSAG-miljøerne. Selvom 401-fejlene bliver filtreret ud af testresultaterne vil serveren dog naturligvis stadig være belastet af at udsende disse reponses, hvorfor det bør aflede en lille reduktion i belastningen af serverne, når 401-fejlene er elimineret, som den manuelle efterbehandling ikke kan tage højde for.

Det er vigtigt at være opmærksom på at POLSAG-miljøet kun bliver belastet med de 13 test cases og samtidig er spredt helt ensartet og mest muligt ud over den tid, der er til rådighed for testen, hvilket gør at testen bør vurderes som et særdeles konservativt skøn over belastningen, idet POLSAG jo også bliver brugt til at udføre andre arbejdsgange end dem, der er repræsenteret i de 13 test cases, ligesom belastningen i de fleste systemer som regel har en tendens til at ligge tættere på den velkendte normalfordelingskurve (eller Gausskurven) end den helt jævne kurve.

Alene af denne årsag yder modelleringen som sådan ikke nogen garantier for at brugernes oplevelse af POLSAG vil være som indikeret af aftestningen. Omvendt forventes der at være sikkerhed for at brugeroplevelsen i hvert fald ikke vil være bedre end angivet af performance review'et.

Efter Globeteams opfattelse betyder dette at Rigspolitiet bør lægge til grund at resultaterne fra performance review'et er de absolutte minimumskrav, der bør stilles til POLSAG. Man bør til gengæld ikke uden videre forudsætte at opfyldelse af de forventninger og krav, der opstilles i forbindelse med performance review'et er tilstrækkeligt til at POLSAG kan levere de forventede svartider i virkeligheden.

Risici

Analyserne og konklusionerne i performance review'et er behæftet med følgende risici:

1. Om uddannelsesmiljøet reelt er repræsentativt – Eftersom uddannelsesmiljøet såvel som produktionsmiljøet tager form af "black boxes" for Globeteam, på grund af at der ikke har været on-line adgang til disse, har det ikke været muligt at validere påstanden om at uddannelsesmiljøet er repræsentativt for produktionsmiljøet, bortset fra muligheden for at sammenligne at testresultaterne er nogenlunde ens på tværs af miljøerne og stemmer overens med Rigspolitiets hidtidige erfaringer.
2. Servermiljøet tager form af en "black box" – Performance review'et er som sådan nødt til at forudsætte at miljøet er konfigureret korrekt i relation til webservere, caching, POLSAG-applikationen, databaseserveren etc. Der har således ikke været nogen muligheder for at validere noget som helst, da Globeteam ikke på noget tidspunkt har haft adgang til at kigge på serveren før, efter eller under test.
3. Servermiljøet er en "black box" under aftestningen – Der har ikke været nogen muligheder for at følge, hvordan miljøet opfører sig løbende under aftestningen, eftersom testerne ikke

havde direkte adgang til servermiljøet og performancedata tidligst forelå med en dags forsinkelse.

4. Målingerne er i al praksis begrænset til den direkte eller indirekte brugeroplevelse af det totale system – Målingerne sker på hele POLSAG-løsningen, inkl. pc-hardware, serverhardware, standardsoftware samt konfigurationen af samme, POLSAG-systemet og randsystemer. Såfremt en eller flere af komponenterne eksempelvis ikke er konfigureret korrekt, så vil det naturligvis have en konsekvens for resultaterne og kan worst-case medføre at resultaterne ikke er retvisende.

Der er blevet gennemført stikprøver på netværkets round trip-tider undervejs i testforløbet. Stikprøverne afdækkede ikke væsentlige forskelle i netværkets round trip-tider. Round trip-tiden fra pc'erne i Midt/Vest, som blev benyttet til testen ligger størstedelen af tiden stabilt omkring 15-16 ms (og nogle enkelte eksempler på round trip-tider på 31 ms).

Der er ligeledes udført en validering af at antallet af requests i test case 0 er retvisende relativt til den caching der foregår på web-browseren på Politiets pc-klient, da dette kan medføre relativt store forskelle i antallet af requests, der udføres under testen. Testen på pc-klienten angiver at klienten anvender følgende antal requests i disse situationer:

- Load af forside: 552 requests
- Reload af forside: 269 requests
- Load af forside efter web-browseren har været lukket: 552 requests

På den baggrund vurderes test case 0 at være retvisende for POLSAG-miljøets opførelse.

Alle de resterende test cases er i øvrigt optaget med en "lunken" web-browser. Med "lunken" menes at optagelsen typisk er taget fra 2. eller 3. gennemkørsel af den givne test case fra pc'en (således at en evt. caching på web-browseren udnyttes), men i en ny web browser instans. Testen sikrer altså at web browseren ikke behøver indlæse evt. statiske ressourcer, fordi de allerede er indlæst i det omfang web-browseren og web-serveren i fællesskab kan finde ud af at styre og koordinere dette.

Brugeren var i øvrigt i alle tilfælde logget direkte på pc'en med den relevante brugerkonto og anvendte således ikke Run As⁷.

Det forhold at POLSAG-servermiljøet tager form af en black box set fra performance review'ets synsvinkel har haft væsentlig indflydelse på testforløbet, idet det medførte at man ved alt andet end helt oplagte fejl i reglen tidligst har opdaget et evt. helt eller delvist nedbrud i servermiljøet et stykke tid efter testen er færdigafviklet og i visse tilfælde tidligst dagen efter testen, hvor leverandøren har udleveret de opsamlede performancedata. I en lang periode modtog Globeteam dog først de aftalte performancedata væsentligt senere (i et enkelt tilfælde skete afleveringen fra leverandøren endog med to ugers forsinkelse).

Det at testen er sket op mod en "black box" betyder endvidere at det ligeledes ikke har været synligt for Globeteam, hvad hostingleverandøren gør/ikke gør i forbindelse med nedbrud samt hvornår dette udføres. Netop denne problematik har desværre været meget mærkbar i forhold til POLSAG performance review'et, idet der har været konstateret ustabilitet i POLSAG-systemet,

⁷ Dette gælder dog ikke for optagelserne fra produktionsmiljøet, hvor det var påkrævet at køre som Run As, idet det ikke var muligt at logge på pc'en med en produktionsbruger.

herunder har der ofte forekommet nedbrud på webserverne og ved flere lejligheder er der ligeledes blevet konstateret nedbrud på adgangen til databaseserveren.

Disse ustabiliteter har bl.a. også bevirket at det ikke på forhånd har været muligt at forudsige, hvor længe det tager at køre en given test (og hvorvidt den er brugbar). På grund af den manglende online-adgang til serverne har der heller ikke været mulighed for at konstatere, hvornår en test reelt er kørt færdig, når der opstår timeouts eller tilsvarende i operationerne. Dette forhold koblet med at det tager endog meget lang tid at køre testene, selvom der er gjort alt for at reducere dem, har betydet at testoperatøren var overladt til at gætte sig frem til, hvornår systemet var klar til næste test.

Dette indebærer en væsentlig øget risiko for overlap mellem testene. Det er generelt lykkedes testoperatøren at begrænse dette overlap til de såkaldte præ-test faser (der lægger data i databasen som forberedelse), som i visse tilfælde ikke er blevet kørt helt til ende før den egentlige test er startet, idet der stadig udestod svar på nogle få operationer (det sidste response, som angiver at sagen er færdigoprettet; i mange tilfælde har det kunne tage flere timer før det kom retur).

Dette kan naturligvis kun påvirke de pågældende testresultater i negativ retning. Den præcise betydning er i sagens natur svær at opgøre helt præcist, men vurderes – baseret på analyser af den samme test kørt under forskellige forhold – ikke ikke at have haft afgørende betydning for nogen af de resultater, der indgår i rapporten.

Der er samtidig en række interessante spørgsmål i relation til, hvordan de forskellige dele af miljøet håndterer de forskellige belastninger, som på nuværende tidspunkt ikke kan besvares i performance review'et på grund af Globeteams manglende indsigt i miljøets dynamik under testen.

Globeteam har i øvrigt bemærket at der muligvis har været udført en enkelt ændring i uddannelsesmiljøet i testperioden. Baggrunden for denne påstand er at en parameter i forbindelse med oprettelse af sager (update_code) uden varsel pludselig havde ændret værdi⁸, hvilket bevirkede at sagsoprettelsesdelen af testen pludselig holdt op med at virke. Globeteam er dog i tvivl om dette har noget med brugerens konfiguration af sikkerhedsgrupper at gøre.

⁸ Et kald til BusinessProcessHandler ifm. oprettelse af sager tog i en lang periode update_code "c2", hvorefter den pludselig var skiftet til "c505"

Denne ændring har ikke haft nogen konsekvenser ud over at testen skulle tilpasses for at fungere. Men der eksisterer som nævnt en risiko for at evt. andre (ikke-autoriserede) ændringer kan have haft konsekvenser for aftestningen.

Evt. fravalg og udeståender

Foranalysens konklusion om at der ikke ville blive oprettet et separat testmiljø med fuld adgang til servermiljø såvel som POLSAG-kildekode betød samtidig et fravalg af mulighederne for at grave dybere i årsagerne til de konstaterede problemer samt at afprøve de forskellige muligheder for fixes og andre forbedringer sammen med leverandøren.

Performance review'et rummer således ikke nogen selvstændig validering af de muligheder, der findes for at forbedre POLSAG, herunder ikke mindst mulighederne for skære ned på antallet af requests fra klienten, udføre databasetuning og forbedre systemets håndtering af randsystemdata, herunder evt. etablering af caching af data fra randsystemerne, som sjældent ændrer sig og ofte finder anvendelse.

Der har ligeledes ikke været nogen muligheder for at imødekomme leverandørens ønske om at justere på parametre undervejs i testen på grund af hensynet til at der bliver indsamlet ensartede og sammenlignelige testresultater.

Introduktion til testene

Denne sektion giver en kort introduktion til testene, der er blevet udført i performance review'et.

Test cases, test case-planen og de opstillede modeller

Alle aftestninger er som tidligere nævnt baseret på de 13 **test cases**, som er specificeret i Appendix A. Disse 13 test cases udgør Rigspolitiets bedste bud på de oftest forekommende og/eller tungeste aktiviteter, der bliver udført på POLSAG.

Det er således formodningen at disse test cases betyder at aftestningen og de deraf afledte resultater er en realistisk model for et udsnit af den reelle brug af POLSAG. Af samme årsag blev test cases forelagt for leverandøren til kommentering undervejs i performance review'et⁹.

Årsagen til at testen består af 13 test cases skyldes ene og alene begrænsningerne i tid og ressourcer.

De 13 test cases indgår i den såkaldte **test case-plan**, som Rigspolitiet vurderer, er repræsentativ for den belastning på POLSAG der forventes at være i løbet af en mandag i en typisk gennemsnitlig politikreds i relation til hver af de 13 test cases:

Tid	0	A	B	C	D	E	F	G	H	I	J	K	L
8-8.15	200										100		
8.15-9	150	300	111	333	39	200	22	1750	39	111	70	30	
9-10	250	300	111	333	39		22	1750	39	111		30	6
10-11	250	300	111	333	39		22	1750	39	111		30	6
11-12	200		111	333	39		22	1750	39	111		50	6
12-13			111	333	39		22	1750	39	111		60	6
13-14			111	333	39		22	1750	39	111		60	6
14-15			111	333	39		22	1750	39	111		50	6
15-16			111	333	39		22	1750	39	111		30	6
16-17			111	333	39		22	1750	39	111		10	

Tid	0	A	B	C	D	E	F	G	H	I	J	K	L
8-8.15	200										100		
8.15-9	150	300	111	333	39		22	1750	39	111	70	30	
9-10	250	300	111	333	39		22	1750	39	111		30	6
10-11	250	300	111	333	39		22	1750	39	111		30	6
11-12	200		111	333	39		22	1750	39	111		50	6
12-13			111	333	39		22	1750	39	111		60	6
13-14			111	333	39		22	1750	39	111		60	6
14-15			111	333	39		22	1750	39	111		50	6
15-16			111	333	39		22	1750	39	111		30	6
16-17			111	333	39		22	1750	39	111		10	

Tallene i test case-planen angiver antallet af kørsler af den pågældende test case (kolonne) i det pågældende tidspunkt (række). Test case-planen er baseret på brugen af POLSAS-systemet for

⁹ Leverandørerne havde nogle bemærkninger, som førte til tilpasninger af testene. Denne del af forløbet blev afsluttet efter leverandørerne ikke havde yderligere bemærkninger.

Midt/Vest politikredsen og er valideret af de relevante parter. Test case-planen er beskrevet i detaljer i Appendix B.

På basis af test case-planen er der udarbejdet to modeller for belastningen af systemet: Belastningsmodellen og 8.00-8.15 modellen.

Belastningsmodellen opstiller et mål for, hvor hvilken belastning i relation til hver af de 13 test cases, som POLSAG forventes at blive udsat for i løbet af en time. Modellen er baseret på en syntetisering af Rigspolitiets estimater på belastningen, som POLSAG formodes at blive udsat for mandag formiddag jævnfør test case-planen.

Belastningsmodellen ser således ud:

Tid	0	A	B	C	D	E	F	G	H	I	J	K	L
Måltal pr time	350	300	110	340	40	200	25	1750	40	110	70	30	

Test case L er ikke indeholdt i den nuværende belastningsmodel, idet L p.t. kun kan køres manuelt.

8.00-8.15 modellen opstiller et mål for, hvor hvilken belastning i relation til hver af de 13 test cases (der består i opstart af POLSAG-systemet på 200 pc'er og visning af 100 døgnrappporter), som POLSAG forventes at blive udsat for i løbet af arbejdsdagens første 15 minutter. Modellen er baseret på den belastning, som POLSAG formodes at blive udsat for mandag mellem 8.00 og 8.15 jævnfør test case-planen.

8.00-8.15 modellen ser således ud:

Tid	0	A	B	C	D	E	F	G	H	I	J	K	L
Måltal pr time	200										100		

De tre tests

De to modeller bliver herefter omformet to tests: Belastningstesten og 8.00-8.15 testen.

Belastningstesten består af det antal kørsler af hver af de 13 test cases, som er angivet i belastningsmodellen. For at spare testtid bliver belastningstesten reduceret til en halv times test, hvilket sker ved at halvere modellens måltal (der jo gælder for en time), hvilket bør have samme validitet og give samme resultat.

Belastningstesten indeholder ikke test case L (bødesager), idet denne test ikke er automatiseret. Dette skyldes at testen den involverer ældre systemer, der ikke med rimelig indsats kan automatiseres.. Denne test case kan om ønsket udføres manuelt, hvilket dog ikke sker i forbindelse med aftestningerne i performance review'et ud fra et ønske om at alle tests skal være så ensartede som muligt.

8.00-8.15 testen består af 200 kørsler af test case 0 og 100 kørsler af test case J, der skal afvikles på 15 minutter, jævnfør 8.00-8.15 modellen.

Der bliver endvidere introduceret en ekstra test ved navn **performancetesten**. Performancetesten består af en kørsel af en af hver af de 13 test cases. Testen lægger således ikke nogen

nævneværdig belastning på POLSAG og benyttes blot til at etablere en baseline for ydeevnen for et system i ro.

Performancetestens formål er som sådan at køre Rigspolitiets test cases manuelt igennem og analysere målingspunkterne optaget ved kørslerne, herunder bla.:

- Hvor mange kald laver en gennemkørsel af hver test case til serveren
- Identificering af langsomme kald
- Identificering af evt. identiske kald

Samtidig repræsenterer performancetesten en lettilgængelig mulighed for at lave approksimationer på antallet af http requests, forbrug af CPU-sekunder mv. ved simple multiplikationer med antallet af gennemløb af performancetesten.

Afviklingen af testene

Testene er opbygget på følgende vis:

1. Manuel kørsel af test cases for at etablere test baseline – Den pågældende test case (se Appendix A) udføres to gange, hvor den anden testkørsel optages med Fiddler til brug for testen. Det tilstræbes at gennemføre testen på samme måde og i samme hastighed som en erfaren POLSAG-bruger ville gøre. Årsagen til at det først er den anden test¹⁰, der optages er at det tilstræbes at testen tager højde for evt. caching af ressourcer for at blive så realistisk som muligt.
2. Kald fra manuel kørsel valideres og implementeres i load tool – Hver enkelt optagelse af test case-kørslen valideres ved manuel gennemgang af indholdet af Fiddler-filen og der udføres i visse tilfælde genkørsler op mod POLSAG-miljøet¹¹. Såvidt det er muligt valideres de enkelte test cases ligeledes ved at gennemse resultaterne i POLSAG-applikationen (ved test case A og D verificeres det således at sagerne er blevet oprettet og at der i tilfælde af A er en anmelder på sagen og i tilfælde af D er både en anmelder og sigtet på sagen). Herefter implementeres det og afvikles fra POLSAG Load Tool for at afprøve at det fungerer efter hensigten. Fiddler-optagelsen af testkørslerne med POLSAG Load Tool bliver gennemgået manuelt fra Fiddler.

Det er de samme manuelle test case-kørsler, der finder anvendelse på tværs af de tre tests.

Testforløbet er som følger:

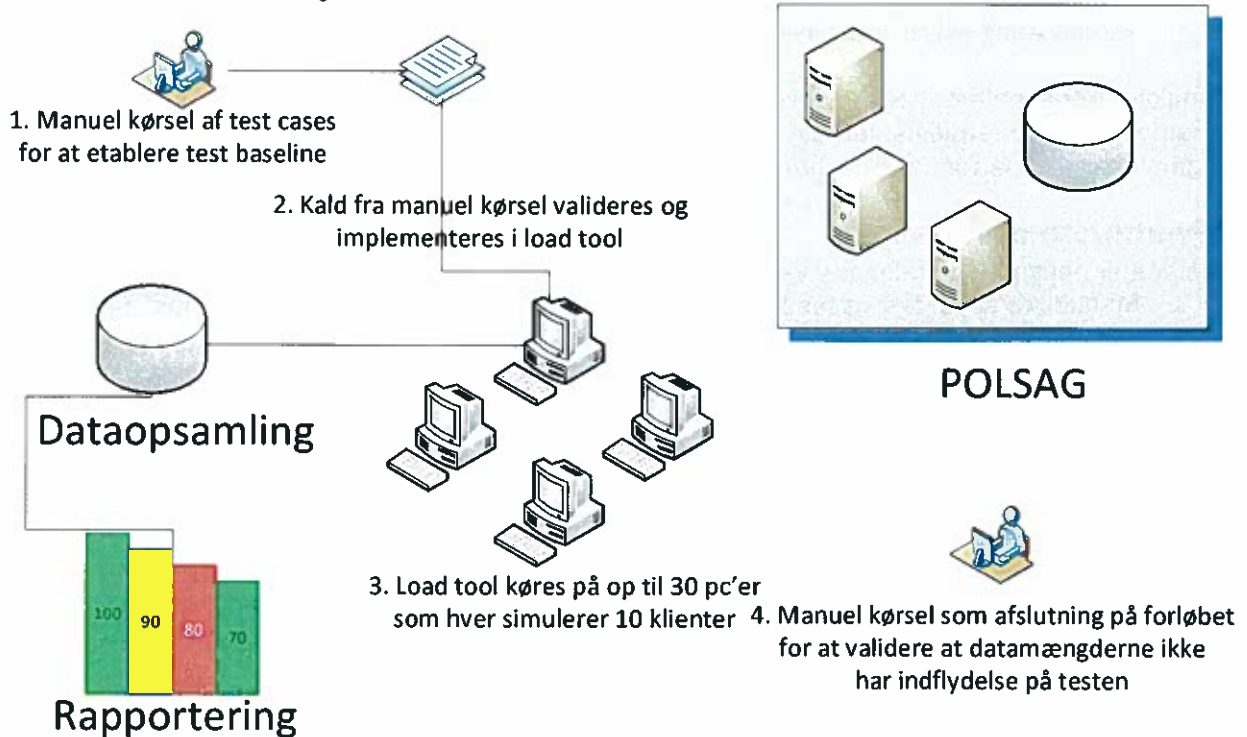
3. POLSAG Load tool køres – POLSAG Load Tool konfigureres som angivet af de detaljerede opsætningsangivelser for den pågældende test på tværs af op til 30 pc'er med angivelse af de korrekte parametre. Hver af de 30 pc'er, som er fysisk placeret i Midt/Vest for at sikre at testen er så virkelighedsnær som muligt, er i stand til at simulere op til 10 klienter.

¹⁰ Det blev i den sammenhæng bemærket at der tilsyneladende ikke sker caching af de statiske ressourcer i forbindelse med test case 0. Dvs. alle kørsler af test case 0 opfører sig på samme måde, som når klienten indlæser forsiden første gang. Selv hvis dette måtte blive optimeret vurderes dette at være en acceptabel tilnærmelse al den stund at klienterne ofte vil skulle indlæse forsiden første gang ved arbejdsdagens begyndelse mandag morgen, medmindre statiske ressourcer opsættes med en for Captia unormalt lang levetid.

¹¹ Der er blevet lagt en stor indsats for at validere testene. POLSAG-applikationens opbygning (hvor http 200 responses i mange sammenhænge benyttes til at returnere fejl som flad tekst) har besværliggjort dette arbejde betydeligt.

4. Manuel kørsel – Ved afslutningen af forløbet sker der i reglen en manuel kørsel, som benyttes til at validere at den forøgelse i datamængden i POLSAG som testen muligvis har afstedkommende (altså de sager, der er blevet oprettet henover testperioden) ikke har haft væsentlig indflydelse på testen.

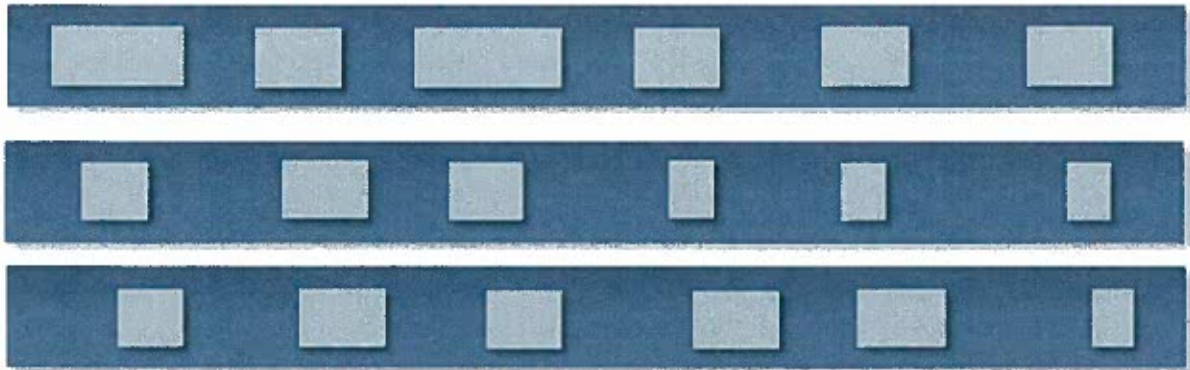
Hele testforløbet er forsøgt illustreret nedenfor:



I relation til opsætningen af testafviklingen i POLSAG Performance Tool er det væsentligt at gøre opmærksom på at testforløbet er udjævnet mest muligt relativt til antallet af testkørsler (hver testkørsel benævnes "en iteration" i testsammenhænge) i hver test case, der skal gennemføres indenfor det pågældende tidsinterval. Dvs. iterationerne er fordelt jævnt ud over tidsperioden, ligesom der opereres med den størst mulige tidsmæssige forskydning i opstarten af hver tråd på test-pc'en.

Testen er ligeledes opsat således at der vil være en tidsmæssig forsinkelse før næste iteration startes op, såfremt testen når at færdiggøre den pågældende iteration indenfor det afsatte tidsinterval. Der sker naturligvis ikke opstart af en ny iteration før den tidligere iteration er gennemført.

Illustration af den fuldstændige udjævning af de enkelte iterationer af testene (dvs. hver tråd er illustreret med de blå kasser; hver iteration, som bliver startet til et bestemt tidspunkt, er illustreret ved en grå kasse; længden af kassen angiver hvor lang tid den pågældende iteration har kørt):



Det forhold at testene er fuldstændig udjævnet over tid og tråde betyder at POLSAG-systemet vil blive udsat for den lavest mulige spidsbelastning. Testen er således for venlig over for POLSAG relativt til spidsbelastningen henset til generelle erfaringer med almindelig brug af systemer, som meget vel kunne vise sig at minde mere om en normalfordeling end en helt jævn belastning.

Testmiljøet

Medmindre andet er angivet eksplicit, så sker alle aftestningerne i performance review'et op mod uddannelsesmiljøet.

Uddannelsesmiljøet består af følgende komponenter på serversiden:

Komponent	DNS-navn	IP-adresse
NLB ¹² (som består af de tre komplekse servere RPLPOLSAGEE901v, RPLPOLSAGEE911v og RPLPOLSAGEE914v)		<anonymiseret>
Simple server	RPLPOLSAGEE916v	<anonymiseret>
Kompleks server	RPLPOLSAGEE915v	<anonymiseret>
Kompleks server (NLB)	RPLPOLSAGEE901v	<anonymiseret>
Kompleks server (NLB)	RPLPOLSAGEE911v	<anonymiseret>
Kompleks server (NLB)	RPLPOLSAGEE914v	<anonymiseret>

Simple servere og komplekse servere er defineret i kontrakten med Alliancen. Den simple server og den komplekse server, der indgår i testen er udstyret med henholdsvis ekstra RAM og en ekstra core relativt til kontrakten.

Den simple server består af 2 cores og 4 GB RAM, mens den komplekse server består af 4 cores og 8 GB RAM.

Alle tests i uddannelsesmiljøet bliver eksekveret ved brug af POLSAG Load Tool-værktøjet, som er installeret på de 30 pc-klienter, der er omfattet af testmiljøet. Pc-klienterne er fysisk placeret i Midt/Vest-politikredsen og udgør således en fuldstændig repræsentativ aftestning af POLSAG for så vidt angår netværksforsinkelse og -båndbredde.

¹² NLB betyder Network Load Balancing; og benyttes her op mod 3 komplekse servere for at sikre optimal fordeling af load blandt disse.

Bearbejdning af testresultaterne i forbindelse med POLSAG Load Tool sker ved at indlæse alle testresultater (i form af Fiddler-optagelserne) i en samlet database, der således indeholder hvert eneste http request. Dette betyder dels at det er muligt at validere hvordan testen er gået relativt til tidligere tests relativt til en række parametre, herunder fx

- Hvor mange iterationer er der blevet gennemført indenfor testperioden relativt til måltallet (dvs. det antal iterationer, der skulle gennemføres for at kunne imødekomme kravene til POLSAGs ydeevne som er udtrykt i testene).
- Hvor mange kald er der udført til serverne
- Hvor lang tid tager hvert kald om
 - o dels at blive transporteret rundt på netværket
 - o dels at blive behandlet af serveren
- Hvordan har systemet reageret på belastningen? Er der evt. opstået fejl i forbindelse med dele af testene¹³?

Det åbner samtidig også mulighed for at sammenligne resultaterne på tværs af flere miljøer.

Hurtigt overblik over de enkelte test cases

Dette afsnit indeholder en kort opstilling af hvordan de enkelte test cases belaster webserverne og netværket baseret på indholdet af de Fiddler-filer, der danner grundlag for alle testkørslerne.

Givet det relativt store antal http requests er det åbenbart at systemet er meget sensitivt overfor forsinkelser eller pakkeab på netværket samt sløve eller pressede webservere.

Et regneeksempel, der demonstrerer dette: Hvis webserveren er i stand til at besvare hvert af klientens requests ved load af forsiden indenfor henholdsvis 10 ms, 20 ms og 100 ms, så vil load af forsiden tage henholdsvis 5,5 sekunder, 11 sekunder og 55 sekunder, eftersom alle requests afvikles synkront.

Eftersom svartiden både indeholder webserverens behandlingstid og round-trip tiden på netværket vil vi generelt vurdere at en gennemsnitlig svartid på 20 ms må betegnes som det teoretisk bedste man kan forvente at opnå i en stor enterprise-installation som Politiets og et system som POLSAG. Svartiden kræver i øvrigt at webserverne er virkelig hurtige til at besvare klientens requests.

Når POLSAG-systemet ikke er belastet (dvs. webservere og databaseserver er i ro) er det generelt oplevelsen at load af forsiden tager ca. 30 sekunder i uddannelsesmiljøet. Erfaringer fra produktionsmiljøet (som anvendes af Bornholms politikreds) er nogenlunde de samme. Bornholm har dog også rapporteret at systemet jævnligt udviser væsentligt højere logintider, hvilket stemmer fint overens med Globeteams oplevelse i uddannelsesmiljøet, når POLSAG-systemet bliver sat under pres.

Test case 0: Åbn forside

Test case 0 har følgende egenskaber:

Fiddler filnavn	0 Initiel load.saz		
Antal requests	554		
	Bytes	KB	MB

¹³ POLSAG-applikationens opbygning (hvor http 200 responses i mange sammenhænge benyttes til at returnere fejl som flad tekst) besværliggør desværre dette arbejde betydeligt.

Sendt	265.558	259,33	0,25
Modtaget	5.349.814	5224,43	5,10

Test case 0 formodes til sammenligning at kunne skæres ned til mindre end 25 requests og en trafikmængde på noget under 100 kb, hvis man udfører operationen fra en optimeret klientapplikation fremfor en web browser. Dette vil indebære at man fjerner alt andet end det absolut nødvendige (dvs. datavalideringer, hentning af defaultværdier, hentning af koder, hentning af Javascript-filer, hentning af gif-billeder, hentning af HTML-dokumenter samt overhead og diverse handshakes elimineres) .

Test case A: Grundregistrering

Test case A har følgende egenskaber:

Fiddler filnavn	A Grundregistrering.saz		
Antal requests	569		
	Bytes	KB	MB
Sendt	425.192	415,23	0,41
Modtaget	1.686.392	1646,87	1,61

Det er i forbindelse med aftestningerne blevet valideret at test case A som minimum vil kunne trimmes til 23 requests og følgende trafikmængde (valideringen består i at der er udviklet en særlig variant af test case A, der benævnes A-Trim, til oprettelse af sager fra POLSAG Load Tool; A-Trim benyttes, når POLSAG Load Tool skal oprette sager forud for testafviklingen), når man udfører operationen fra en optimeret klientapplikation fremfor en web browser:

	Bytes	KB	MB
Sendt	29.304	28,62	0,03
Modtaget	40.465	39,52	0,04

Det forventes at være muligt at skære endnu nogle requests af test case A.

Test case B: Opret dokument

Test case B har følgende egenskaber:

Fiddler filnavn	B_USO_ Opret dokument.saz		
Antal requests	274		
	Bytes	KB	MB
Sendt	316.831	309,41	0,30
Modtaget	1.491.869	1456,90	1,42

Det vurderes at test case B vil kunne trimmes ned til 4 requests og følgende trafikmængde, hvis operationen udføres fra en optimeret klientapplikation fremfor en web browser:

	Bytes	KB	MB
Sendt	75.375	73,61	0,07
Modtaget	1.020	1,00	0,00

Test case C: Sagsplacering

Test case C har følgende egenskaber:

Fiddler filnavn	C_USO_Ny sagsplacering.saz		
Antal requests	145		
	Bytes	KB	MB
Sendt	95.980	93,73	0,09
Modtaget	463.104	452,25	0,44

Det vurderes at test case C vil kunne trimmes ned til 9 requests og følgende trafikmængde, hvis operationen udføres fra en optimeret klientapplikation fremfor en web browser:

	Bytes	KB	MB
Sendt	11.381	11,11	0,01
Modtaget	10.784	10,53	0,01

Test case D: Sigtelse

Test case D har følgende egenskaber:

Fiddler filnavn	D_USO_Sigtelse.saz		
Antal requests	438		
	Bytes	KB	MB
Sendt	351.070	342,84	0,33
Modtaget	852.740	832,75	0,81

Det vurderes at test case D vil kunne trimmes ned til 16 requests og følgende trafikmængde, hvis operationen udføres fra en optimeret klientapplikation fremfor en web browser:

	Bytes	KB	MB
Sendt	29.689	28,99	0,03
Modtaget	23.225	22,68	0,02

Test case E: Sikret genstand

Test case E har følgende egenskaber:

Fiddler filnavn	E_USO_Sikret genstand.saz		
Antal requests	180		
	Bytes	KB	MB
Sendt	119.921	117,11	0,11
Modtaget	295.221	288,30	0,28

Det vurderes at test case E vil kunne trimmes ned til 5 requests og følgende trafikmængde, hvis operationen udføres fra en optimeret klientapplikation fremfor en web browser:

	Bytes	KB	MB
Sendt	3.945	3,85	0,00
Modtaget	5.487	5,36	0,01

Test case F: Køretøj

Test case F har følgende egenskaber:

Fiddler filnavn	F_USO_Køretøj.saz		
Antal requests	184		
	Bytes	KB	MB
Sendt	128.851	125,83	0,12
Modtaget	226.878	221,56	0,22

Det vurderes at test case F vil kunne trimmes ned til 9 requests og følgende trafikmængde, hvis operationen udføres fra en optimeret klientapplikation fremfor en web browser:

	Bytes	KB	MB
Sendt	7.576	7,40	0,01
Modtaget	13.478	13,16	0,01

Test case G: Søg

Test case G har følgende egenskaber:

Fiddler filnavn	G Søg.saz		
Antal requests	13		
	Bytes	KB	MB
Sendt	9.424	9,20	0,01
Modtaget	31.435	30,70	0,03

Det vurderes at test case G vil kunne trimmes ned til 2 requests og følgende trafikmængde, hvis operationen udføres fra en optimeret klientapplikation fremfor en web browser:

	Bytes	KB	MB
Sendt	1.690	1,65	0,00
Modtaget	8.962	8,75	0,01

Test case H: Dan sagsstyring

Test case H har følgende egenskaber:

Fiddler filnavn	H_USO_Dan sagsstyring.saz		
Antal requests	433		
	Bytes	KB	MB
Sendt	358.042	349,65	0,34
Modtaget	519.953	507,77	0,50

Det vurderes at test case H vil kunne trimmes ned til 23 requests og følgende trafikmængde, hvis operationen udføres fra en optimeret klientapplikation fremfor en web browser:

	Bytes	KB	MB
Sendt	44.671	43,62	0,04
Modtaget	29.094	28,41	0,03

Test case I: Afslut sag

Test case I har følgende egenskaber:

Fiddler filnavn	I_USO_Afslut sag.saz		
Antal requests	676		
	Bytes	KB	MB
Sendt	523.724	511,45	0,50
Modtaget	710.104	693,46	0,68

Det vurderes at test case I vil kunne trimmes ned til 41 requests og følgende trafikmængde, hvis operationen udføres fra en optimeret klientapplikation fremfor en web browser:

	Bytes	KB	MB
Sendt	73.902	72,17	0,07
Modtaget	61.692	60,25	0,06

Test case J: Træk døgnrapport

Test case J har følgende egenskaber:

Fiddler filnavn	J Træk døgnrapport.saz		
Antal requests	202		
	Bytes	KB	MB
Sendt	128.910	125,89	0,12
Modtaget	1.623.117	1585,08	1,55

Det vurderes at test case J vil kunne trimmes ned til 4 requests og følgende trafikmængde, hvis operationen udføres fra en optimeret klientapplikation fremfor en web browser:

	Bytes	KB	MB
Sendt	19.255	18,80	0,02
Modtaget	5.561	5,43	0,01

Test case K: Fremsøg pakkedokumenter

Test case K har følgende egenskaber:

Fiddler filnavn	K_USO_Fremsøg pakke dokumenter.saz		
Antal requests	917		

	Bytes	KB	MB
Sendt	818.517	799,33	0,78
Modtaget	2.239.840	2187,34	2,14

Det vurderes at test case K vil kunne trimmes ned til 32 requests og følgende trafikmængde, hvis operationen udføres fra en optimeret klientapplikation fremfor en web browser:

	Bytes	KB	MB
Sendt	131.970	128,88	0,13
Modtaget	42.841	41,84	0,04

Observationer omkring forskellene mellem performance review-testen og den kontraktuelle test i AP3

Den følgende analyse er baseret på kørsel af den kontraktuelle performance test i August 2011 (altså forud for det tidspunkt, hvor leverandøren opgraderede AP3-miljøet).

I forbindelse med gennemgangen af resultaterne fra performance review'et har Globeteam bemærket at leverandørens kontraktmæssige test i AP3-miljøet ser ud til at indeholde væsentligt færre requests end Globeteams test, som baserer sig på Rigspolitiets oplysninger omkring belastningen i POLSAS.

Baseret på indholdet af IIS logs ser det ud til at leverandørens kontraktmæssige test består af ca. 200.000 http requests henover 90 minutter, hvilket angiveligt repræsenterer 1/8 belastning og dermed mindst 1½ kredse.

Til sammenligning består belastningstesten af 656.000 http requests henover en time for en kreds (hvilket så vokser til 984.000 requests på 90 minutter og 1.476.000 requests for 1½ kredse i 90 minutter).

Da kontrakten udsiger at den kontraktmæssige test skal ske med normal belastning (hvilket er defineret til højst 80% belastning af systemet) skulle man således formode at der ville blive afspillet mindst 6 gange flere http requests i AP3-testen.

En lidt mere detaljeret sammenligning baseret på IIS logs fra AP3-miljøet fortæller følgende (bemærk at leverandørens test kører i 90 minutter):

URI	Antal http requests i belastningstesten over 60 minutter	Antal http requests i den kontraktuelle test i AP3 over 90 minutter
/rplSAGu/ClientServices/Index2Lib.ashx	841	0
/rplSAGu/ClientServices/Index2.ashx	5074	639
/rplSAGu/ClientServices/F0081.ashx?	223	0
/rplSAGu/ClientServices/BPHandler.ashx	4663	2961
/rplSAGu/create.asp?	592	388
/rplSAGu/CreateFromUrl.asp*	3475	426
/rplSAGu/search.asp*	112579	120586

De forskellige kald der er vist ovenfor er:

- Index2Lib: Benyttes til at hente ident'er (fx til at sætte på sagerne)
- Index2: Benyttes til at søge efter ident'er og lave ping (IdentAdviseringAlive)
- F0081: Udskil og Dan sagsstyring (hvilket så åbenbart ikke er omfattet af AP3-testen)
- BPHandler, create.asp og CreateFromUrl: Benyttes til at oprette eller ændre data. Det kræver dog adgang til at se parametrene for at afgøre præcis hvad der sker.
- Search er den generelle søgemekanisme ind i Captia

Baseret på denne information udleder vi at AP3-testen ikke benytter POLSAGs funktioner til at hente ident'er samt Udskil og Dan sagsstyring funktionen. Vi formoder således at de opretter sager ved brug af nødident'er eller evt. med ident'er, der er hard-code't til testapplikationen.

Men hvad der vel nok er vigtigst: Der er en klar overvægt af søgninger i testen, hvilket angiveligt ikke er repræsentativt for Politiets brug af systemet. Søgninger er i øvrigt noget af det, der perform'er og skalerer bedst på POLSAG-systemet, hvorfor det også gør testen mindre repræsentativ og på et lavere belastningsniveau end den forventede belastning på systemet, som angivet af Rigspolitiet.

Testresultater

Dette afsnit indeholder resultater for de forskellige tests der er blevet afviklet mod POLSAG-miljøet.

Beskrivelsen af de forskellige tests der vises er placeret i Appendix C. Hvis ikke læseren allerede er bekendt med den pågældende test bør man således læse Appendix C før resultaterne studeres.

Indledning

Under vores testkørsler har det i alt for mange tilfælde været svært at køre testene succesrigt igennem af følgende grunde:

- POLSAG-miljøet har vist sig at gå helt eller delvist ned ved høj belastning.
- POLSAG-miljøet har vist sig at have svært ved at klare belastninger, hvilket kommer til udtryk ved at der er mange requests, der ganske enkelt ikke bliver besvaret eller en stor variation i svarhastigheden og antallet af requests, der kommer igennem.

Alene ovenstående bevirker, at Globeteam ikke mener at det vil være forsvarligt at idriftsætte POLSAG på nuværende tidspunkt.

Det er ligeledes essentielt at forstå følgende termer, der benyttes i forbindelse med opgørelsen af resultaterne:

- ❖ Servicetid/request: Den tid fra klienten sender et request til den modtager en response fra serveren.
- ❖ Iteration: Gennemførelse af en kørsel af en given test case (fx opstart af en klient eller oprettelsen af en sag).
- ❖ Servicetid/iteration: Summen af servicetid brugt for en iteration. Dvs. den effektive tid, der er forbrugt til en gennemkørsel af en given test case.
- ❖ Servicetid / Request: Den gennemsnitlige tid, hver request tager (målt i servicetid) for en given test case.

Det er meget vigtigt at forstå at servicetid betyder at man fjerner alle de pauser der er lagt ind i testene for at emulere en normal brugers måde at arbejde på samt tiden til JavaScript rendering og andre aktiviteter på klienten. Servicetid angiver altså præcis den tid, der bliver brugt fra klienten sender et http request til den har modtaget svaret. Servicetid giver som sådan et godt svar på, hvor meget tid der bliver forbrugt til den givne aktivitet, excl. klientens og brugerens tidsforbrug.

Brugeren oplever med andre ord **altid** længere svartider end angivet af servicetiden, fordi servicetiden ikke medregner brugerens reaktionstid og klientens tidsforbrug til fortolkning af svarene, rendering af skærbilleder etc.

Derudover er der medtaget følgende performance counters for hovedparten af testene for at give et indblik i, hvordan den pågældende test har belastet webserveren:

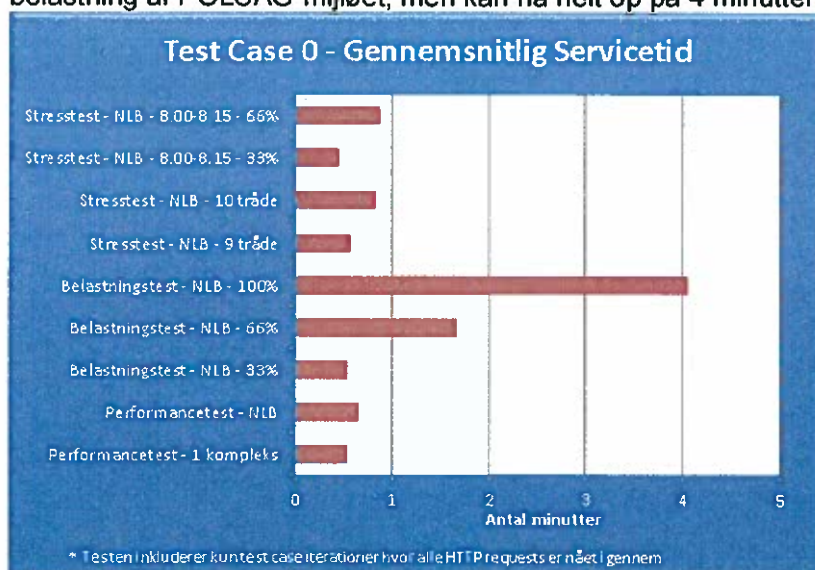
- Active Server Pages\Request Wait Time – Antallet af millisekunder, som det seneste request ventede i køen på ASP (Active Server Pages).
- Active Server Pages\Requests Executing – Antallet af requests, der p.t. er under afvikling på ASP (Active Server Pages).
- Active Server Pages\Requests/Sec – Det gennemsnitlige antal requests, der blev afviklet pr. sekund på ASP (Active Server Pages).

- Memory\Available MBytes – Størrelsen af fysisk hukommelse (antallet af bytes), der er tilgængeligt for de processer, der bliver afviklet på serveren.
- Processor(_Total)\% Processor Time – Den gennemsnitlige procentdel af den forbrugte tid, hvor alle CPU'erne i serveren er aktiv. Aktiv defineres som den tid, der er blevet forbrugt til at udføre non-idle tråde.

Generelle testsammenligninger

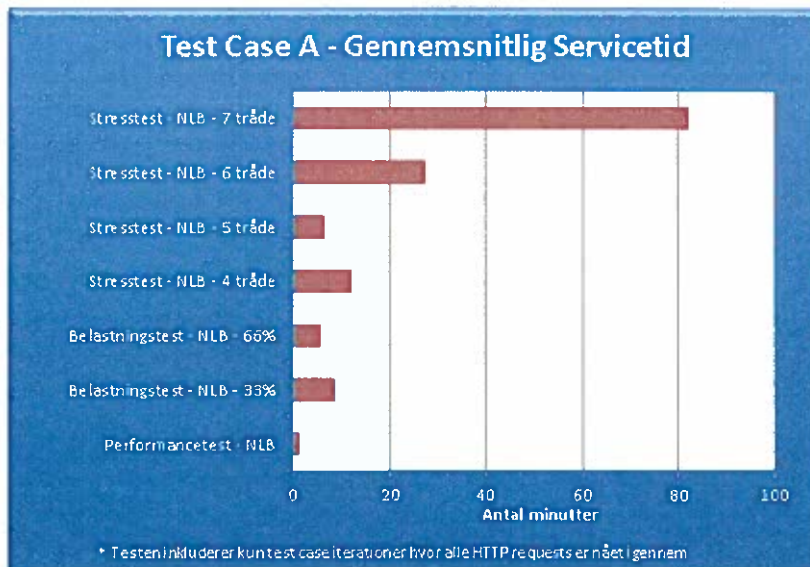
Nedenfor er en oversigt på tværs af forskellige testscenarier, hvilket har til formål at demonstrere hvor stor variation, der er i POLSAG-miljøets performance, givet både forskellige hardware og belastningsscenarier.

Test case 0 tester en af de mest basale scenarier, nemlig det at starte POLSAG op (dvs. hentning af POLSAGs startside). Dette vil normalt tage omkring 30 sekunder ved ingen eller meget lav belastning af POLSAG-miljøet, men kan nå helt op på 4 minutter, når systemet er under pres.



Som ovenstående viser, så vil servicetiden pr. iteration af test case 0 for POLSAG-miljøet generelt blive forværret betragteligt, desto mere belastning, der lægges på systemet.

Oplevelsen varierer dog også meget med test casen. Nedenfor er vist de tilsvarende tal for test case A (sagsoprettelse).



Her ses det ligeledes, at høj belastning giver en alvorlig forøgelse af svartiden. Ved en stresstest på NLB med 7 tråde, når servicetiden pr. iteration af test case A således op på over en time i gennemsnit!

Der findes i øvrigt to variationer af sagsoprettelser i POLSAG-aftestningen; "A" og "Trim-A". Førstnævnte ("A") er den sædvanlige sagsoprettelsesproces, som den foregår i POLSAG, mens den sidstnævnte ("Trim A") er en trimmet version af sagsoprettelsesprocessen, som er reduceret til det minimale antal http requests, der er nødvendige for at få POLSAG-miljøet til at oprette en sag.

Test case A består af 440 requests, mens Trim-A kun indeholder 23 requests. Trim-A bliver benyttet til den forudgående sagsoprettelse, der er nødvendig i forhold til kørsel af test cases B, C, D, E, F, H, I og K. I analyseøjemed adskilles Trim-A altid fra belastningsberegningerne for de enkelte test cases, fordi den bliver kørt før selve testen.

Det er som udgangspunkt ikke muligt, at sammenligne målinger for forskellige test cases med hinanden – eksempelvis at sammenligne test case A med test case 0 – da de requests, der bliver udført i hver test case-forløb i reglen både medfører forskellig belastning på webserverne og databasen, men også er forskellig relativt til antallet af requests og den trafik, der sendes henover netværket.

Nedenfor vises hvor mange bytes, der sendes og modtages, i de enkelte test cases:

Test Case	Bytes Sendt	Bytes Modtaget
0	265.558	5.349.814
A	425.192	1.686.392
A-Trim	29.304	40.465
B	316.831	1.491.869
C	95.980	463.104
D	351.070	852.740
E	119.921	295.221

F	128.851	226.878
G	9.424	31.435
H	358.042	519.953
I	1.047.448	1.420.208
J	128.910	1.623.117
K	818.517	2.239.840

På de følgende sider vil vi teste systemet under forskellige forhold – såvel hvad angår hardware som belastning – for at undersøge, hvor godt systemet performer og om systemet kan håndtere den belastning, Rigspolitiet forventer systemet vil blive udsat for.

Baselining – Performancetests

I dette afsnit vises performancetests, som egner sig til at få et indledende indblik i POLSAG-miljøets grundmæssigheder.

Test: Performancetest mod 1 kompleks server

Her vises resultatet for performancetesten kørt mod en kompleks server.

Denne test benyttes som baseline for den mest optimale kørsel det bør være muligt at opnå på en enkelt server, eftersom systemet slet ikke er belastet under testkørslen.

Test Case Navn	Test Case	Antal Requests	Servicetid/Iteration (minutter)
0 Initial load	0	542	0,5364333
A Grundregistrering	A	440	1,0866833
A TRIM Grundregistrering	A	23	0,2045666
B_USO_ Opret dokument	B	262	0,3972166
C_USO_ Ny sagsplacering	C	135	0,33065
D_USO_ Sigtelse	D	282	0,4841333
E_USO_ Sikret genstand	E	170	0,19595
F_USO_ Køretøj	F	108	0,08925
G Søg	G	9	0,0871666
H_USO_ Dan sagsstyring	H	333	0,4433333
I_USO_ Afslut sag	I	551	1,21235
J Træk døgnrapport	J	170	3,5459
K_USO_ Fremsøg pakke dokumenter	K	770	0,9902166

Test: Performancetest mod NLB

Her vises resultaterne for performancetesten kørt mod NLB'et. Her afvikles alle test cases samtidig (og alle Trim-A cases er som vanligt afviklet forud for testen).

Klient	Test Case	Servicetid / iteration (minutter)	Servicetid / Iteration A-Trim (minutter)	Faktiske Requests	Forventede Requests	A-Trim Faktiske Requests (af 23)	Servicetid / Request (ms)	Servicetid / Request A-Trim (ms)
X0308202	0	0,801083		542	542		88	
X0308203	A	1,050383		440	440		143	
X0308204	B	0,2931	3,735217	263	263	23	66	9744
X0308205	C	0,297883	3,63945	135	135	23	132	9494
X0308206	D	0,480363	1,8681	282	282	23	102	4873
X0308207	E	0,544483	3,3945	170	170	23	192	8855
X0308208	F	0,2697	2,786533	108	108	23	149	7269
X0308209	G	0,347333		9	9		2315	
X0308210	H	0,92955	2,215533	333	333		167	5779
X0307212	I	3,0518	1,448583	551	551	23	332	3778
X0308211	J	8,510683		170	170		1501	
X0307213	K	5,388017	2,982983	771	771	23	419	7781

Der bemærkes specielt at Trim-A tager længere tid at afvikle end test case A i denne test. Dette formodes at skyldes der bliver oprettet 8 sager parallelt (dvs. alle Trim-A'er bliver kørt på samme tid umiddelbart forud for selve performancetesten), hvorimod der kun oprettes en sag i form af test case A under performancetesten. Tallene viser som sådan at sagsoprettelser bliver markant langsommere, når man forsøger at oprette flere sager på samme tid fremfor en enkelt sag ad gangen.

Der gøres opmærksom på at test case J har vist sig at variere meget med forskellige afviklingsforhold. Der er ikke udført en undersøgelse af de mulige årsager til de meget forskellige resultater for de to performancetests.

Belastningstests

Belastningstesten er den vigtigste test, da den dækker over alle use cases, der bliver afviklet baseret på den test case-plan, der beskriver belastningen i en kreds mandag formiddag.

Belastningstesten vurderes at repræsentere et konservativt skøn over den normale belastning på POLSAG mandag formiddag, idet POLSAG jo også bliver brugt til at udføre andre arbejdsgange end dem, der er repræsenteret i de 12 test cases.

Den opstillede model yder som sådan ikke nogen garanti for at POLSAG vil opføre sig *så godt* som aftestningen angiver. Omvendt må det formodes at POLSAG i hvert fald ikke opfører sig bedre end hvad aftestningen indikerer. Såfremt POLSAG-miljøet er i stand til at levere fornuftige svartider ved den angivne belastning samt at scale out-testen demonstrerer at der er en vis luft til det POLSAG-miljøets performancemæssige loft, så synes POLSAG altså at være klar til mere intensiv aftestning forud for en eventuel pilotudrulning i den næste kreds i produktionsmiljøet.

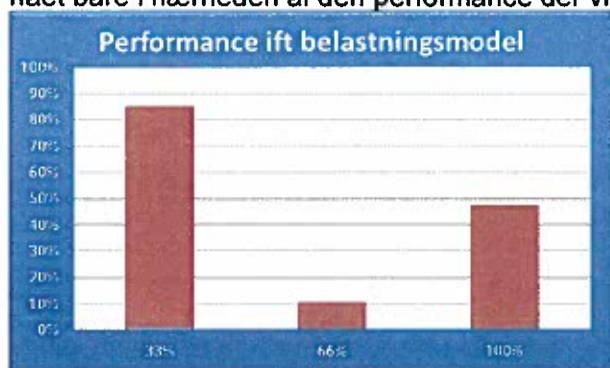
De primære formål med belastningstesten er at afdække POLSAGs opførsel under normal belastning:

- Hvordan systemet opfører sig under den forventede belastning.
- At afdække hvilke kritiske aktiviteter, der belaster systemet mest, herunder afdække om der er ikke-kritiske aktiviteter, der belaster systemet unødigt meget.

Test: Belastningstest mod NLB

Belastningstesten består rent praksis i at køre med tre forskellige belastningsgrader for at se, hvordan systemet håndterer gradvist højere belastning. Der startes ved 33%, som må ses som forholdsvist begrænset belastning af systemet, og skiftes til 66% og afsluttes ved 100%, som svarer til Rigspolitiets model for belastning i en kreds fordelt over de 12 test cases.

Som det ses af nedenstående generelle oversigt, er det kun ved 33% belastning, at systemet er nået bare i nærheden af den performance der vil være nødvendig.

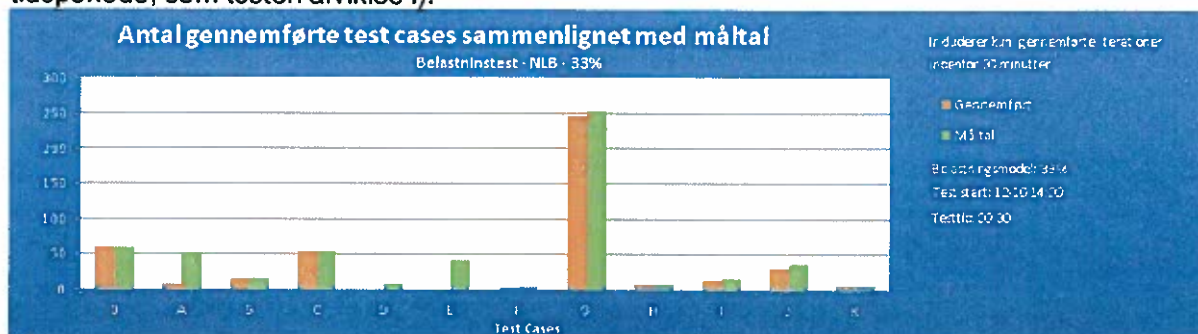


Bemærk, at 66%-belastningstesten såvel som 100%-belastningstesten får POLSAG-miljøet til at bryde sammen. Miljøet bryder dog sammen på forskellige måder, hvilket er årsagen til at 66%-testen på papiret ser ud til at klare sig endnu dårligere end 100% belastning.

Aftestningerne demonstrerer at det er umuligt, at komme helskindet igennem en belastningstest med en belastning på over 33%, idet testkørsler fejler hver eneste gang. Kapitlet "Belastningstests - Problemoveersigt" gennemgår nogle af de problemer, vi er stødt på i forbindelse med afviklingen af belastningstestene.

Da belastningsmodellens måltal er defineret som de test cases der kan køres færdige indenfor en specifikt afgrænset tidsperiode (testperioden), så ser vi bort fra iterationer som måtte slutte efter tidsperiodens udløb i vore analyser. For belastningstests anvendes der generelt altid en testperiode på en halv time.

I nedenstående oversigt ses hvor langt systemet nåede med de enkelte test cases i forhold til testens måltal (dvs. det antal iterationer, som den ideelt set bør nå at gennemføre indenfor den tidsperiode, som testen afvikles i).



Ved 33% belastning når POLSAG-miljøet at gennemføre de fleste test cases.

Der er dog nogle få undtagelser for dette udsagn:

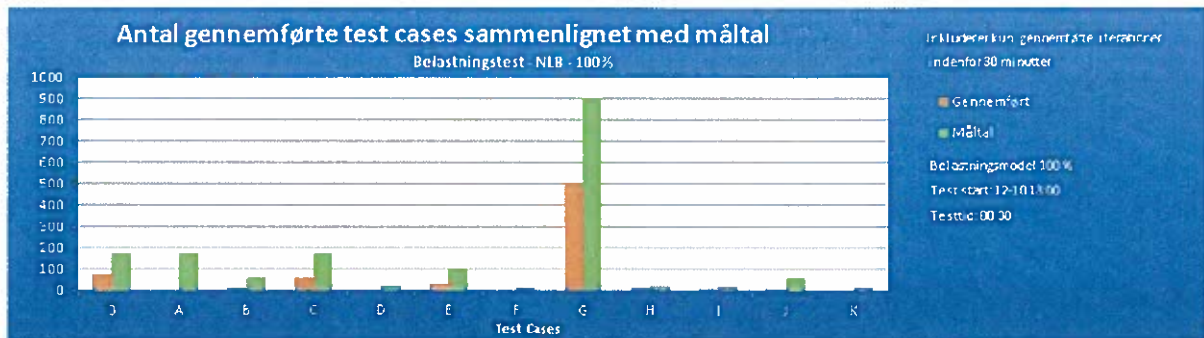
- Et meget stort antal af test case A når ikke at blive gennemført indenfor testperioden, da de http requests der laver sagsoprettelsen timer ud (i enkelte tilfælde er der noteret over en 1 times svartid på det sidste request, hvori sagsoprettelsen bliver udført)
- Test case E når ikke i nærheden af de opsatte måltal.
- For test case D når der heller ikke at blive gennemført nogen iterationer.

Ved 66% og 100% belastning mod NLB går testen generelt helt gået galt, da systemet simpelthen ikke er i stand til at håndtere den højere belastning.

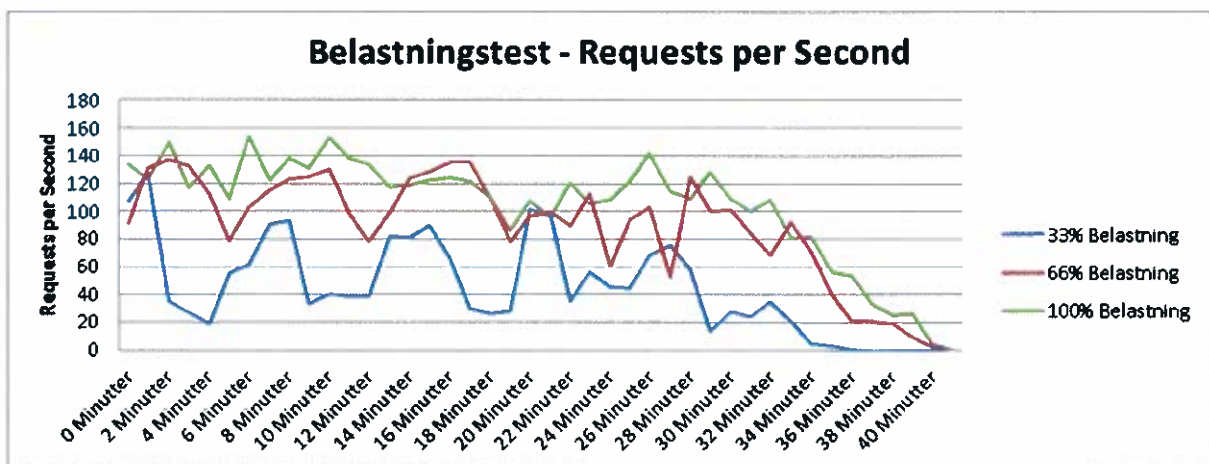
I et senere kapitel ("Belastningstests - Problemoveksigt") går vi lidt mere i dybden med med hvad der præcist er sket og forsøge at analysere nogle af de 66 og 100% tests der er fejlet.

Vi har dog en færdiggjort 100% belastningstest mod NLB, som vi i det følgende vil bruge for at demonstrere vores oplevelser ved højere belastninger. Det skal dog understreges, at systemet generelt holder op med at opføre sig korrekt ved den højere belastning, hvorfor performancetallene ikke er troværdige.

Ved 100% belastning ser vi tydeligt, at systemet ganske enkelt ikke kan opfylde den forventede belastningsmodel. Generelt køres næsten ingen af de specificerede test cases til ende og som bruger ville situationen i al praksis opleves som et totalt systemnedbrud på grund af svartiden.



Af grafen herunder ses endvidere, at der stort set ikke flere requests igennem ved 100% belastning sammenlignet med 66%, hvilket tyder på, at systemet ikke kan skalere yderligere allerede på dette niveau. Denne test kører som nævnt mod NLB'et i POLSAG-uddannelsesmiljøet.



I de efterfølgende afsnit gennemgår vi resultaterne af de enkelte test cases under afviklingen af belastningstesten. Som nævnt gør vi opmærksom på at tallene for 100%-testen ikke kan betragtes som fuldt repræsentative, idet POLSAG-miljøet går ned og det således er en brøkdel af operationerne, der lykkes indenfor testperioden. Alene begrænsningerne for såvidt angår antallet af leverede requests per second-tider indikerer at Rigspolitiet ikke vil kunne benytte POLSAG som tilsigtet, ligesom at risikoen for at miljøerne går ned peger på at det vil være uforsvarligt at tage POLSAG i anvendelse.

Analyse af anvendelse af randsystemer i de enkelte test cases

Dette afsnit dokumenterer den belastning som de enkelte testcases påfører randsystemerne (og dermed ESB'en).

Resultaterne er baseret på informationerne i den såkaldte SOAPLog, som er implementeret i POLSAG. Alle resultater er fra uddannelsesmiljøet og indhentet ved separat afvikling af hver enkelt test case.

Nedenstående tabel viser antallet af kald som registreres i SOAPLog.

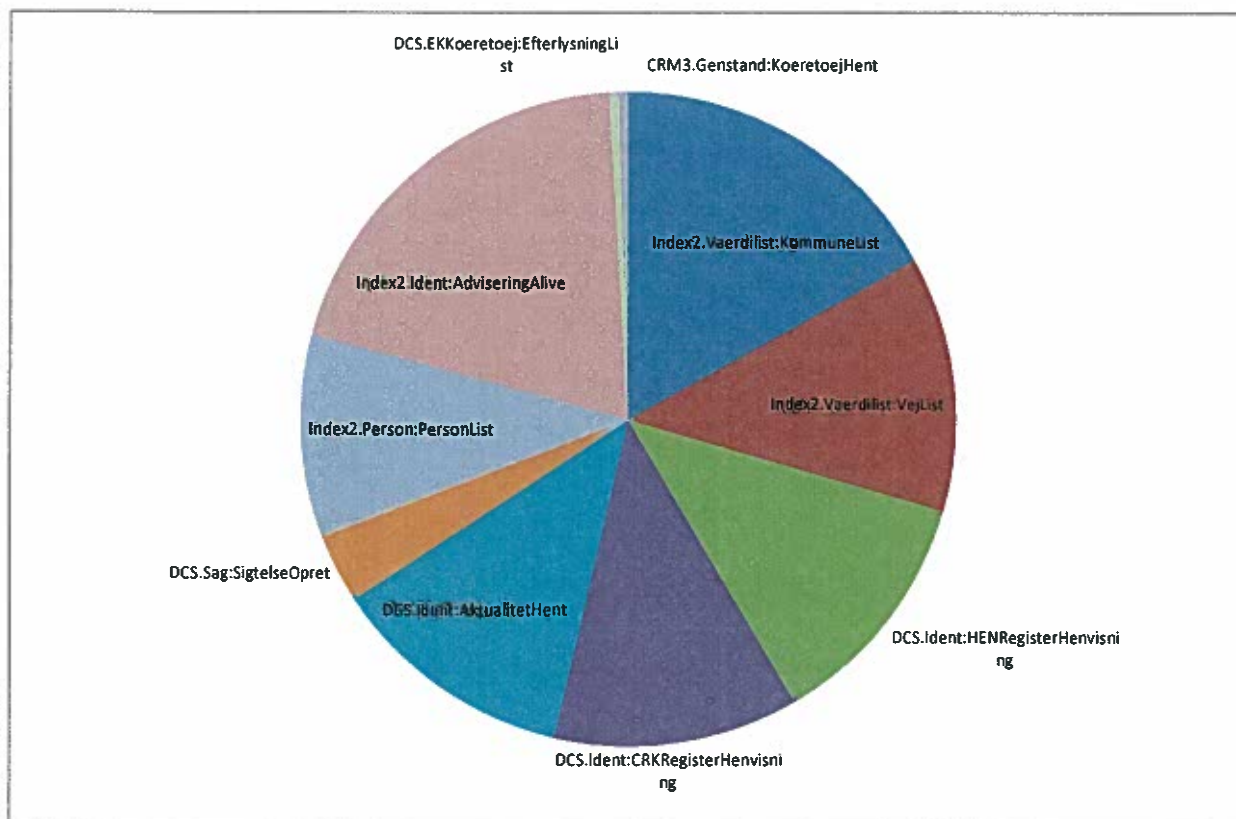
Der gøres opmærksom på at der er noteret relativt store udsving i det præcise tidsforbrug til randsystemkaldene i SOAPLog'en. Det vides ikke om det skyldes en begrænset præcision på SOAPLog'en eller om der reelt kan være store forskelle på randsystemkaldenes svartid. Det skal yderligere bemærkes at tiderne som nævnt stammer fra uddannelsesmiljøet, hvor integrationsplatformen og randsystemerne såvidt vi forstår ikke er så kraftigt bestykket som i produktionen.

Tabellen forneden viser antallet af kald til randsystemerne for de enkelte test cases.

Test case	A	D	F	H	I	K
Index2.Vaerdilist:KommuneList	2	1	0	1	1	3
Index2.Vaerdilist:VejList	2	0	0	0	0	2
DCS.Ident:HENRegisterHenvisning	1	2	0	2	1	2
DCS.Ident:CRKRegisterHenvisning	1	2	0	2	1	2
DCS.Ident:AktualitetHent	1	2	0	2	1	2
DCS.Sag:SigttelseOpret	0	1	0		1	1
Index2.Person:PersonList	1	1	0	1	1	1
Index2.Ident:AdviseringAlive	2	2	0	2	2	2
DCS.EKKoeretoej:EfterlysningList	0	0	1	0	0	0
CRM3.Genstand:KoeretoejHent	0	0	1	0	0	0
Gennemsnitligt tidsforbrug pr. test case (ms)	4509	3419	2640	2712	2450	7277
Total servicetid (ms)	140344	62837	21535	58141	213286	304498
Randsystemers andel af servicetid	3.2%	5.4%	12,3%	4,7%	1,1%	2,4%

I det følgende er vist en række tabeller og figurerer, som dokumenterer antallet af kald til randsystemerne, som skal gennemføres, når belastningsmodellen kører i en time.

Den relative fordeling af antallet af kald til randsystemerne er som følger:



Det skal bemærkes at 30% af kaldene til randsystemerne går til Kommune og vejliste (dvs. hentning af kommune- og vejliste via Index2) og at 20% af kaldene går til en særlig "PING"-funktion i Index2 (før Index2 kaldes checker POLSAG at den er i live ved et såkaldt PING-kald). Det vurderes at være en helt oplagt optimering af POLSAG, hvis disse kald henholdsvis caches i en lokal database og elimineres.

Tabellen viser en beregning af antallet af kald til randsystemerne for en times kørsel af belastningsmodellen:

Test case	A	D	F	H	I	K
Index2.Vaerdilist:KommuneList	600	40	0	40	110	90
Index2.Vaerdilist:VejList	600	0	0	0	0	60
DCS.Ident:HENRegisterHenvisning	300	80	0	80	110	60
DCS.Ident:CRKRegisterHenvisning	300	80	0	80	110	60
DCS.Ident:AktualitetHent	300	80	0	80	110	60
DCS.Sag:SigtelseOpret	0	40	0	0	110	30
Index2.Person:PersonList	300	40	0	40	110	30
Index2.Ident:AdviseringAlive	600	80	0	80	220	60

DCS.EKKoeretoej:EfterlysningList	0	0	25	0	0	0
CRM3.Genstand:KoeretoejHent	0	0	25	0	0	0

Tabellen viser antallet af kald til randsystemerne for en times kørsel af belastningsmodellen beregnet for hele POLSAG (12 gange antallet af kald fra belastningsmodellen):

Test case	A	D	F	H	I	K
Index2.Vaerdilist:KommuneList	7200	480	0	480	1320	1080
Index2.Vaerdilist:VejList	7200	0	0	0	0	720
DCS.Ident:HENRegisterHenviisning	3600	960	0	960	1320	720
DCS.Ident:CRKRegisterHenviisning	3600	960	0	960	1320	720
DCS.Ident:AktualitetHent	3600	960	0	960	1320	720
DCS.Sag:SigtelsetOpret	0	480	0	0	1320	360
Index2.Person:PersonList	3600	480	0	480	1320	360
Index2.Ident:AdviseringAlive	7200	960	0	960	2640	720
DCS.EKKoeretoej:EfterlysningList	0	0	300	0	0	0
CRM3.Genstand:KoeretoejHent	0	0	300	0	0	0

Der er ikke gennemført en egentlig performance/belastningstest af randsystemerne. Det må således anbefales at man sikrer sig at ESB'en er dimensioneret til at håndtere de 20 requests pr. sekund, som POLSAG formodes at ville belaste randsystemerne (og dermed ESB'en) med.

Vi har udtaget SOAPlogs fra kørslen af henholdsvis 33% og 100% belastningstesten, som ikke viser nogen signifikante ændringer i relation til randsystemernes performance relativt til tallene ovenfor. Dvs. randsystemerne er tilsyneladende ikke en medvirkende faktor til den oplevede nedsættelse af POLSAGS performance under belastning.

Der er ikke blevet gennemført yderligere analyser af SOAPlogs på grund af disse resultater.

Det skal samtidig bemærkes at randsystemerne (og dermed ESB'en) ikke har været belastet med mere end ca. ½ requests pr. sekund grundet vanskelighederne med at køre 100% belastningstesten succesfuldt igennem, hvorfor der p.t. ikke foreligger information om, hvordan disse vil være i stand til at håndtere 100% belastning, der forventes at udløse en belastning på ca. 1½ requests pr. sekund.

For kompletthed vil vi gøre opmærksom på at de tests som er udført som en del af denne analyse kun giver indblik i billedet af randsystemernes indvirkning på POLSAGS performance i relation til de benyttede test cases. Da vi undersøgte SOAPlogs fra uddannelsesmiljøet stødte vi således ind i et par andre eksempler på randsystem-anvendelser, som stammer fra brugen af POLSAG til

uddannelse (og således ikke bliver anvendt at vores test). Her optræder der nogle kald til randsystemerne med væsentligt længere svartider end observeret i de benyttede testcases:

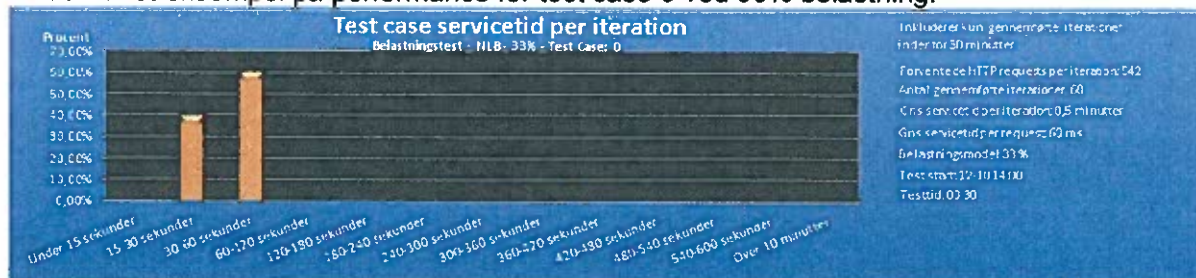
Navn på operation	Tidsforbrug (ms)
POLSAG.Ident:SekundaereNavneOpdater	11775
POLSAG.Ident:KontaktoplysningerOpdater	11697

Det bør bemærkes at dette er opdateringer til randsystemerne, ligesom at svartider fra randsystemerne i uddannelsesmiljøet formentligt skal omgås med forsigtighed, da tidligere sammenligninger mellem uddannelsesmiljøet og produktionsmiljøet indikerer at randsystemerne i uddannelsesmiljøet svarer noget langsommere end i produktionsmiljøet.

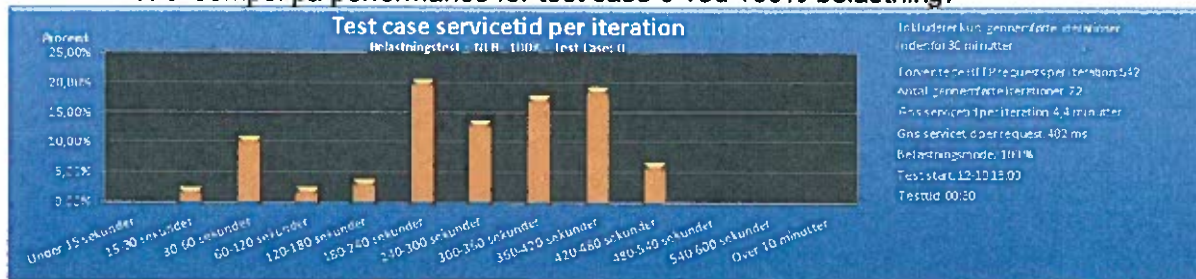
Belastningstest mod NLB – Test Case 0

Test case 0 for 'Åbn forsiden'.

Nedenfor et eksempel på performance for test case 0 ved 33% belastning.



Nedenfor et eksempel på performance for test case 0 ved 100% belastning.



Analyse

Test casen har opnået et rimeligt antal succesfulde iterationer ved både 33 og 100% belastning.

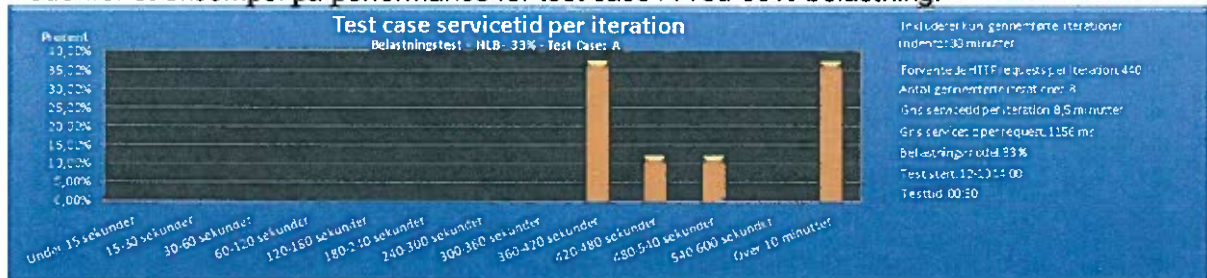
Det er tydeligt at se at responstiden forværres kraftigt ved højere belastning. Den gennemsnitlige tid hvert request tager øges med 800% hvis man sammenligner 33% belastning med 100%.

Nogle brugere ville ved 100% belastning skulle vente helt op til 8 minutter for at se forsiden. Reelt set er det dog meget få brugere der med succes vil fået loadet forsiden.

Belastningstest mod NLB – Test Case A

Test case A for 'Grundregistrering'.

Nedenfor et eksempel på performance for test case A ved 33% belastning.



Nedenfor et eksempel på performance for test case A ved 100% belastning.



Analyse

Test Case A er en af de cases, hvor der noteres de væsentligste problemer. Som det fremgår af illustrationen, så vil der ganske enkelt ikke blive oprettet sager, hvis systemet er under høj belastning.

Den gennemsnitlige tid for sagsoprettelsesoperationen vil ved 33% belastning være over 8 minutter. Det væsentligste problem er, at der ved de fleste iterationer er enkelte requests, der ganske enkelt ikke når igennem og timer ud efter ca. 1 times ventetid.

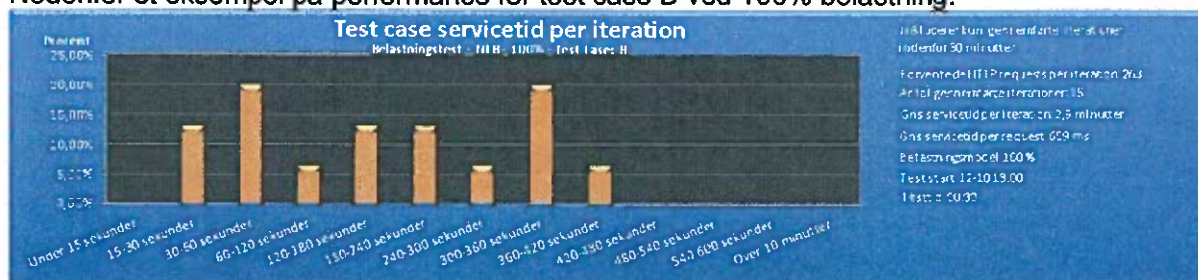
Belastningstest mod NLB – Test Case B

Test case B for 'Opret dokument'.

Nedenfor et eksempel på performance for test case B ved 33% belastning.



Nedenfor et eksempel på performance for test case B ved 100% belastning.



Analyse

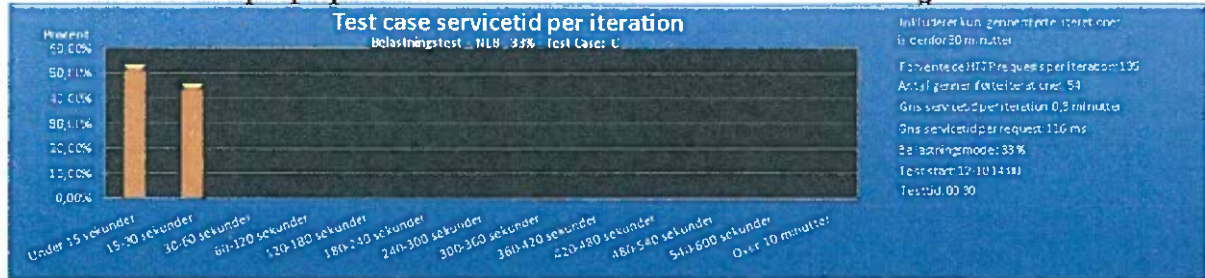
Test case B viser med tydelighed den forringelse af responstider som brugere ville opleve, når POLSAG-miljøet udsættes for højere belastninger.

Som nævnt er der en lang række operationer, der enten fejler eller ikke når igennem indenfor testperioden ved 100% belastning.

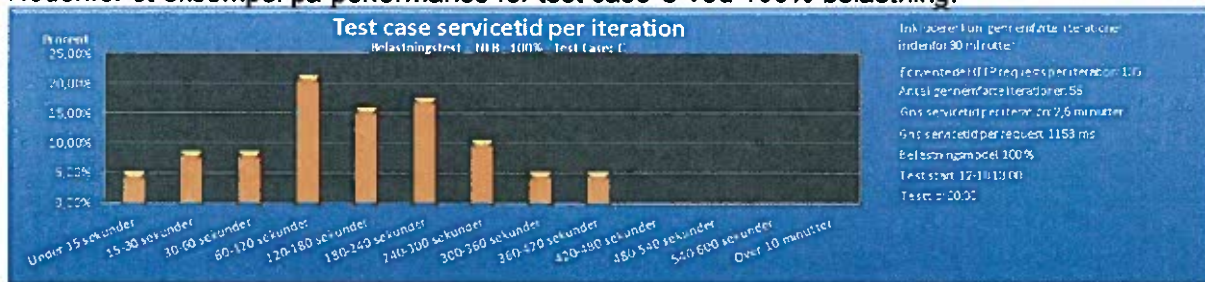
Belastningstest mod NLB – Test Case C

Test case C for 'Sagsplacering'.

Nedenfor et eksempel på performance for test case C ved 33% belastning.



Nedenfor et eksempel på performance for test case C ved 100% belastning.



Analyse

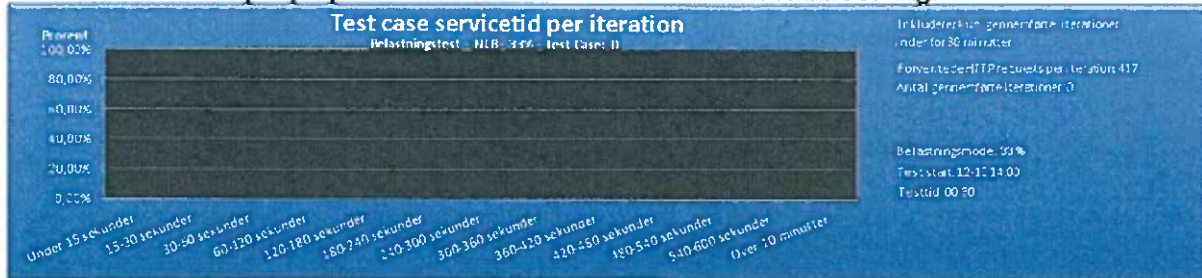
Også test case C er endnu et eksempel på den meget forringede hastighed som man oplever på POLSAG-miljøet ved højere belastning.

Som nævnt er der en lang række operationer, der enten fejler eller ikke når igennem indenfor testperioden ved 100% belastning.

Belastningstest mod NLB – Test Case D

Test case D for 'Sigtelse'.

Nedenfor et eksempel på performance for test case D ved 33% belastning.



Nedenfor et eksempel på performance for test case D ved 100% belastning.



Analyse

Test case D er sårbar overfor enhver form for belastning. Selv ved 33% har vi ikke registreret succesfulde iterationer.

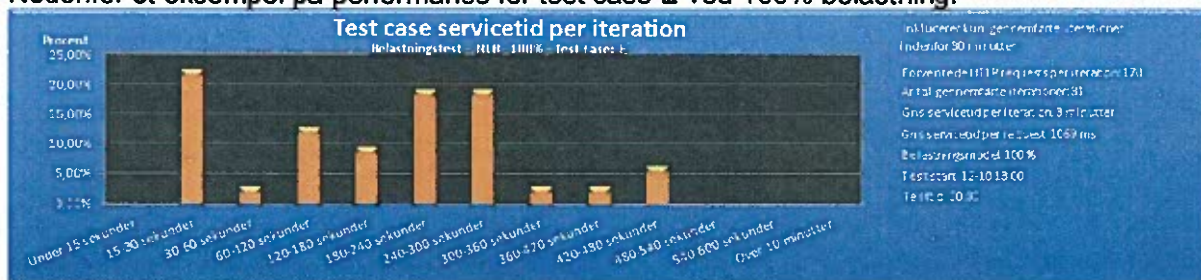
Belastningstest mod NLB – Test Case E

Test case E for 'Sikret genstand'.

Nedenfor et eksempel på performance for test case E ved 33% belastning.



Nedenfor et eksempel på performance for test case E ved 100% belastning.



Analyse

Ved test case E oplever vi lidt et særtilfælde, hvor test casen faktisk performer lidt bedre ved den høje belastning (dvs. der er nogle få requests der når igennem indenfor testperioden). Indtil videre har vi henført dette til at der er tre gange større mulighed for at en enkelt operation lykkes, når man sender tre gange flere operationer afsted til POLSAG-miljøet, da vi ikke forventer at højere belastning reelt set er fordelagtig for test case E.

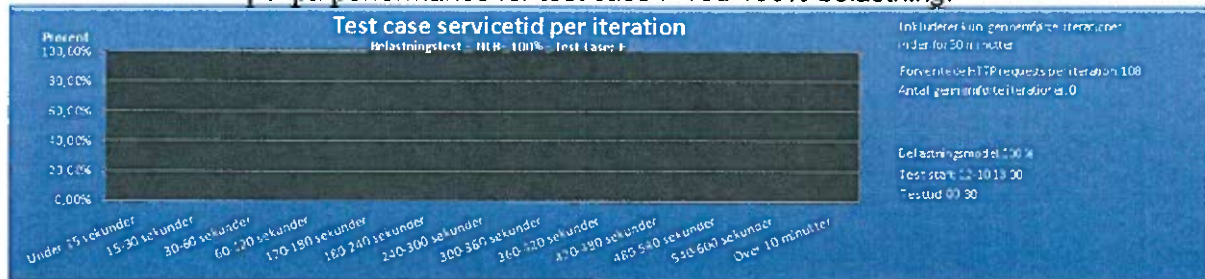
Belastningstest mod NLB – Test Case F

Test case F for 'Køretøj'.

Nedenfor et eksempel på performance for test case F ved 33% belastning.



Nedenfor et eksempel på performance for test case F ved 100% belastning.



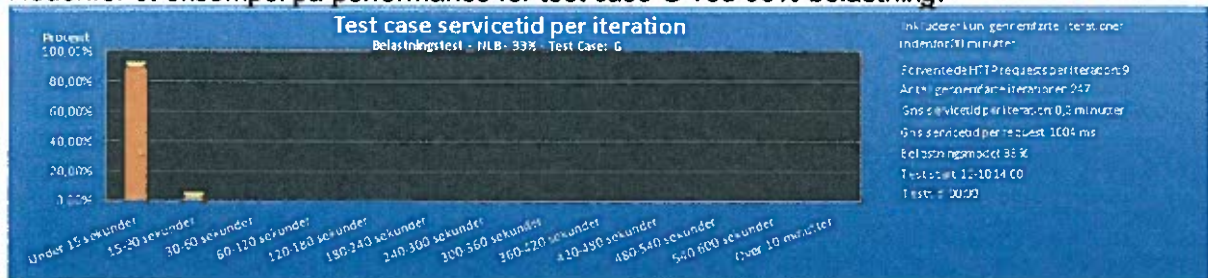
Analyse

Det er kun ganske få test cases, der er kørt igennem ved test case F og kun ved 33% belastning. De test cases, der når igennem, kommer til gengæld relativt hurtigt og smertefrit i mål.

Belastningstest mod NLB – Test Case G

Test case G for 'Søg'.

Nedenfor et eksempel på performance for test case G ved 33% belastning.



Nedenfor et eksempel på performance for test case G ved 100% belastning.



Analyse

Ved både 33 og 100% belastning lykkes det POLSAG-miljøet at udføre et relativt stort antal af denne "lette" test case. Igen kan man tydeligt aflæse konsekvensen af den højere belastning i svartiderne.

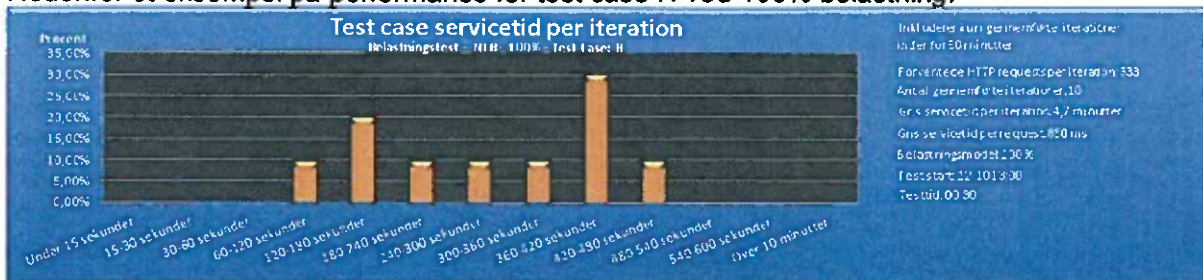
Belastningstest mod NLB – Test Case H

Test case H for 'Dan sagsstyring'.

Nedenfor et eksempel på performance for test case H ved 33% belastning.



Nedenfor et eksempel på performance for test case H ved 100% belastning.



Analyse

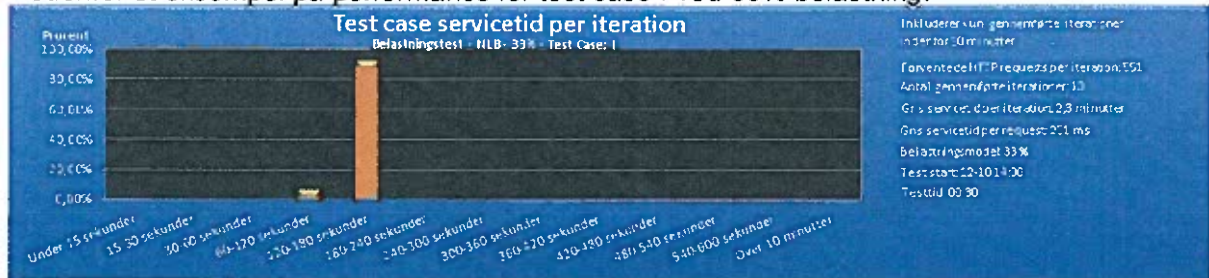
Test case H kører endog meget stabilt og ensartet igennem ved 33% belastning.

Men også test case H demonstrerer endnu engang et eksempel på konsekvensen af højere belastning. Den gennemsnitlige servicetid per iteration stiger således fra 36 sekunder til over 4 minutter.

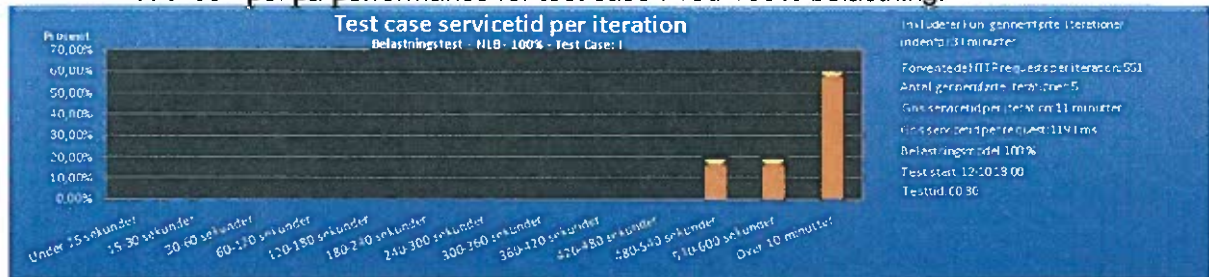
Belastningstest mod NLB – Test Case I

Test case I for 'Afslut sag'.

Nedenfor et eksempel på performance for test case I ved 33% belastning.



Nedenfor et eksempel på performance for test case I ved 100% belastning.



Analyse

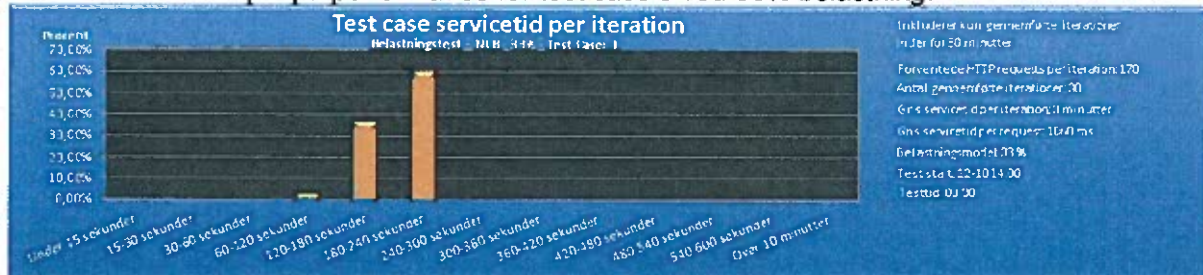
Test case I kører stabilt og ensartet igennem ved 33% belastning.

Test case I demonstrerer tydeligt, hvor meget langsommere systemet bliver ved den højere belastning.

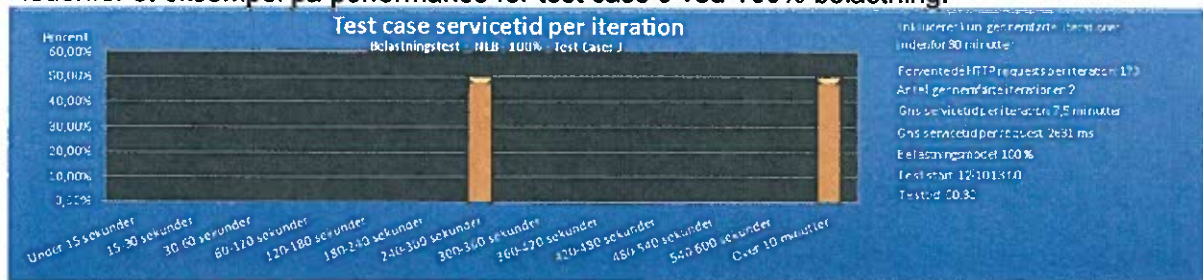
Belastningstest mod NLB – Test Case J

Test case J for 'Træk døgnrapport'.

Nedenfor et eksempel på performance for test case J ved 33% belastning.



Nedenfor et eksempel på performance for test case J ved 100% belastning.



Analyse

Ved 33% belastning virker gennemførelsesprocenten og svartiderne i relation til træk af døgnrapporter fornuftige henset til baseline.

Ved 100% belastning er der til gengæld kun to iterationer der lykkes og servicetiden per iteration for disse er langt højere end ved 33% belastning.

Det formodes at årsagen til de bedre svartider for døgnrapporterne end ved performancetesten skyldes at der sker en caching af resultaterne på Oracle og den således er i stand til at besvare de efterfølgende træk af døgnrapporter væsentligt hurtigere end det der er tilfældet ved første request.

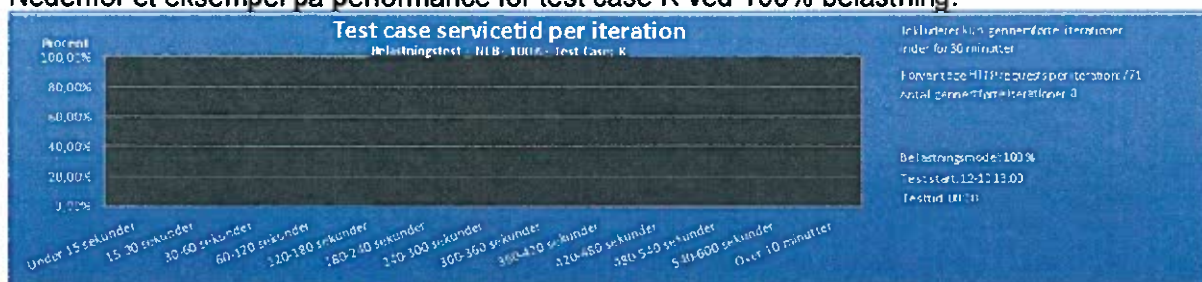
Belastningstest mod NLB – Test Case K

Test case K for 'K_USO_ Fremsøg pakke dokumenter'.

Nedenfor et eksempel på performance for test case K ved 33% belastning.



Nedenfor et eksempel på performance for test case K ved 100% belastning.



Analyse

Her er der igen et eksempel på, at POLSAG-miljøet ved højere belastning ganske enkelt ikke magter at få kørt nogen tests cases igennem indenfor testperioden.

Belastningstest mod NLB – Serverperformance

Præ-test fasen hvor Trim-A oprettes vises i den orange boks, og den røde boks viser perioden hvor hovedtesten køres.

Her test perioden illustreret for 33% belastning for server RPLPOLSAGEE914V.



Og for server RPLPOLSAGEE901V.

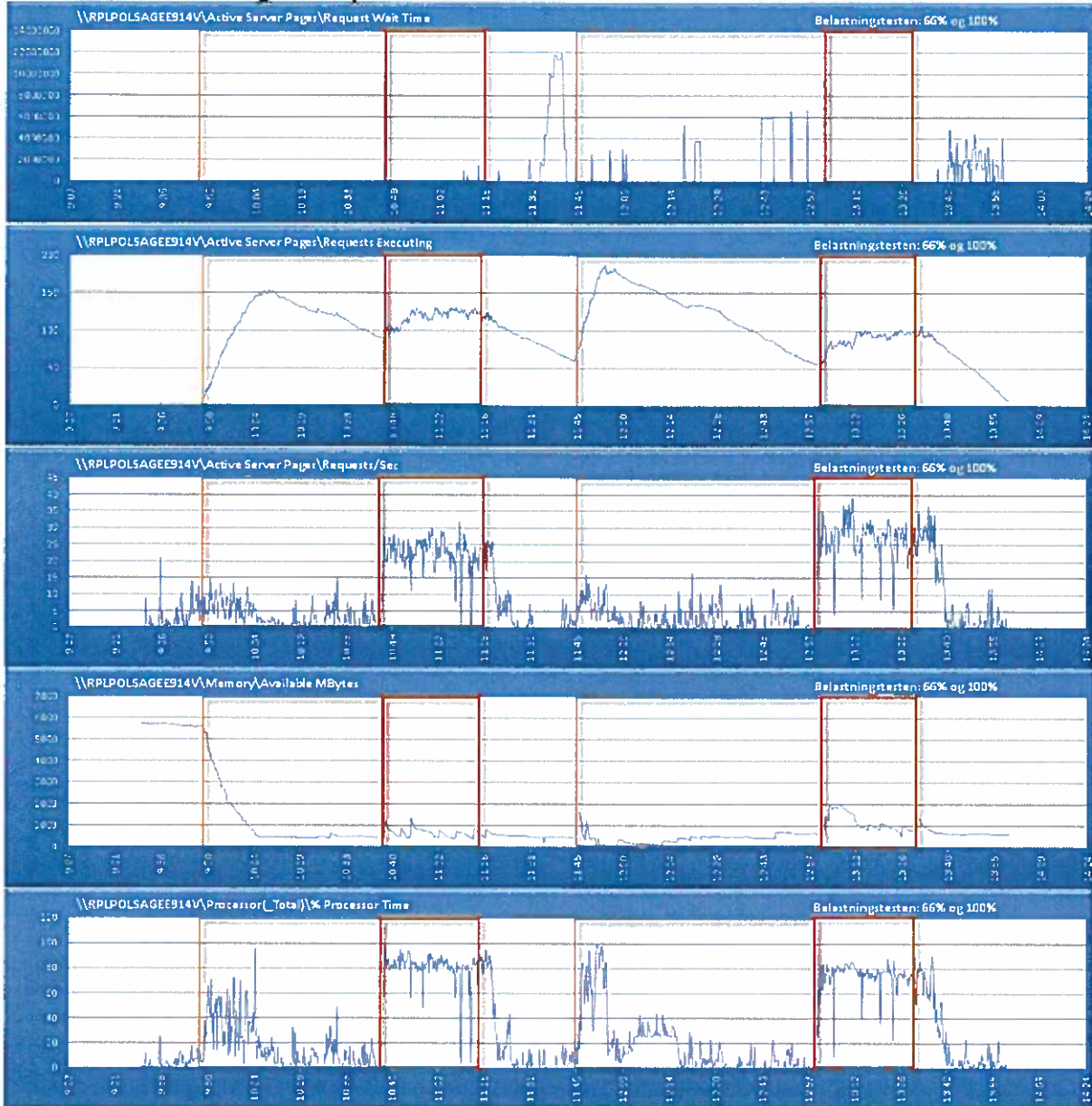


Samt for server RPLPOLSAGEE911V.

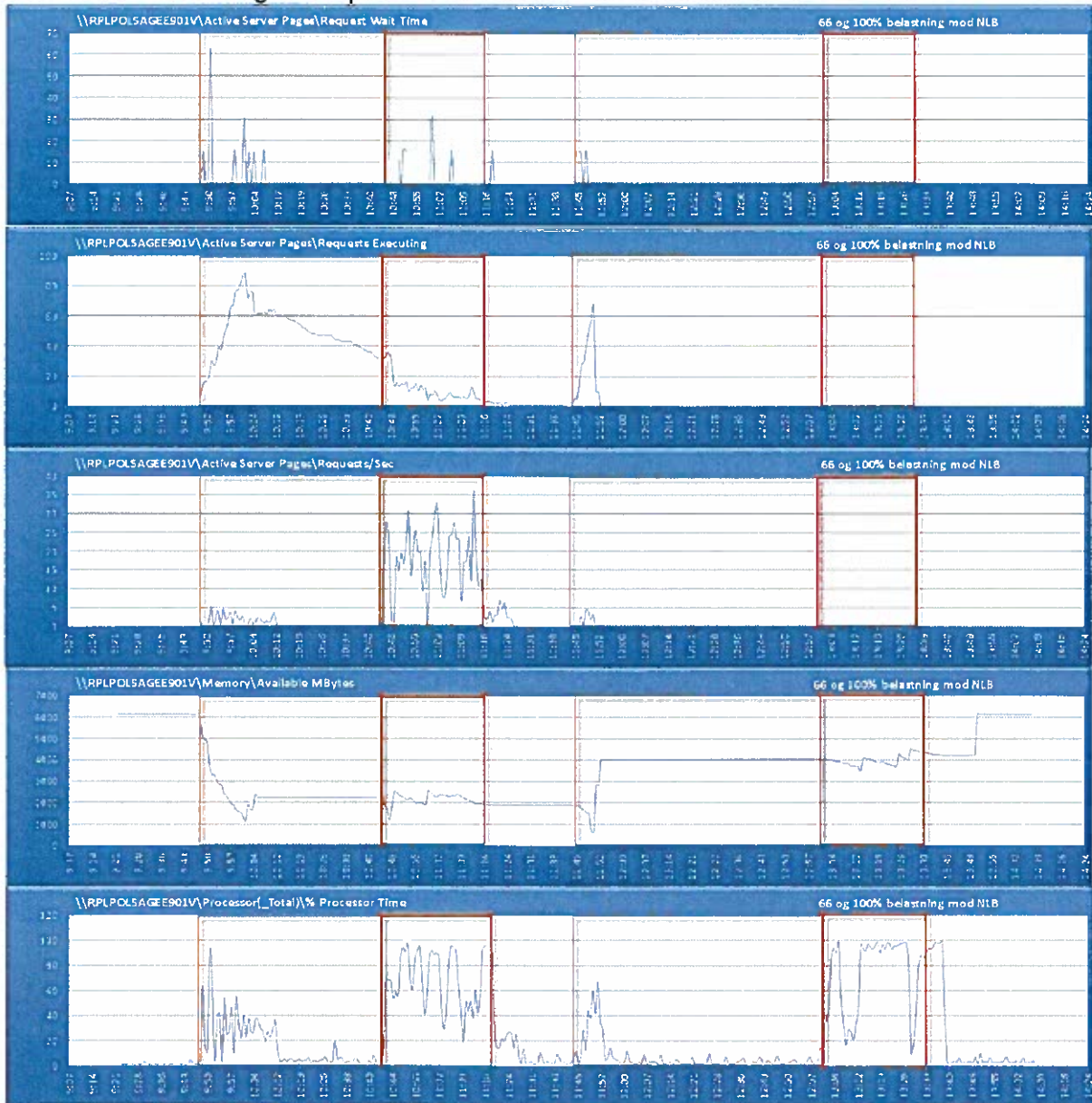


I de nedenstående tabeller vise vi både testkørslen for belastning 66 og 100% i samme oversigt, da de blev kørt samme dag. 66% belastning er illustreret af de to farvede bokse på venstre side og 100% de to bokse på højre side.

Testkørslen for 66% og 100% på server RPLPOLSAGEE914V.



Testkørslen for 66% og 100% på server RPLPOLSAGEE901V.



Testkørslen for 66% og 100% på server RPLPOLSAGEE911V.



Det er tydeligt at belastningen allerede ved 66% presser systemet helt i bund.

Graferne for de tre servere er nogenlunde ens, hvilket indikerer at NLB'et er godt til at sprede belastningen og at POLSAG-miljøet som sådan yder så godt som det kan i testen.

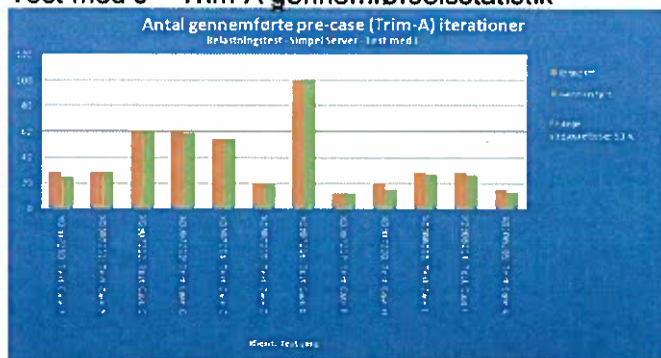
Belastningstest mod 1 simpel server

I et forsøg på at afdække, hvordan POLSAGs ydelse varierer med hardware har vi endvidere forsøgt at afvikle den fulde belastningsmodel mod en simpel server.

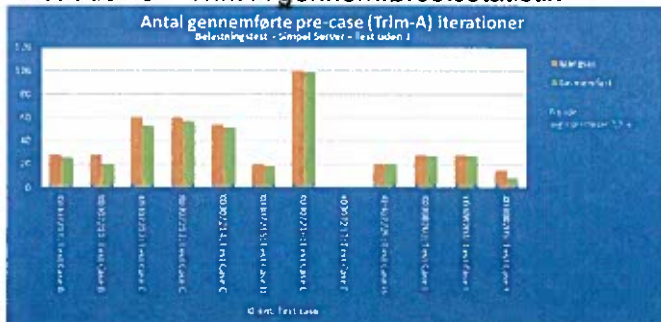
Da test case J kan være hård mod serveren har vi udført testforløbet både med og uden denne test case. Vi vil i de følgende afsnit referere til disse som "Test med J" og "Test uden J".

Forud for testene er der som vanligt blevet kørt Trim-A cases, som helt generelt er forløbet uden for store vanskeligheder.

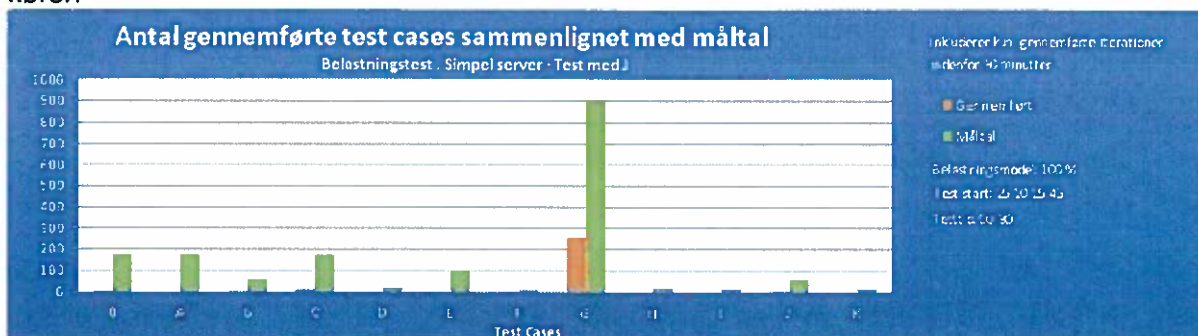
Test med J – Trim-A gennemførelsesstatistik



Test uden J – Trim-A gennemførelsesstatistik

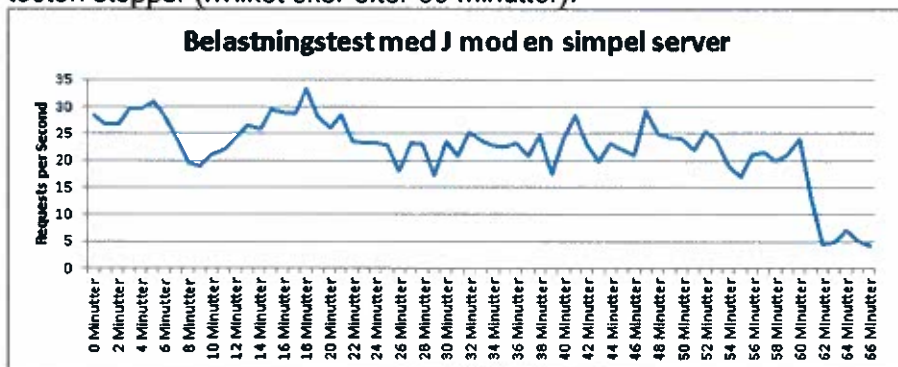


I nedenstående oversigt vises, hvor langt systemet nåede med de enkelte test cases, i forhold til givne måltal. Måltallene er udregnet i forhold til det forventede antal på den halve time, som testen kører.

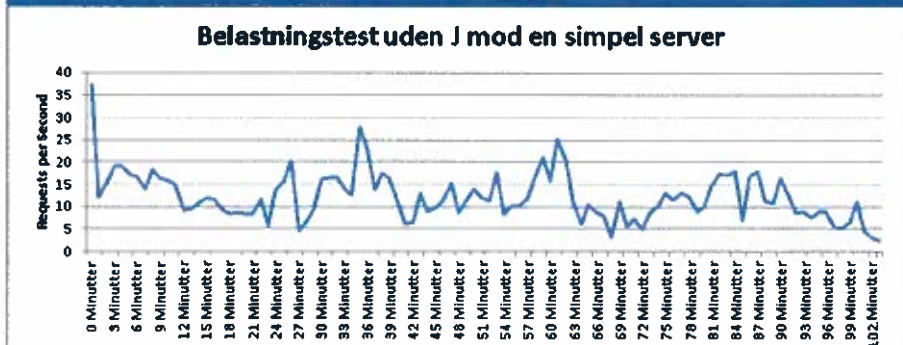
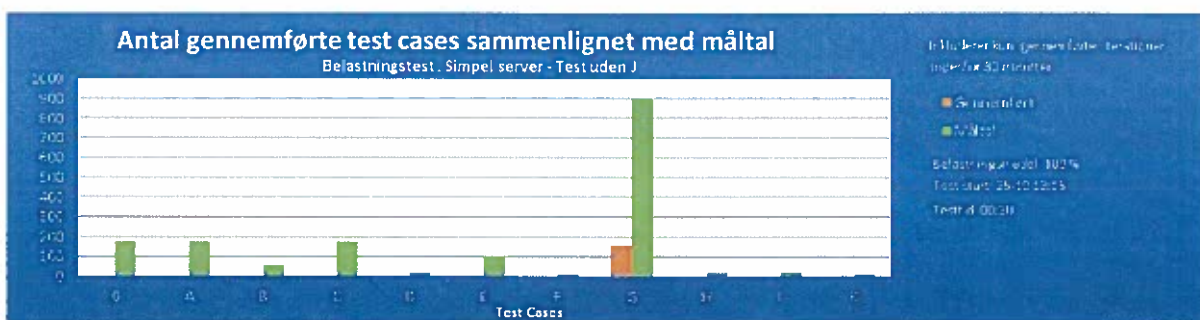


Testen er således efter en halv time ikke nærheden af at nå måltallene.

Givet de resultater der er opnået mod NLB var dette dog ikke overraskende. Herudover ser vi også på nedenstående diagram, hvordan serveren er belastet med udeståender fra testen lang tid efter testen stopper (hvilket sker efter 30 minutter).



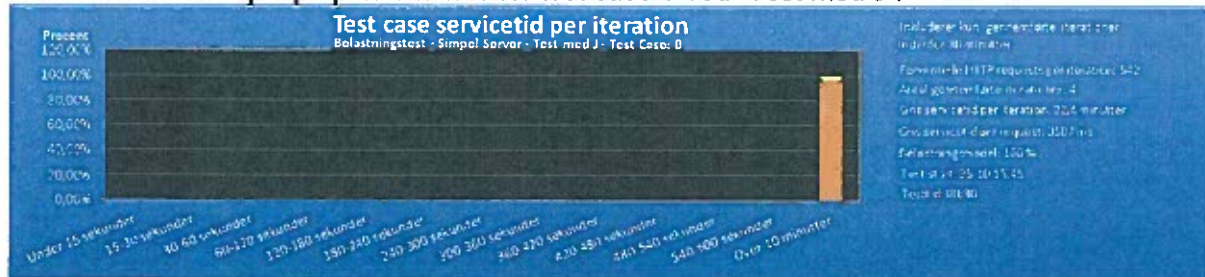
Uden test case J opnås der et lignende resultat; blot er systemet gået i endnu mere hårdknude med lavere requesthåndtering og fortsætter med at køre i endnu længere tid efter testen er afsluttet.



Belastningstest mod 1 simpel server – Test case 0

Test case 0 for 'Åbn forsiden'.

Nedenfor et eksempel på performance for test case 0 ved "Test med J".



Nedenfor et eksempel på performance for test case 0 ved "Test uden J".



Analyse

Det er tydeligt ved denne test case at serveren ganske enkelt ikke kan håndtere requests på en fornuftig vis. Det er kun et absolut fåtal af iterationerne, der kommer igennem, og der opleves yderst lange servicetider for disse.

Belastningstest mod 1 simpel server – Test case B

Test case B for 'Opret dokument'.

Nedenfor et eksempel på performance for test case B ved "Test med J".



Nedenfor et eksempel på performance for test case B ved "Test uden J".



Analyse

Det er tydeligt ved denne test case at serveren ganske enkelt ikke kan håndtere requests på en fornuftig vis. Det er kun et absolut fåtal af iterationerne, der kommer igennem, og der opleves yderst lange servicetider for disse.

Belastningstest mod 1 simpel server – Test case D

Test case D for 'Sigtelse'.

Nedenfor et eksempel på performance for test case D ved "Test med J".



Nedenfor et eksempel på performance for test case D ved "Test uden J".



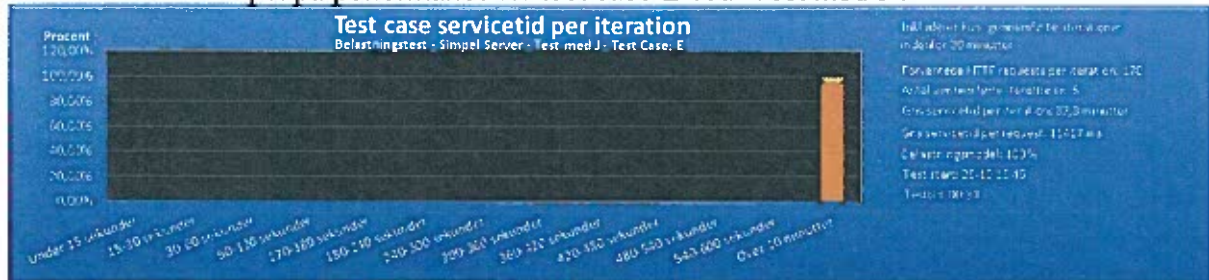
Analyse

Her færdiggøres ingen test cases, hvilket viser at denne hardwarekonfiguration ganske enkelt ikke er i stand til at håndtere belastningen.

Belastningstest mod 1 simpel server – Test case E

Test case E for 'Sikret genstand'.

Nedenfor et eksempel på performance for test case E ved "Test med J".



Nedenfor et eksempel på performance for test case E ved "Test uden J".

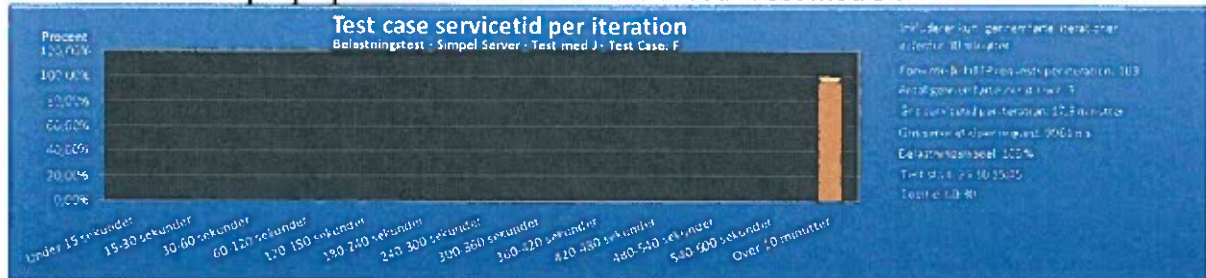


Analyse

Det er tydeligt ved denne test case at serveren ganske enkelt ikke kan håndtere requests på en fornuftig vis. Det er kun et absolut fåtal af iterationerne, der kommer igennem, og der opleves yderst lange servicetider for disse.

Belastningstest mod 1 simpel server – Test case F Test case F for 'Køretøj'.

Nedenfor et eksempel på performance for test case F ved "Test med J".



Nedenfor et eksempel på performance for test case F ved "Test uden J".



Analyse

Det er tydeligt ved denne test case at serveren ganske enkelt ikke kan håndtere requests på en fornuftig vis. Det er kun et absolut fåtal af iterationerne, der kommer igennem, og der opleves yderst lange servicetider for disse.

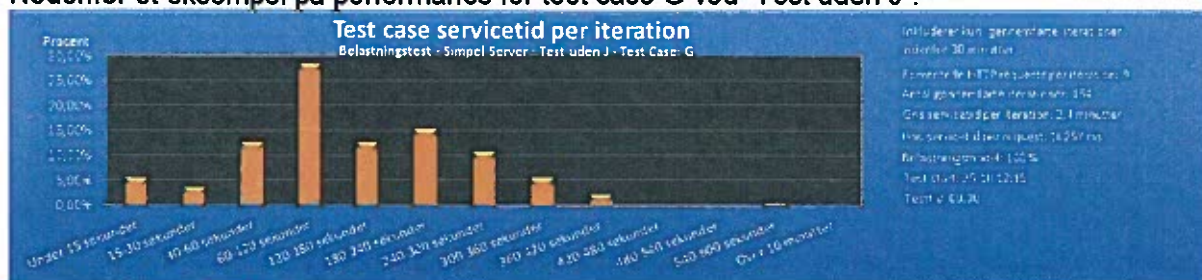
Belastningstest mod 1 simpel server – Test case G

Test case G for 'Søg'.

Nedenfor et eksempel på performance for test case G ved "Test med J".



Nedenfor et eksempel på performance for test case G ved "Test uden J".



Analyse

Ved denne "lette" test case lykkes det POLSAG-miljøet at færdiggøre relativt mange test cases. Den er dog stadig ikke i nærheden af at levere, hvad belastningstesten forudsætter skal gennemføres indenfor testperioden.

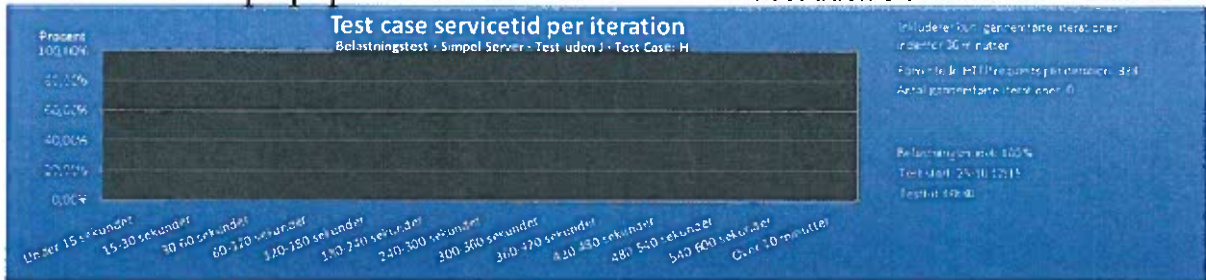
Belastningstest mod 1 simpel server – Test case H

Test case H for 'Dan sagsstyring'.

Nedenfor et eksempel på performance for test case H ved "Test med J".



Nedenfor et eksempel på performance for test case H ved "Test uden J".



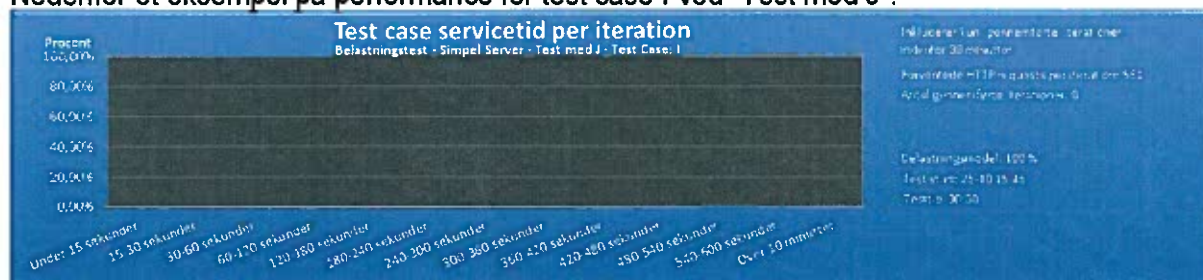
Analyse

Her færdiggøres ingen test cases, hvilket viser at denne hardwarekonfiguration ganske enkelt ikke er i stand til at håndtere belastningen.

Belastningstest mod 1 simpel server – Test case I

Test case I for 'Afslut sag'.

Nedenfor et eksempel på performance for test case I ved "Test med J".



Nedenfor et eksempel på performance for test case I ved "Test uden J".



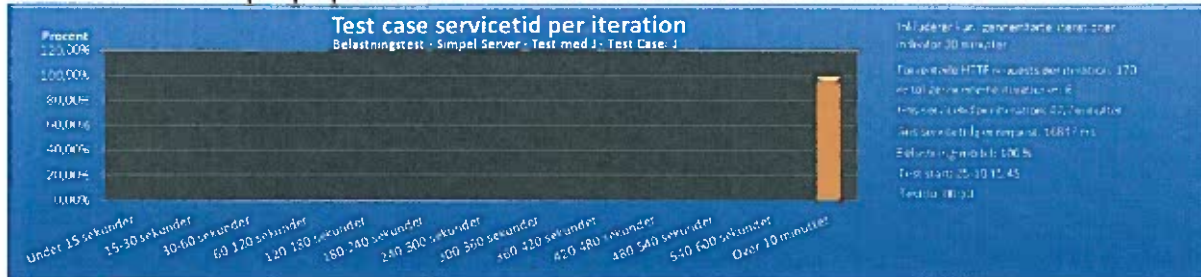
Analyse

Her færdiggøres ingen test cases, hvilket viser at denne hardwarekonfiguration ganske enkelt ikke er i stand til at håndtere belastningen.

Belastningstest mod 1 simpel server – Test case J

Test case J for 'Træk døgnrapport'.

Nedenfor et eksempel på performance for test case J ved "Test med J".



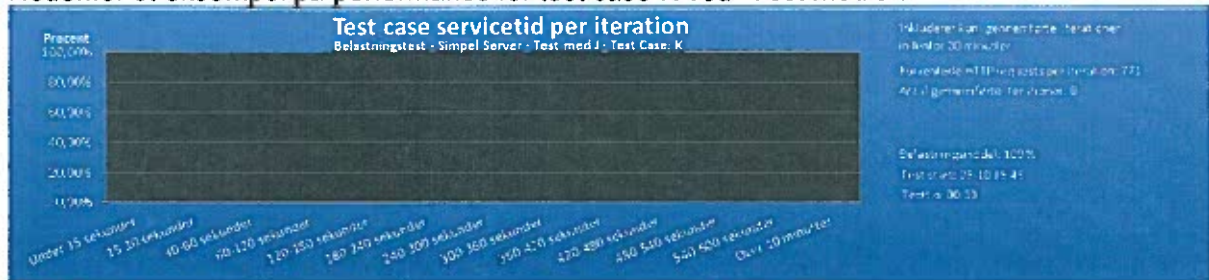
Analyse

Det er tydeligt ved denne test case at serveren ganske enkelt ikke kan håndtere requests på en fornuftig vis. Det er kun et absolut fåtal af iterationerne, der kommer igennem, og der opleves yderst lange servicetider for disse.

Belastningstest mod 1 simpel server – Test case K

Test case K for 'K_USO_ Fremsøg pakke dokumenter'.

Nedenfor et eksempel på performance for test case K ved "Test med J".



Nedenfor et eksempel på performance for test case K ved "Test uden J".



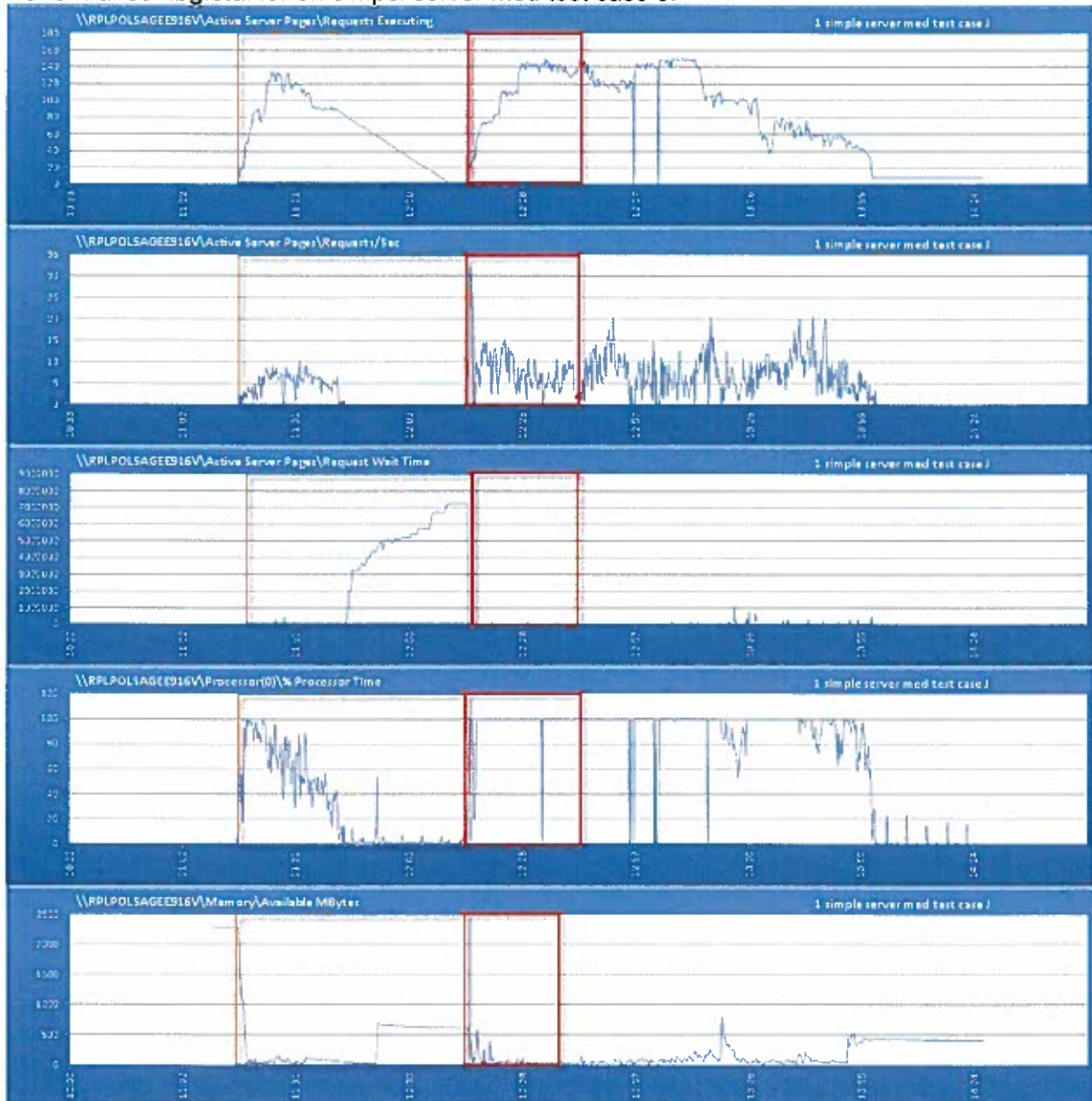
Analyse

Her færdiggøres ingen test cases, hvilket viser at denne hardwarekonfiguration ganske enkelt ikke er i stand til at håndtere belastningen.

Belastningstest mod 1 simpel server – Server performance

Som det ses forinden er systemet presset til det yderste i forbindelse med både afviklingen af belastningstest med og uden J på en simpel server.

Performance nøgletal for en simpel server med test case J.



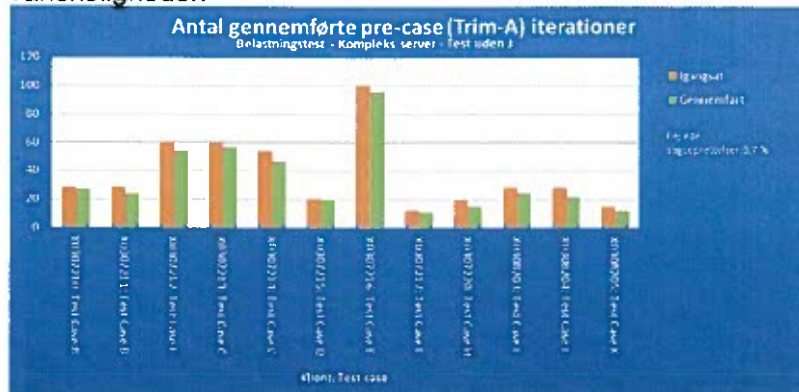
Performance nøgletal for 1 simpel server uden test case J.



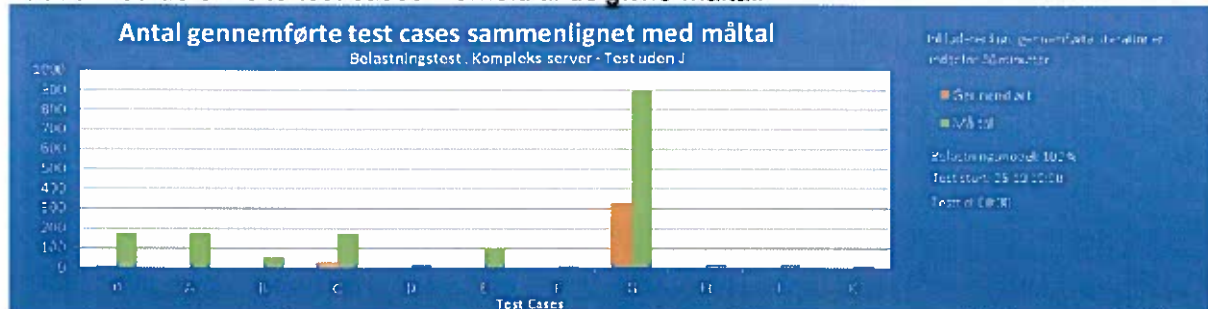
Belastningstest mod 1 kompleks server

I et forsøg på at afdække, hvordan POLSAGs ydelse varierer med hardware har vi endvidere forsøgt at afvikle den fulde belastningsmodel mod en kompleks server. Der er i dette tilfælde benyttet "belastningsmodel uden test case J".

Forud for testene er der som vanligt blevet kørt Trim-A cases, som er forløbet uden for store vanskeligheder.



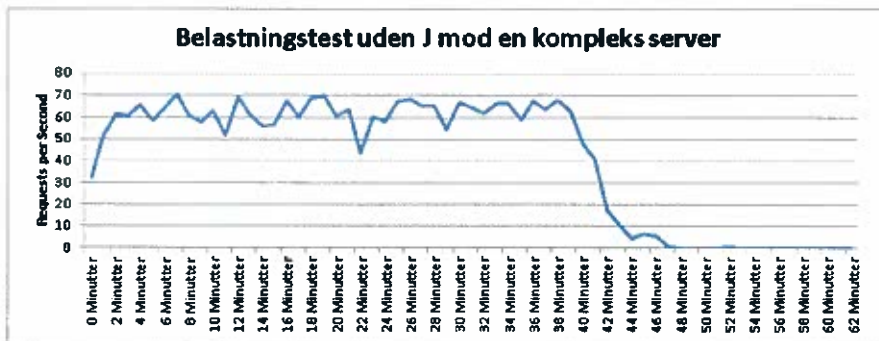
I nedenstående oversigt er den øverste graf derfor en oversigt der viser, hvor langt systemet nåede med de enkelte test cases i forhold til de givne måltal.



Testen har kørt i en halv time og er ikke nærheden af at nå sine måltal. På baggrund af de tilsvarende resultater for NLB-miljøet er dette dog forventeligt.

Til gengæld kan man glimtvis opleve forbedringer i forhold til den simple server. Dee ligger dog på den beskedne side.

Kigger vi på de requests fra testen som den komplekse server har håndteret går billedet også igen i relation til at den komplekse server har skulle arbejde endog lang tid efter testens ophør for at færdiggøre håndteringen af requests (altså requests, der var sendt afsted før testen stoppede).



Belastningstest mod 1 kompleks server – Test case 0 Test case 0 for 'Åbn forsiden'.

Nedenfor et eksempel på performance for test case 0 ved "Test uden J".

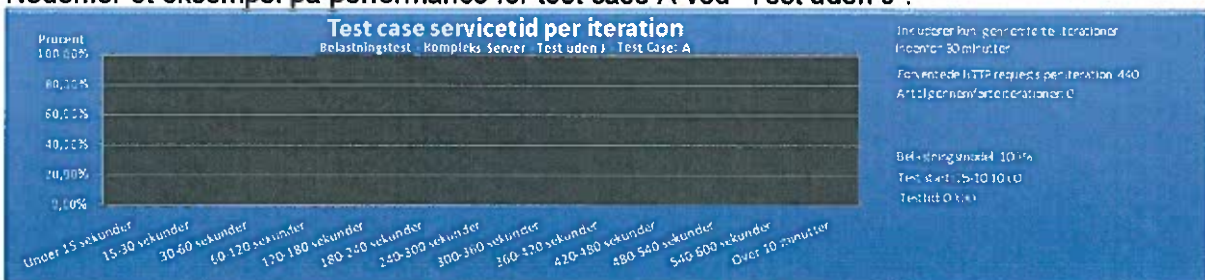


Analyse

Det er tydeligt ved denne test case at serveren ganske enkelt ikke kan håndtere requests på en fornuftig vis. Det er kun et absolut fåtal af iterationerne, der kommer igennem, og der opleves yderst lange servicetider for disse.

Belastningstest mod 1 kompleks server – Test case A Test case A for 'Grundregistrering'.

Nedenfor et eksempel på performance for test case A ved "Test uden J".



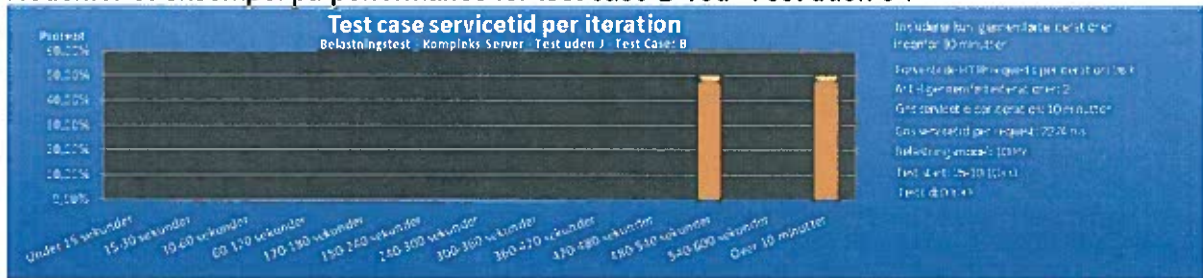
Analyse

Her færdiggøres ingen test cases, hvilket viser at denne hardwarekonfiguration ganske enkelt ikke er i stand til at håndtere belastningen.

Belastningstest mod 1 kompleks server – Test case B

Test case B for 'Opret dokument'.

Nedenfor et eksempel på performance for test case B ved "Test uden J".



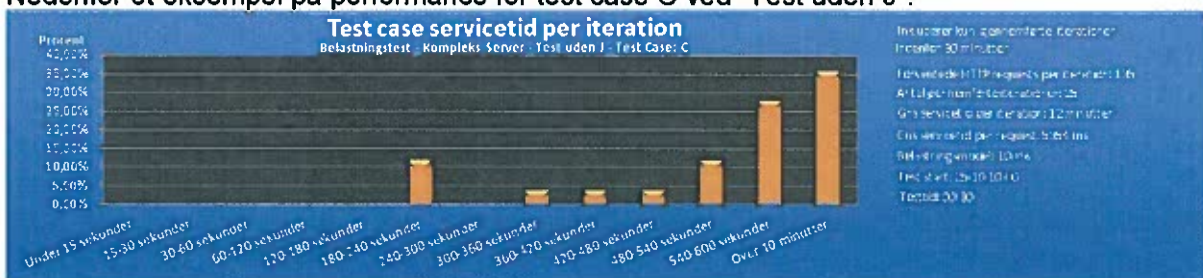
Analyse

Det er tydeligt ved denne test case at serveren ganske enkelt ikke kan håndtere requests på en fornuftig vis. Det er kun et absolut fåtal af iterationerne, der kommer igennem, og der opleves yderst lange servicetider for disse.

Belastningstest mod 1 kompleks server – Test case C

Test case C for 'Sagsplacering'.

Nedenfor et eksempel på performance for test case C ved "Test uden J".



Analyse

Det er tydeligt ved denne test case at serveren ganske enkelt ikke kan håndtere requests på en fornuftig vis. Det er kun et absolut fåtal af iterationerne, der kommer igennem, og der opleves yderst lange servicetider for disse.

Belastningstest mod 1 kompleks server – Test case D Test case D for 'Sigtelse'.

Nedenfor et eksempel på performance for test case D ved "Test uden J".

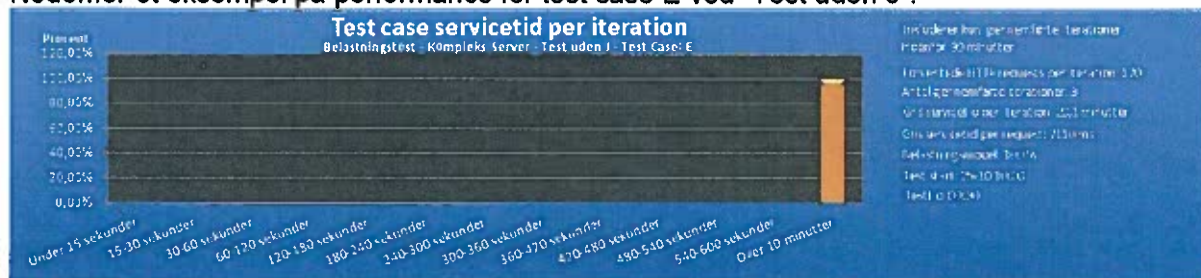


Analyse

Her færdiggøres ingen test cases, hvilket viser at denne hardwarekonfiguration ganske enkelt ikke er i stand til at håndtere belastningen.

Belastningstest mod 1 kompleks server – Test case E Test case E for 'Sikret genstand'.

Nedenfor et eksempel på performance for test case E ved "Test uden J".

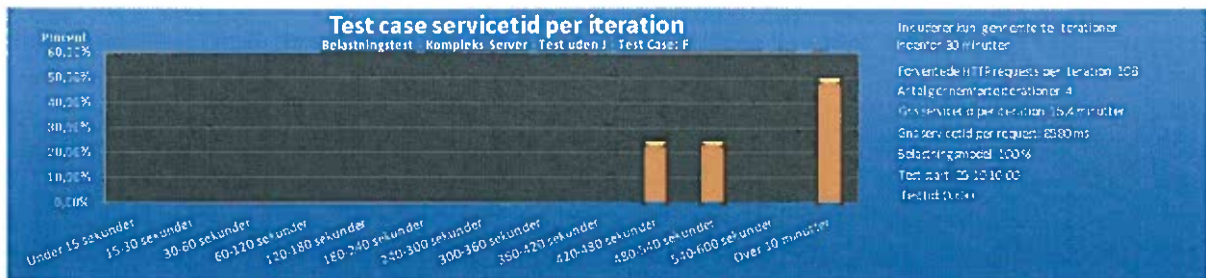


Analyse

Det er tydeligt ved denne test case at serveren ganske enkelt ikke kan håndtere requests på en fornuftig vis. Det er kun et absolut fåtal af iterationerne, der kommer igennem, og der opleves yderst lange servicetider for disse.

Belastningstest mod 1 kompleks server – Test case F Test case F for 'Køretøj'.

Nedenfor et eksempel på performance for test case F ved "Test uden J".



Analyse

Det er tydeligt ved denne test case at serveren ganske enkelt ikke kan håndtere requests på en fornuftig vis. Det er kun et absolut fåtal af iterationerne, der kommer igennem, og der opleves yderst lange servicetider for disse.

Belastningstest mod 1 kompleks server – Test case G

Test case G for 'Søg'.

Nedenfor et eksempel på performance for test case G ved "Test uden J".



Analyse

Ved denne "lette" test case lykkes det POLSAG-miljøet at færdiggøre relativt mange test cases. Den er dog stadig ikke i nærheden af at levere, hvad belastningstesten forudsætter skal gennemføres indenfor testperioden.

Belastningstest mod 1 kompleks server – Test case H

Test case H for 'Dan sagsstyring'.

Nedenfor et eksempel på performance for test case H ved "Test uden J".



Analyse

Her færdiggøres ingen test cases, hvilket viser at denne hardwarekonfiguration ganske enkelt ikke er i stand til at håndtere belastningen.

Belastningstest mod 1 kompleks server – Test case I

Test case I for 'Afslut sag'.

Nedenfor et eksempel på performance for test case I ved "Test uden J".



Analyse

Her færdiggøres ingen test cases, hvilket viser at denne hardwarekonfiguration ganske enkelt ikke er i stand til at håndtere belastningen.

Belastningstest mod 1 kompleks server – Test case K

Test case K for 'K_USO_ Fremsøg pakke dokumenter'.

Nedenfor et eksempel på performance for test case K ved "Test uden J".

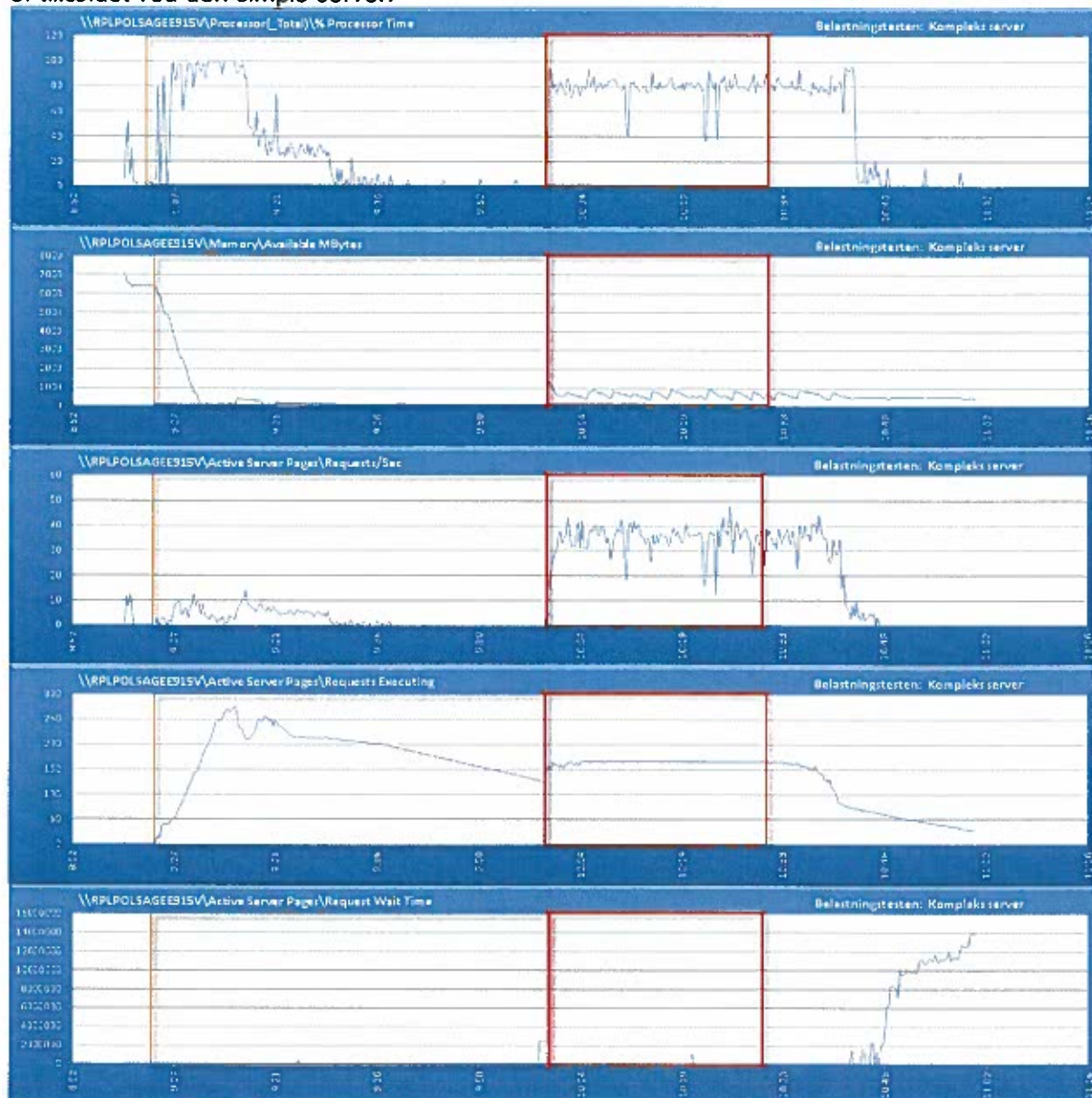


Analyse

Her færdiggøres ingen test cases, hvilket viser at denne hardwarekonfiguration ganske enkelt ikke er i stand til at håndtere belastningen.

Belastningstest mod 1 kompleks server – Server performance

Som det ses forneden er systemet presset helt i bund i forbindelse med afviklingen af "belastningstest uden J" på den komplekse server. Belastningen er dog ikke helt så hård som det er tilfældet ved den simple server.

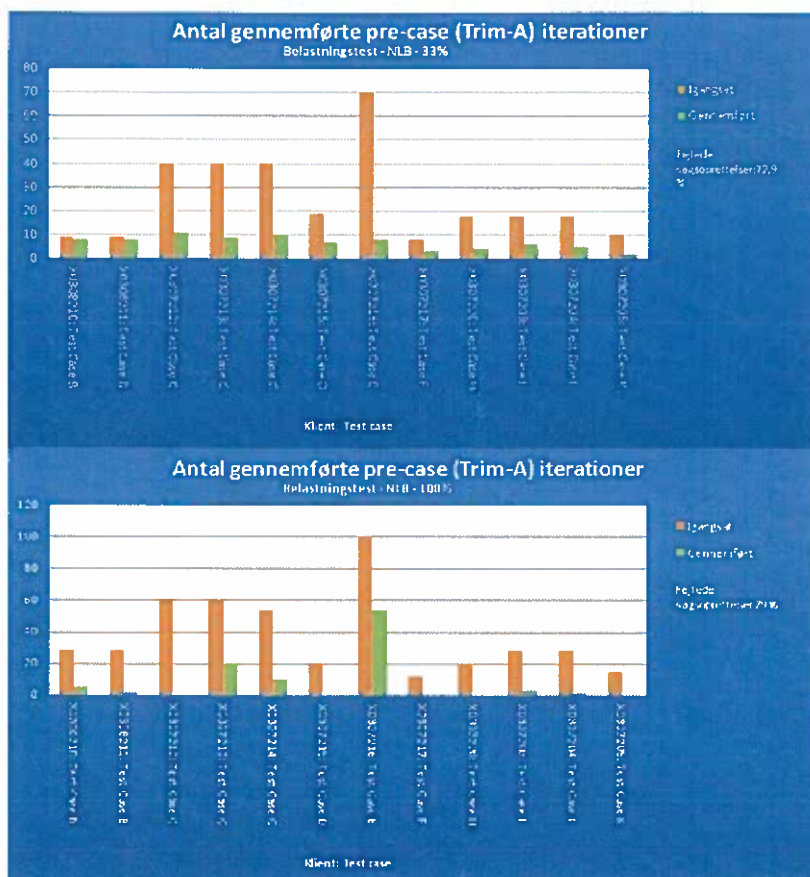


Belastningstests - Problemløst

Det har været utroligt svært, at køre belastningstests mod systemet, da systemet igen og igen er crashet med ubrugelige testresultater til følge. Dette er selvfølgelig i sig selv er med til at underbygge, at POLSAG-miljøet ganske enkelt ikke er i stand til at håndtere den forventede belastning.

POLSAG-miljøet bliver endda ustabil, når det udsættes for to tredjedele af den belastning, det er meningen det bør kunne håndtere på en enkelt server.

Eksempelvis køres der forud for belastningstestene en præfase, hvor sager oprettes. Vi kalder disse typer sager for Trim-A, da de er en "trimmet" version af den normale sagsoprettelsestest. Selv ved disse tests er der problemer med at få færdige iterationer. Nedenfor vises hvor få test cases der i nogle situationer reelt bliver kørt til ende (her vist ved henholdsvis 33% og 100% belastningstest på NLB).



Når Trim-A ikke kan nå at køre, så er det noget der tegner rigtig skidt for systemet, da det jo reelt udgør en lille del af de requests, som er påkrævet i forbindelse med den "komplette" sagsoprettelsesproces i POLSAG.

Selvom Trim-A ikke lykkes fuldt ud er testene i øvrigt stadig valide, idet vi kører det samme antal http requests. Det er dog naturligvis ikke helt lige så godt som hvis Trim-A er lykkes. Men forskellen er dog negligerbar henset til de meget klare performancemæssige problemer, der påvises.

Under de fleste belastningstests er serverne ligefrem stået af og begyndt at sende status 503 tilbage på http requests. 503 status indikerer at webserveren (som afvikles POLSAG-systemet) ikke længere kan håndtere http requests grundet temporær overloading.

Nedenfor er vist nogle eksempler fra vores testkørsler på, hvor mange sådanne 503 statussvar vi har observeret:

Dato	Testtype	Antal 503 status svar
12-10-2011	Belast NLB 100	78803
19-okt	Belastning 4 servere - 100%	50604
28-okt	Belastning NLB - 66%	63478
28-okt	Belastning NLB - 100%	204463
28-okt	Belastning NLB - 33%	50130
01-nov	Belastning 2 servere - NLB	153998

Belastningstestene, der er dokumenteret i rapporten, repræsenterer i øvrigt de bedste udfald, der har været ud af de mange belastningstests, der er blevet afviklet i uddannelsesmiljøet.

Døgnrapporter mod NLB

Der er ialt blevet kørt 11 tests med døgnrapporter med forskellige sagsantal.

Nedenstående tabel angiver, hvor mange requests der blev afviklet i den pågældende døgnrapport-kørsel og tidsforbruget til den pågældende test. Som det kan ses har mange af testene kørt i endog meget lang tid og der er noteret store forskelle i antallet af requests. Årsagen til dette forhold vil blive afsløret lidt længere nede i afsnittet.

Test	Test start	Test slut	Antal requests
X0307202 træk døgnrapport 10sager i periode 18-10-2011 00-23	21-10-11 11:21	21-10-11 12:31	52
X0307202 træk døgnrapport 20sager i periode 20-10-2011 00-23	21-10-11 12:41	21-10-11 12:42	28
X0307204 træk døgnrapport 10sager i periode 05-05-2011 00-23	21-10-11 9:06	21-10-11 9:09	81
X0307204 træk døgnrapport 10sager i periode 19-10-2011 00-23	21-10-11 10:53	21-10-11 10:54	27
X0307204 træk døgnrapport 10sager i periode 20-10-2011 00-23	21-10-11 9:34	21-10-11 10:35	30
X0307204 træk døgnrapport 20sager i periode 05-05-2011 00-23	21-10-11 9:14	21-10-11 9:17	93
X0307204 træk døgnrapport 20sager i periode 20-10-2011 00-23	21-10-11 12:20	21-10-11 12:20	27
X0307204 træk døgnrapport 30sager i periode 05-05-2011 00-23	21-10-11 9:22	21-10-11 9:25	96
X0307204 træk døgnrapport 30sager i periode 20-10-2011 00-23	21-10-11 9:34	21-10-11 10:35	30
X0307204 træk døgnrapport 50sager i periode 05-05-2011 00-23	21-10-11 9:27	21-10-11 9:30	99
X0307204 træk døgnrapport 50sager i periode 20-10-2011 00-23	21-10-11 14:45	21-10-11 14:45	27

Ved første overfladiske kig er der ikke så meget at sige om denne test.

Tingene bliver dog langt mere interessante, når man også begynder at dykke dybere i det forhold, at hovedparten af rapporttrækkene aldrig kommer retur fra serveren.

Webserveren svarede i hovedparten af tilfældene ikke tilbage med listen af døgnrapporter, uanset om det drejede sig om 10, 20, 30 og 50 sager. Testen blev herefter afbrudt. I efterfølgende analyse af Fiddler logs fra de pågældende kørsler er det dokumenteret, at webserveren modtager klientens http request der indeholder søgeparametre, men at serveren ikke kommer tilbage med et response indenfor testens løbetid, hvilket er årsagen til de mange relativt lave request-antal.

Set fra brugerens synspunkt ville disse situationer opleves som at browseren fryser og ikke reagerer på brugerens input. I praksis betyder det at systemet ikke kan anvendes af af den enkelte sagsbehandler i en periode.

Et detaljeret studie af de afviklede tests af døgnrapporten viser at webserveren konsekvent ikke vender tilbage med et svar, når der trækkes rapporter i oktober måned. Til gengæld fungerer rapporttrækket fint for maj måned. I oktober måned er der typisk blevet oprettet mellem 500 og 1.000 sager om dagen på grund af testkørslerne, hvorimod der kun formodes at være oprettet ganske få sager i maj måned, hvor miljøet stadig udelukkende blev benyttet til uddannelse.

Det synes således nærliggende at konkludere at svartiden på døgnrapporterne er meget følsom for antallet af sager, der findes i den valgte tidsperiode¹⁴. Til gengæld lader antallet af sager, der medtages i døgnrapporten, ikke at have nogen væsentlig betydning for svartiderne.

I det følgende vises traces fra hver enkelt træk af døgnrapport.

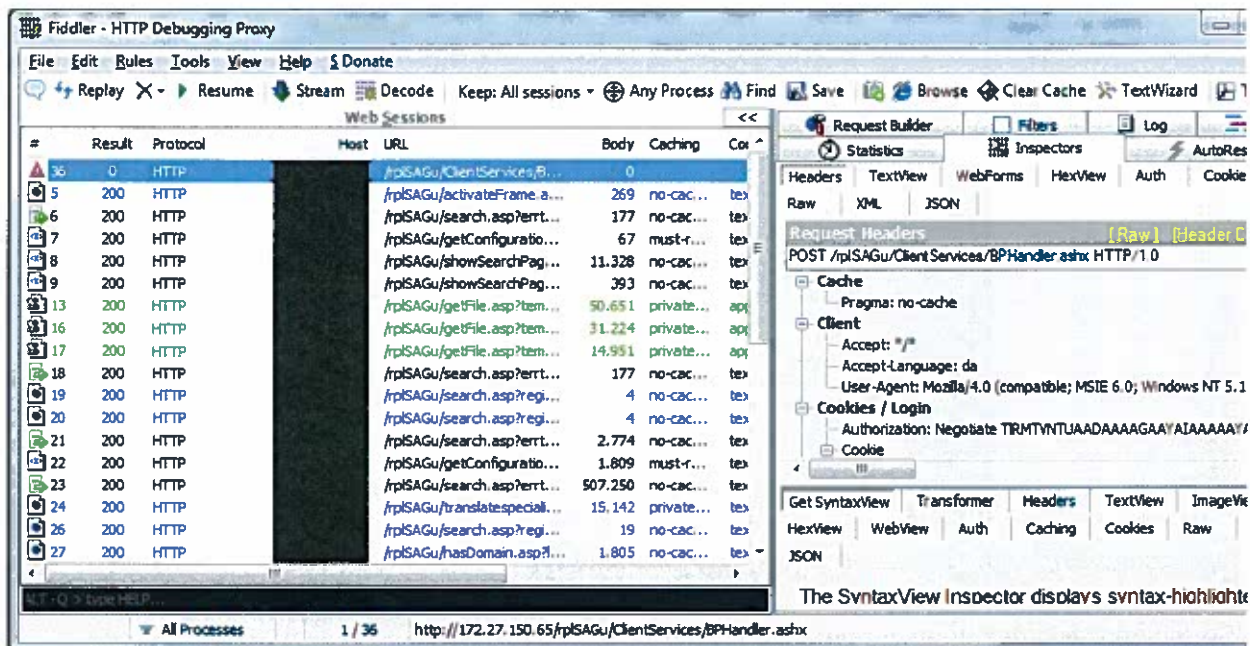
X0307202 træk døgnrapport 10sager i periode 18-10-2011 00-23

#	Result	Protocol	Host	URL	Body	Caching	Content Type
38	0	HTTP		/fpISAGU/ClientServices/...	0		
5	200	HTTP		/fpISAGU/activateFrame.a...	269	no-cac...	text
6	200	HTTP		/fpISAGU/search.asp?errt...	177	no-cac...	text
7	200	HTTP		/fpISAGU/getConfiguratio...	67	must-r...	text
8	200	HTTP		/fpISAGU/showSearchPag...	11.328	no-cac...	text
9	200	HTTP		/fpISAGU/showSearchPag...	393	no-cac...	text
13	200	HTTP		/fpISAGU/getFile.asp?tem...	50.651	private...	application
16	200	HTTP		/fpISAGU/getFile.asp?tem...	31.224	private...	application
17	200	HTTP		/fpISAGU/getFile.asp?tem...	14.951	private...	application
18	200	HTTP		/fpISAGU/search.asp?errt...	177	no-cac...	text
19	200	HTTP		/fpISAGU/search.asp?regi...	4	no-cac...	text
20	200	HTTP		/fpISAGU/search.asp?regi...	4	no-cac...	text
21	200	HTTP		/fpISAGU/search.asp?errt...	2.774	no-cac...	text
22	200	HTTP		/fpISAGU/getConfiguratio...	1.809	must-r...	text
23	200	HTTP		/fpISAGU/search.asp?errt...	507.250	no-cac...	text
24	200	HTTP		/fpISAGU/translateSpecia...	15.142	private...	text
26	200	HTTP		/fpISAGU/search.asp?regi...	19	no-cac...	text
29	200	HTTP		/fpISAGU/search.asp?regi...	19	no-cac...	text

Søgningen timede ud.

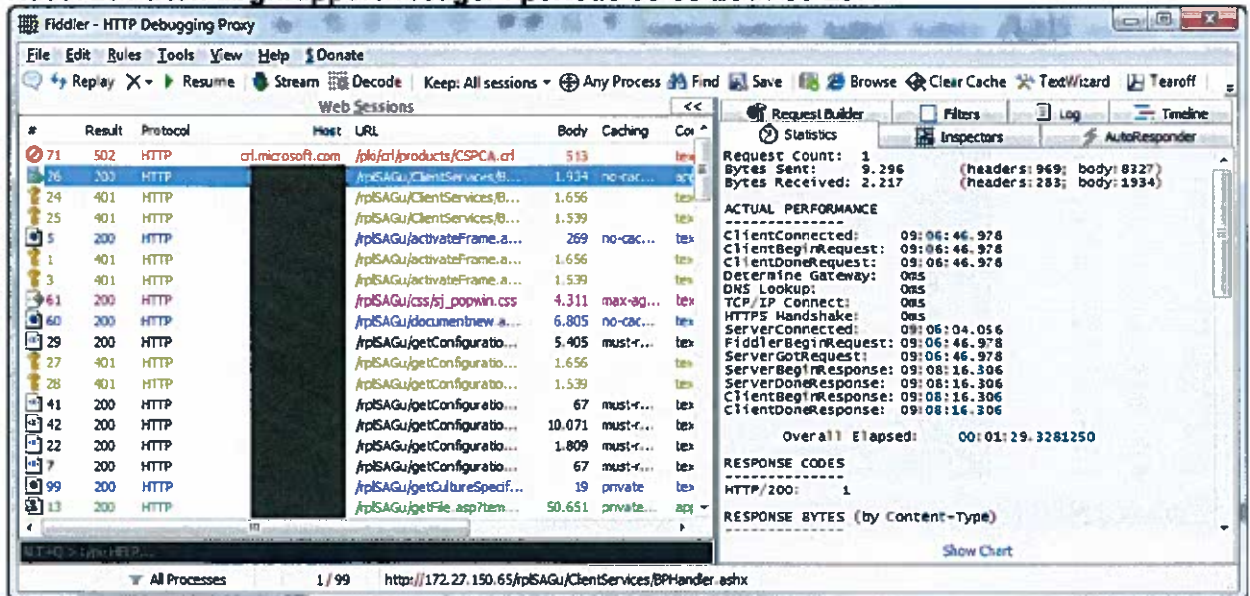
X0307202 træk døgnrapport 20sager i periode 20-10-2011 00-23

¹⁴ I alle andre kørsler af test case J (dvs. i alle de tests, der ellers er blevet afviklet i uddannelsesmiljøet) sker døgnrapporttrækket for perioden 13/6 kl. 12 til 14/6 kl. 12. Der formodes således ikke at være ret mange sager involveret i disse tests med døgnrapporten.



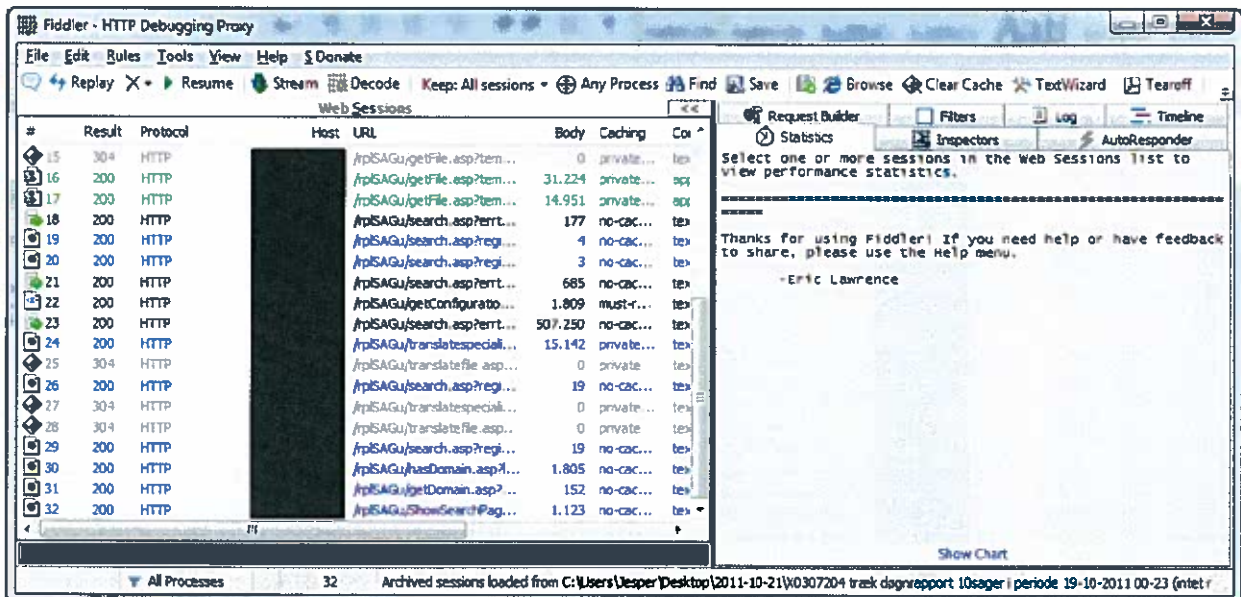
Søgningen timede ud.

X0307204 træk døgnrapport 10sager i periode 05-05-2011 00-23



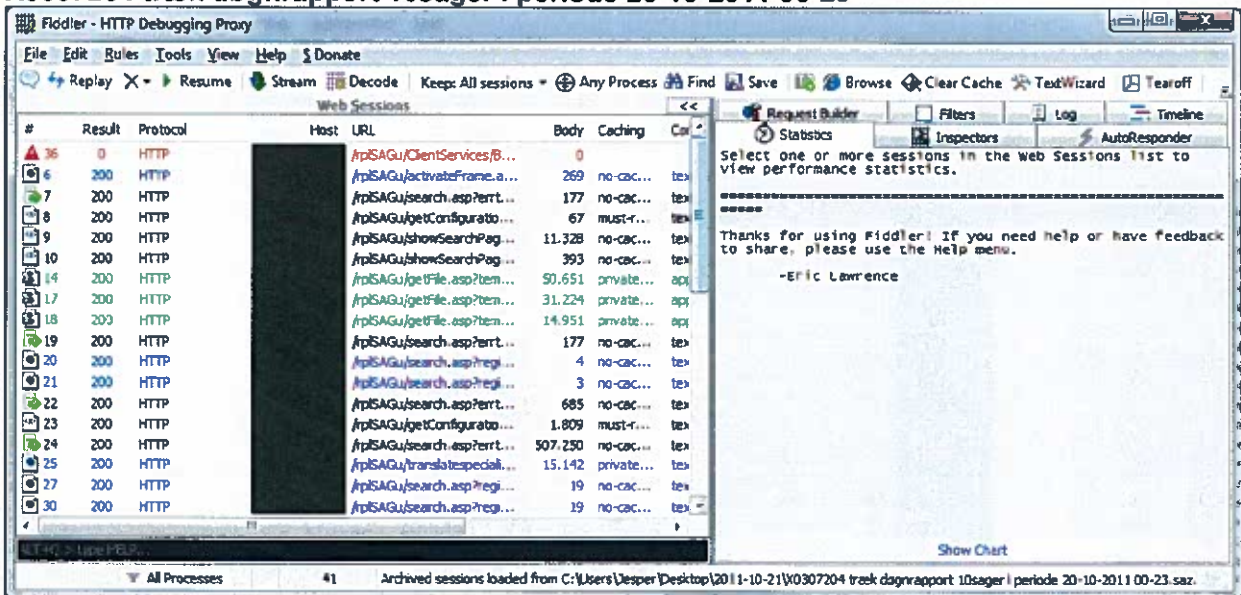
Denne kørte igennem. Det tog 2 minutter at udføre søgningen.

X0307204 træk døgnrapport 10sager i periode 19-10-2011 00-23



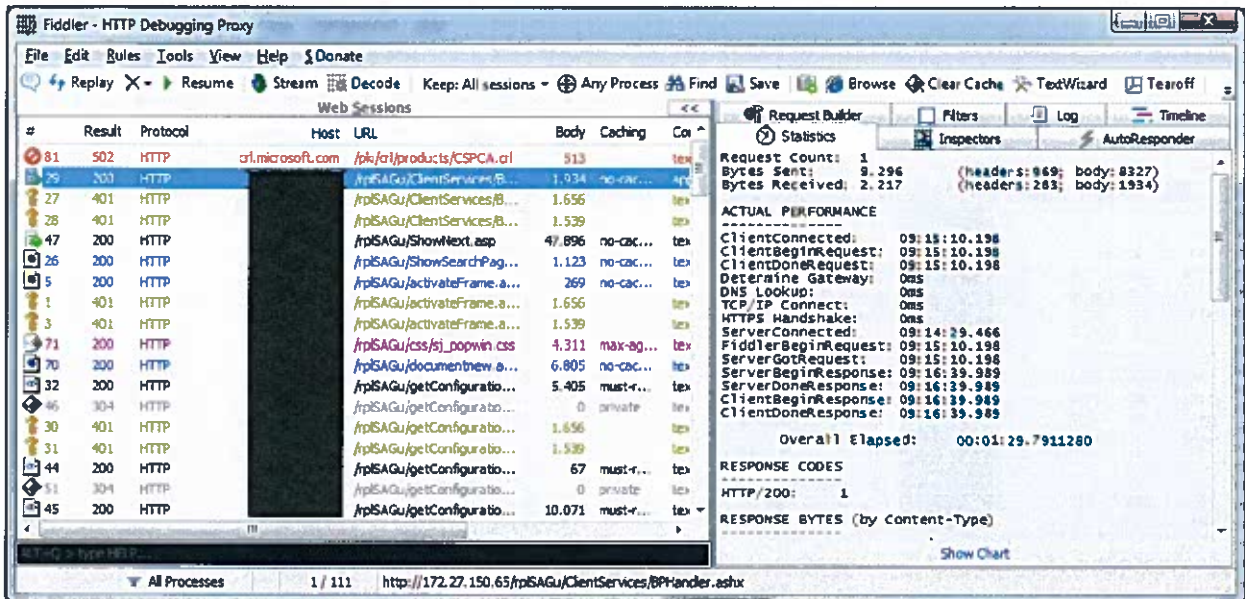
Søgningen lykkedes, men der var ingen sager i søgningsresultatet.

X0307204 træk døgnrapport 10sager i periode 20-10-2011 00-23



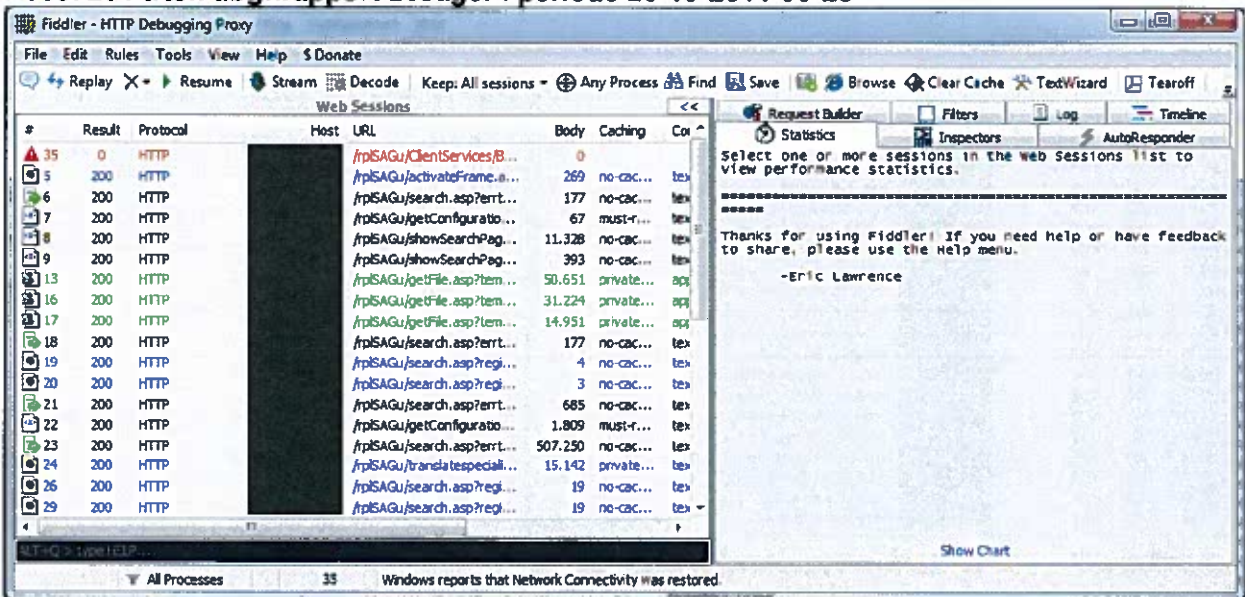
Søgningen timede ud.

X0307204 træk døgnrapport 20sager i periode 05-05-2011 00-23.saz



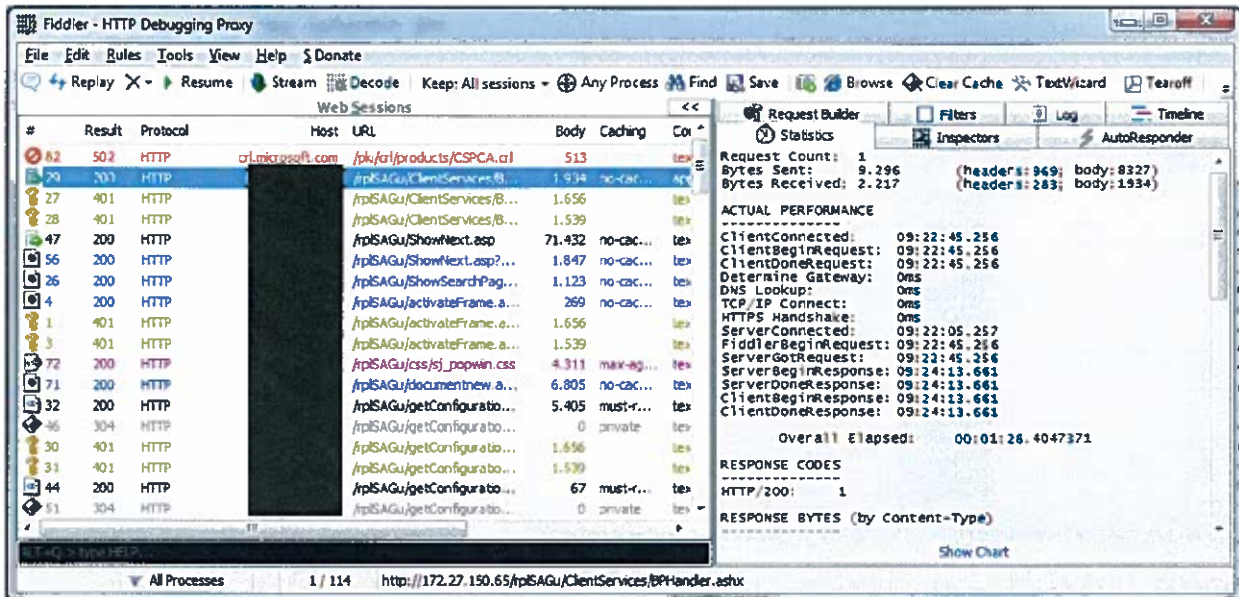
Denne kørte igennem. Det tog 1½ minut at udføre søgningen.

X0307204 træk døgnrapport 20sager i periode 20-10-2011 00-23



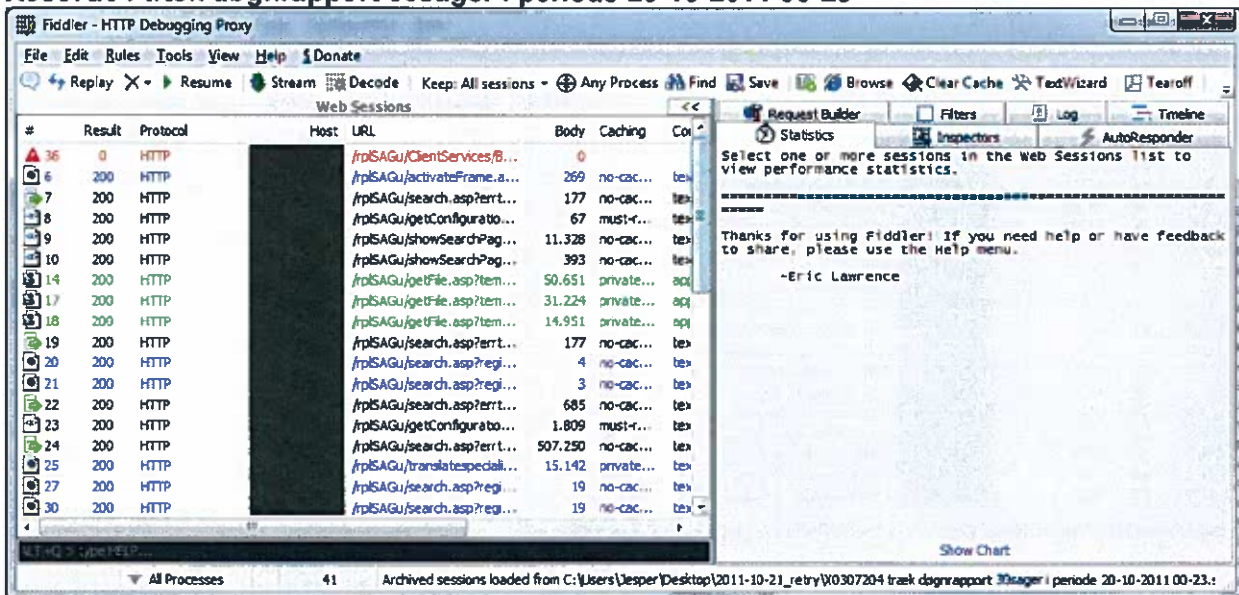
Søgningen timede ud.

X0307204 træk døgnrapport 30sager i periode 05-05-2011 00-23



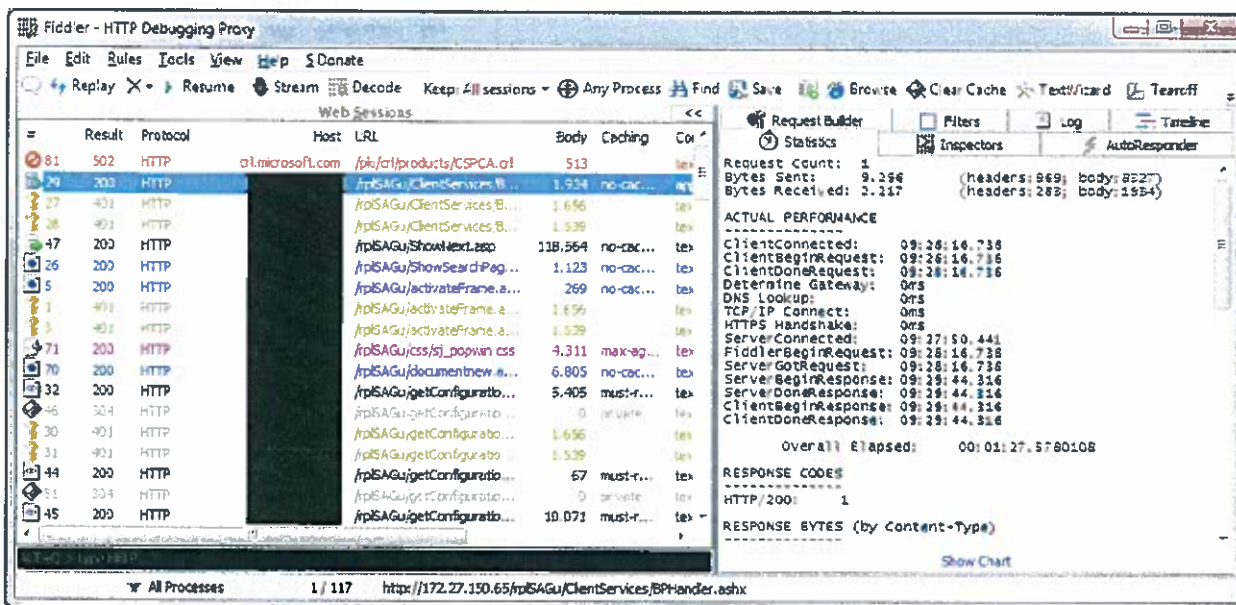
Denne kørte igennem. Det tog 1½ minut at udføre søgningen.

X0307204 træk døgnrapport 30sager i periode 20-10-2011 00-23



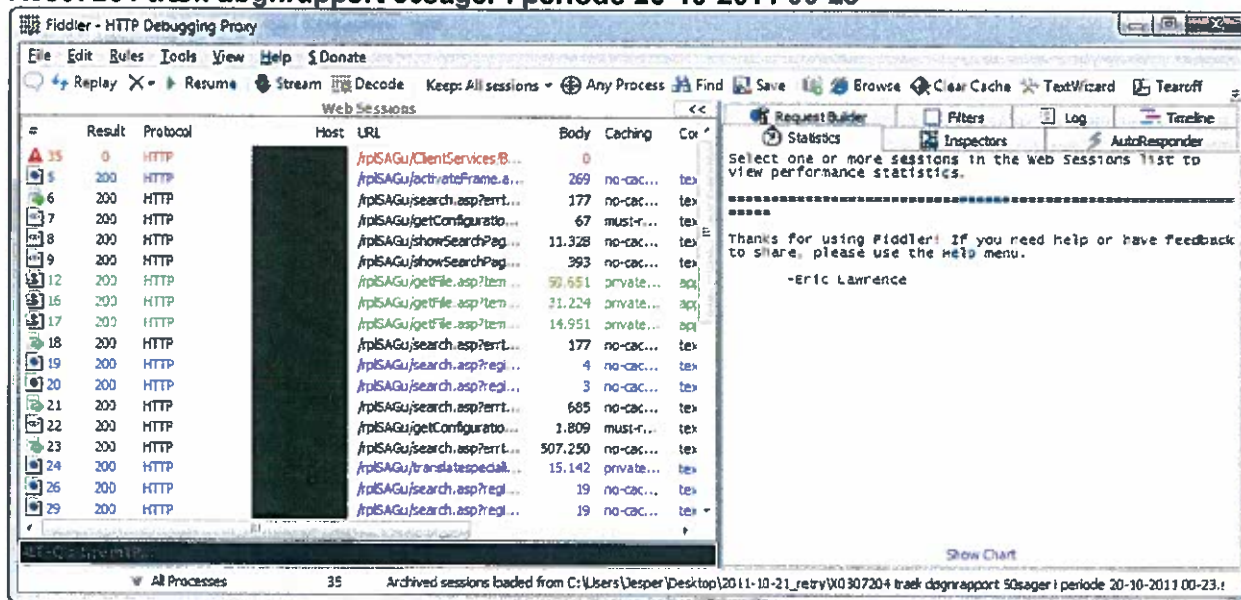
Søgningen timede ud.

X0307204 træk døgnrapport 50sager i periode 05-05-2011 00-23



Denne kørte igennem. Det tog 1½ minut at udføre søgningen.

X0307204 træk døgnrapport 50sager i periode 20-10-2011 00-23



Søgningen timede ud.

8.00-8.15 tests

8.00-8.15 opsætter kravene til det antal iterationer af to test cases, som Rigspolitiet forventer at systemet vil skulle understøtte i den travle periode om morgenen mellem 8:00 og 8:15, hvor et stort antal medarbejdere erfaringsmæssigt vil logge ind i systemet og trække døgnrapporter.

8.00-8.15 testen vurderes at repræsentere et fornuftigt – og formentlig lidt konservativt – skøn over den normale belastning på POLSAG mandag morgen, idet POLSAG formentlig vil blive brugt til at udføre andre arbejds gange end dem, der er repræsenteret i de 12 test cases.

Den opstillede model yder ikke nogen garanti for at POLSAG-miljøet vil opføre sig så godt som aftestningen angiver. Omvendt må det formodes at POLSAG-miljøet i hvert fald ikke vil opføre sig bedre end hvad aftestningen indikerer.

Det primære formål med 8.00-8.15 testen er at afdække POLSAGs opførelse under normal belastning mandag morgen:

- Hvordan systemet opfører sig under den forventede belastning.
- At afdække hvilke kritiske aktiviteter, der belaster systemet mest, herunder afdække om der er ikke-kritiske aktiviteter, der belaster systemet unødigt meget.

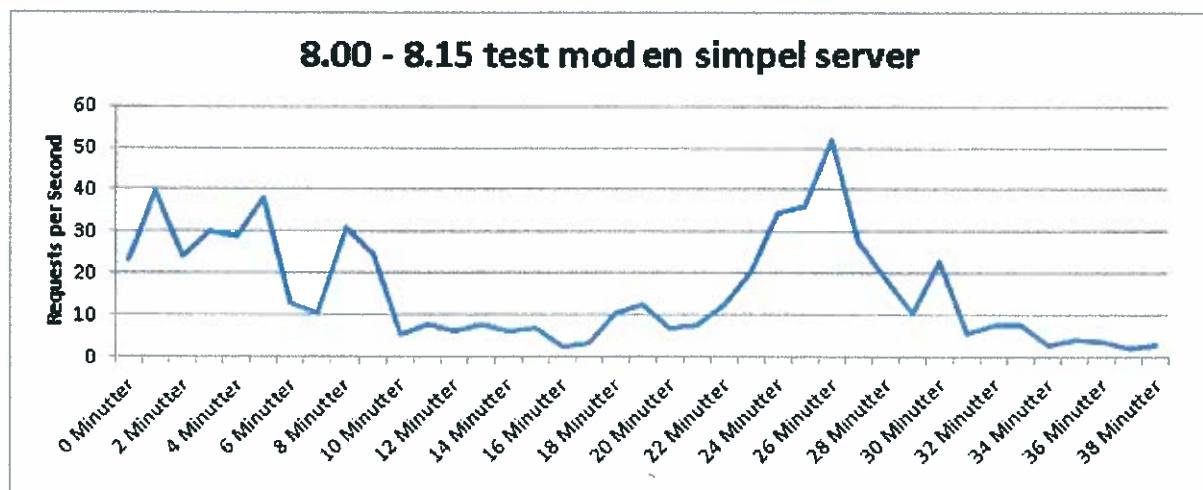
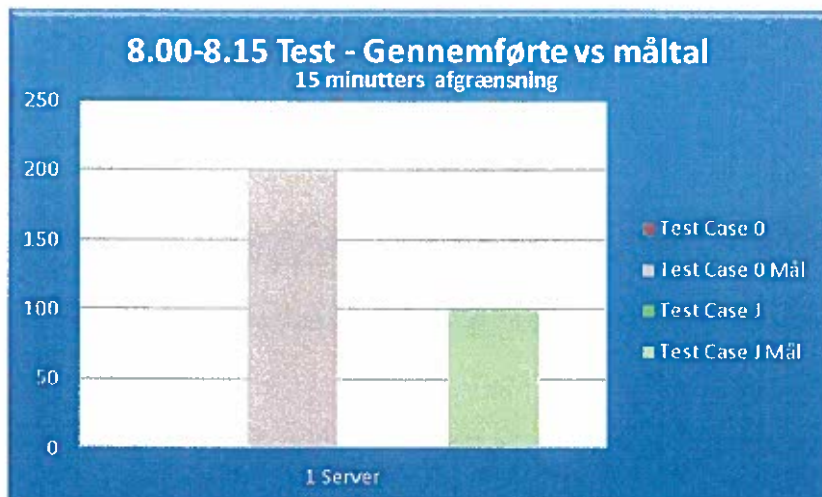
Der er konstateret en stor ustabilitet under testafviklingen af testcase J, som ikke kun er forbundet til belastning. Denne ustabilitet vurderes generelt at have positiv indvirkning på testresultaterne sådan at forstå at de afvikles hurtigere end ellers.

Der er under aftestningen observeret forskellige niveauer af følgende problemer ved kaldet til BPHandler (som er det tunge kald i døgnrapporten):

1. *Kaldet kommer aldrig igennem Authentication, der kommer 3 gange http 401*
2. *POLSAG returnerer en underlig fejlmeddelelse omkring rettighedsfejl (efter lang tid)¹⁵. Forespørgslen har i alle tilfælde tidligere eller senere været afviklet med succes for samme bruger og parametre.*
3. *POLSAG-koden fejler med en C#-fejl¹⁶. Forespørgslen har i alle tilfælde tidligere eller senere været afviklet med succes for samme bruger og parametre.*

8.00-8.15 test mod 1 simpel server

Vi lægger ud med at køre 8.00-8.15 testen mod en simpel server for at se, hvordan testen klarer sig på den hardware, som kontrakten forudsætter kan håndtere belastningen i en kreds. Som det kan ses, er dette for stor en mundfuld for systemet, da ikke en eneste iteration bliver gennemført indenfor de første 15 minutter.

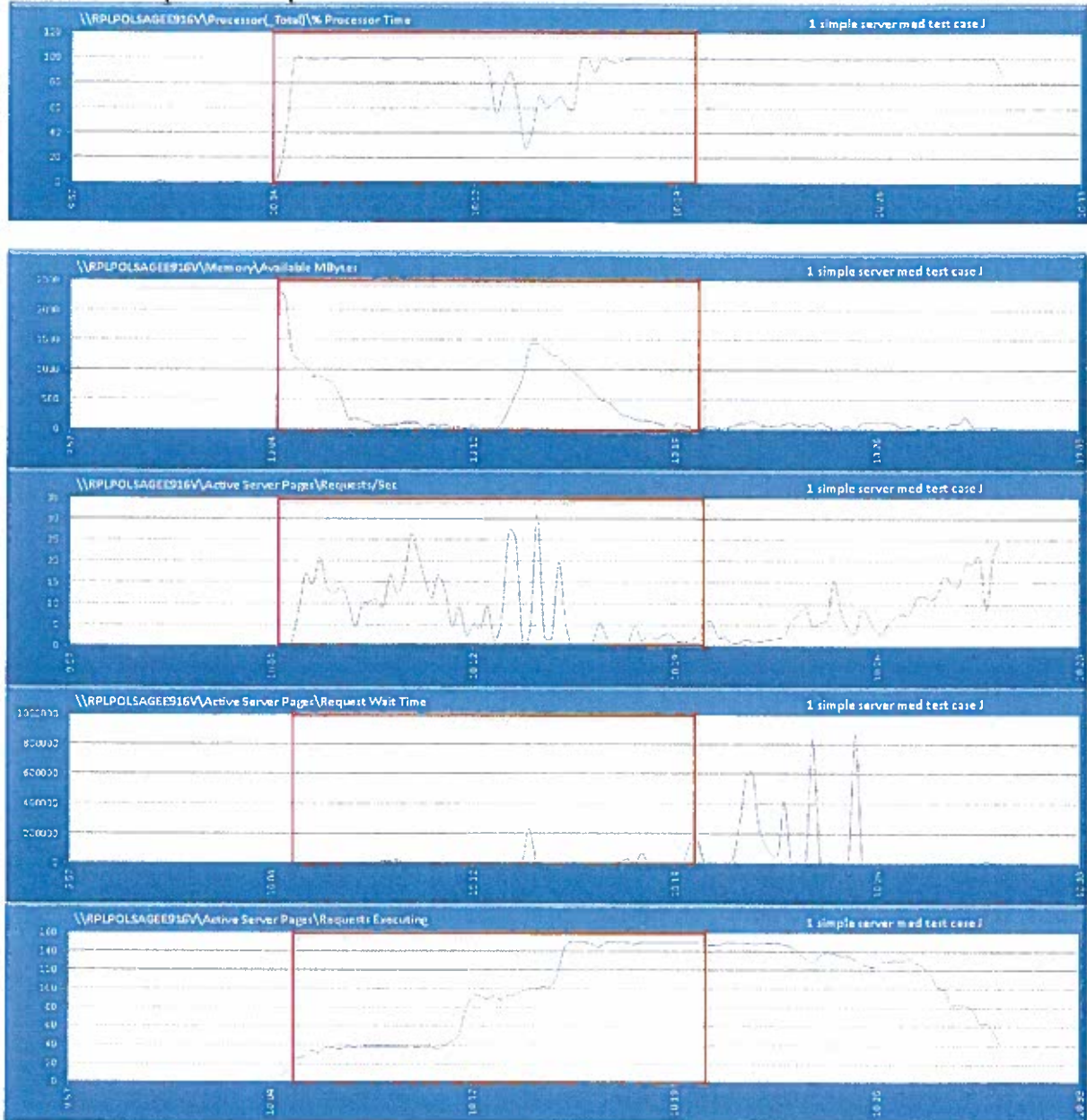


Grunden til at testen fortsætter udover de 15 minutters løbetid er, at der for 8.00-8.15 testen er at der blev afviklet et ekstra sæt iterationer for at simulere hvordan arbejdsbelastningen vil være ved afslutningen af 8.00-8.15 testen (brugermængden vil jo selvfølgelig ikke aftage i den travle morgenstund).

altt udføres på side 117

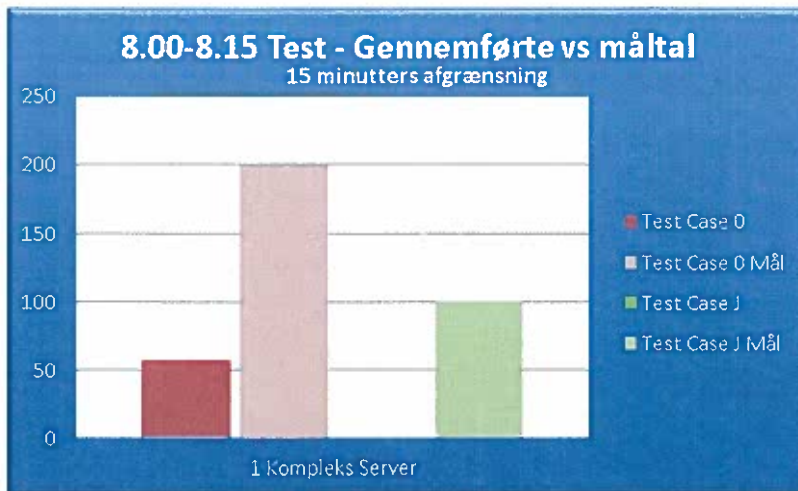
Som vi kan se af hvor mange requests systemet udfører per sekund over testens kørsel, så "sander" systemet ganske enkelt til efter få minutter. Når systemet "sander til" kan det ikke anvendes af Rigspoliets organisation før situationen er udbedret.

Nedenfor vises det tydeligt hvordan systemet er belastet til det yderste i forbindelse med kørsel af 8-8.15 testen på den simple server.

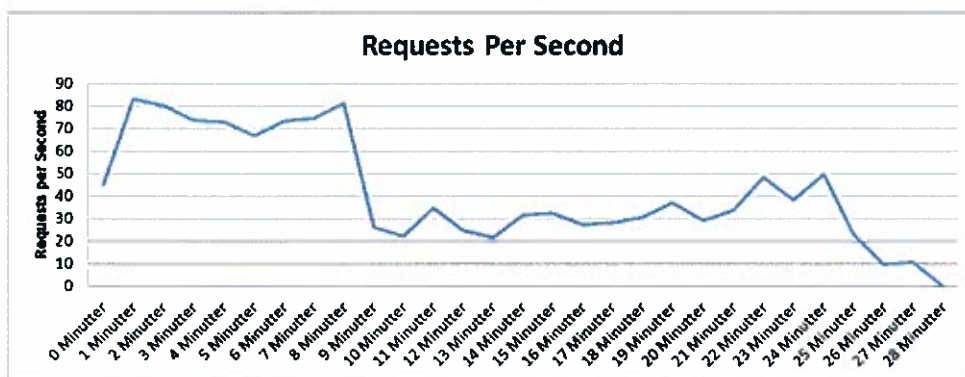


8.00-8.15 test mod 1 kompleks server

Hvis vi skifter over til at afvikle testen på en kompleks server, så ses det, at systemet begynder at få færdiggjort en brøkdelt af de påtænkte test case 0-iterationer. Serveren ligger dog stadig langt fra måltallet på 200 logins indenfor 15 minutter.

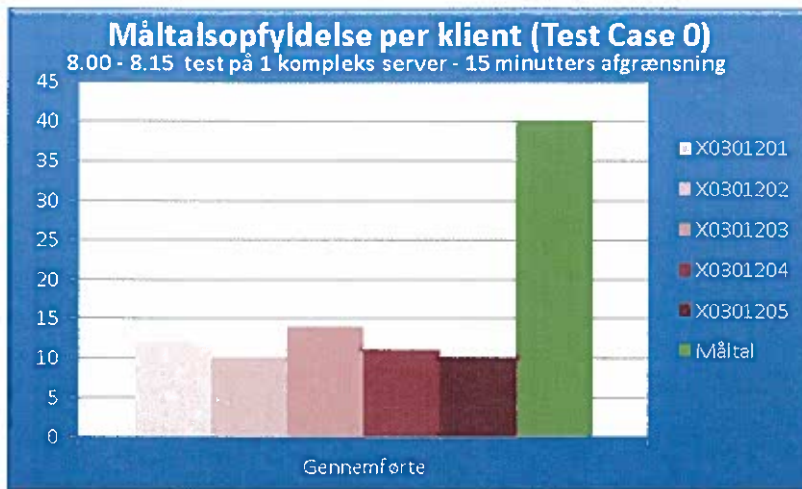


Som det fremgår herunder sker der et markant fald i antallet af requests pr. sekund som POLSAG systemet håndterer i takt med at testen kører. Systemet sender simpelthen til efter ca. 10 minutters test.

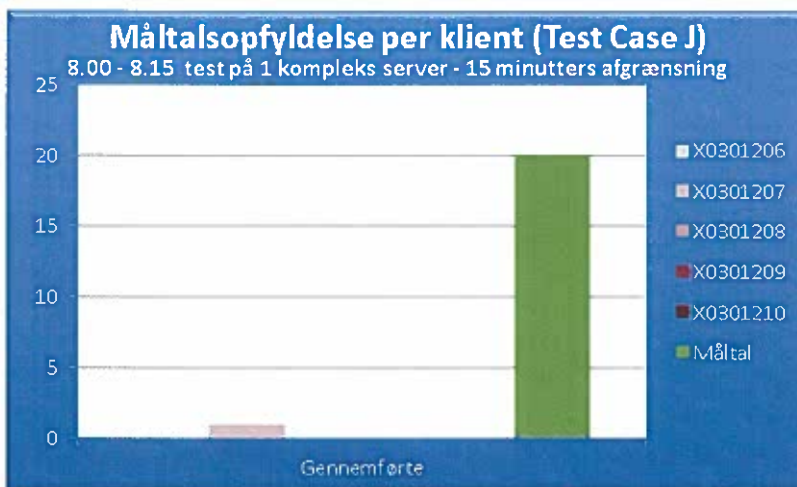


Grunden til at testen fortsætter udover de 15 minutters løbetid er, at der for 8.00-8.15 testen er at der blev afviklet et ekstra sæt iterationer for at simulere hvordan arbejdsbelastningen vil være ved afslutningen af 8.00-8.15 testen (brugermængden vil jo selvfølgelig ikke aftage i den travle morgenstund).

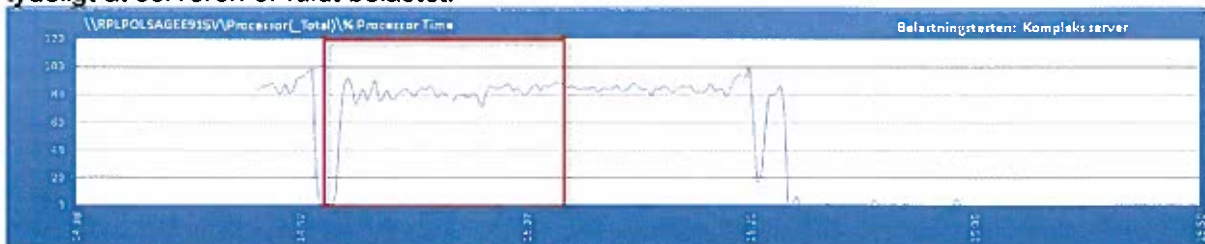
Kigger vi på hvordan de enkelte klienter håndterer 8.00-8.15 testen ses det, at antallet af succesrige iterationer er fordelt temmelig ensartet for test case 0 henover klienterne.

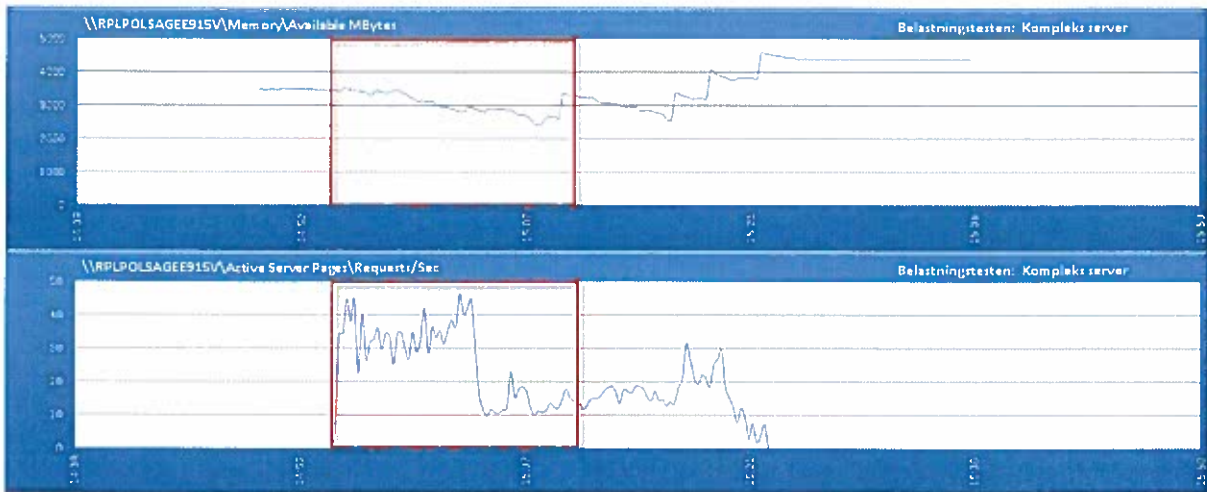


Der er til gengæld ikke meget at se for så vidt angår test case J, da det kun er lykkedes at gennemføre 1 iteration med succes.

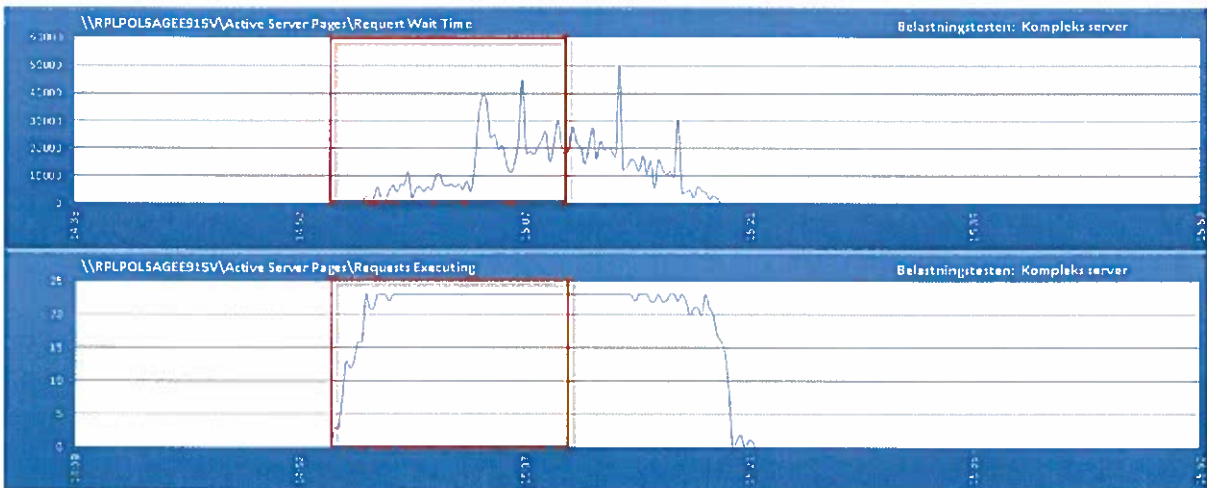


Nedenfor vises performance-nøgletallene for den komplekse server for perioden. Det fremgår tydeligt at serveren er fuldt belastet.





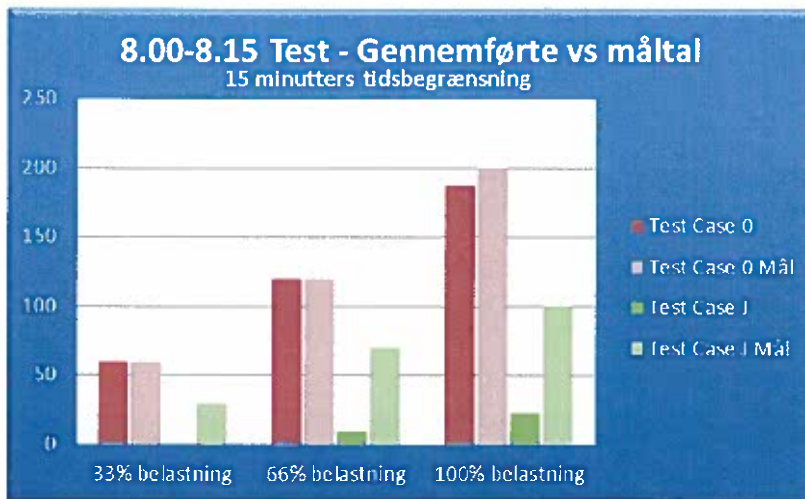
Ligeledes er det nedenfor også tydeligt, at mange forespørgsler sendt mod systemet oplever lange ventetider inden webserveren kan håndtere dem. Nedenstående graf viser hvor lang tid webserveren må lade indkommende forespørgsler vente i kø, da den er fuldt beskæftiget med at håndtere andre forespørgsler.



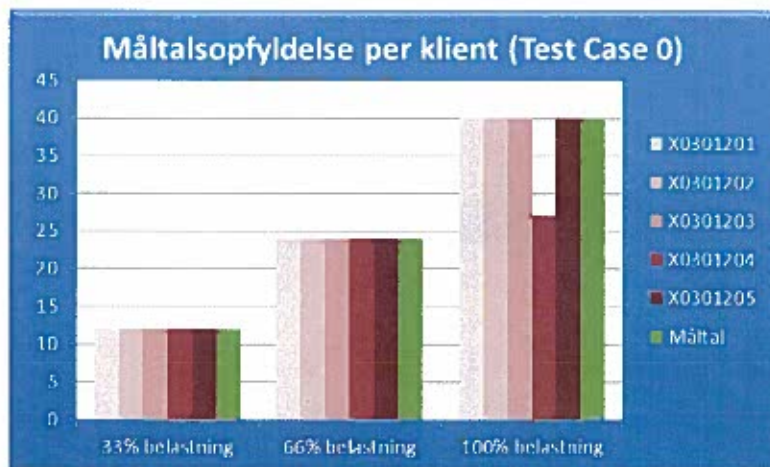
8.00-8.15 test mod NLB

Her demonstreres, hvordan 8.00-8.15 testen er blevet afviklet på NLB'et. Ligesom det er tilfældet for belastningstesten er der kørt tre "versioner" af testen mod NLB'et, hvor der lægges henholdsvis 33%, 66% og 100% belastning relativt til testen.

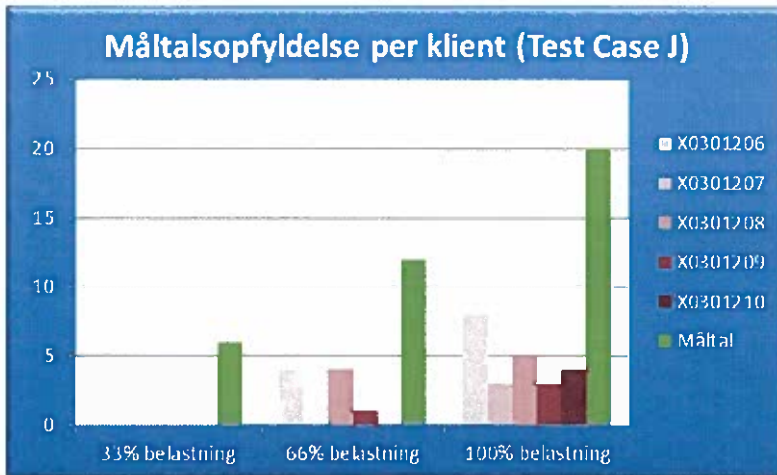
Som det kan ses nedenfor går det væsentligt bedre for test case 0 på NLB'et. Til gengæld fremgår det også tydeligt, hvordan test case J får store problemer, når systemet udsættes for pres.



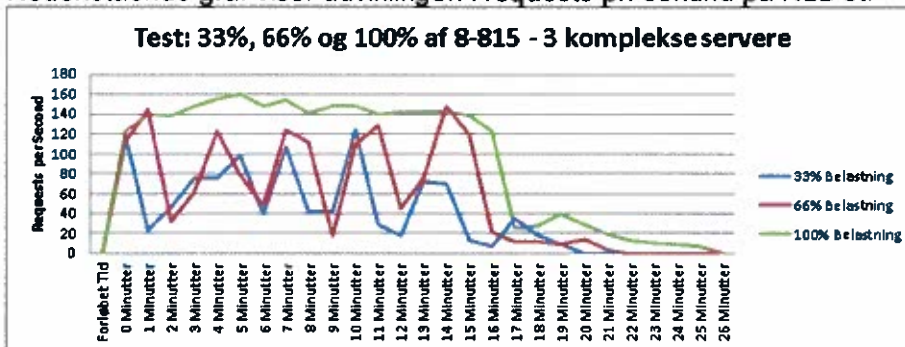
For alle belastninger gælder det, at testen har kørt stort set alle forventede test case 0-iterationer igennem uden at miste et eneste request. Kun en enkelt klient er kørt lidt i stå ved 100% belastning.



For test case J er det dog en helt anden historie. Ved 33 % belastning er ikke een iteration blevet kørt fuldt igennem og ved 66 og 100% er det kun ganske få. Det formodes at årsagen til at der er flere iterationer der kommer igennem ved den højere belastning er, at der bliver sat væsentligt flere tests i gang og der således er en større chance for at en af iterationerne når til vejs ende indenfor tidsperioden.



Nedenstående graf viser udviklingen i requests pr. sekund på NLB'et.



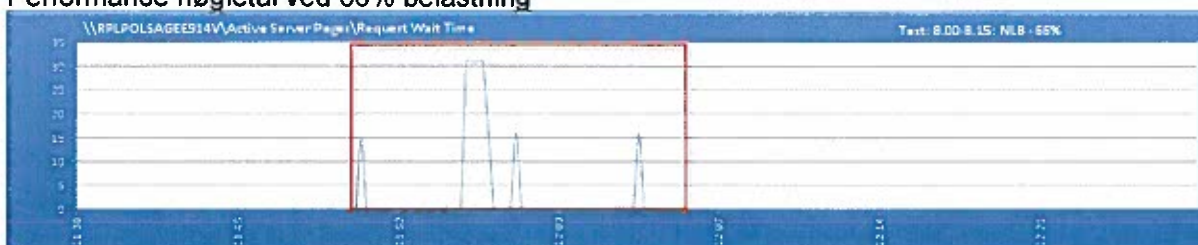
Grunden til at testen fortsætter udover de 15 minutters løbetid er, at der for 8.00-8.15 testen er at der blev afviklet et ekstra sæt iterationer for at simulere hvordan arbejdsbelastningen vil være ved afslutningen af 8.00-8.15 testen (brugermængden vil jo selvfølgelig ikke aftage i den travle morgenstund).

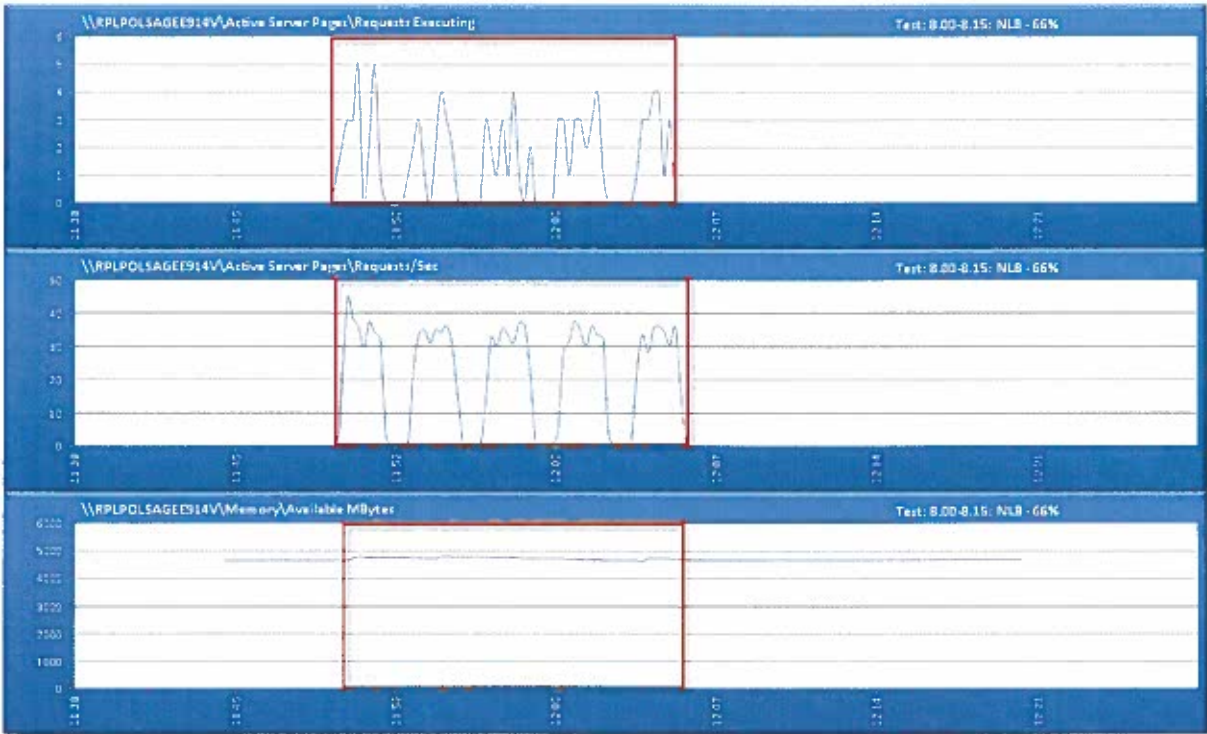
Nedenfor vises performance nøgletal for en af de servere i NLB'et, mens testen kører. Den røde boks angiver perioden, hvor testen køres.

Performance nøgletal ved 33% belastning

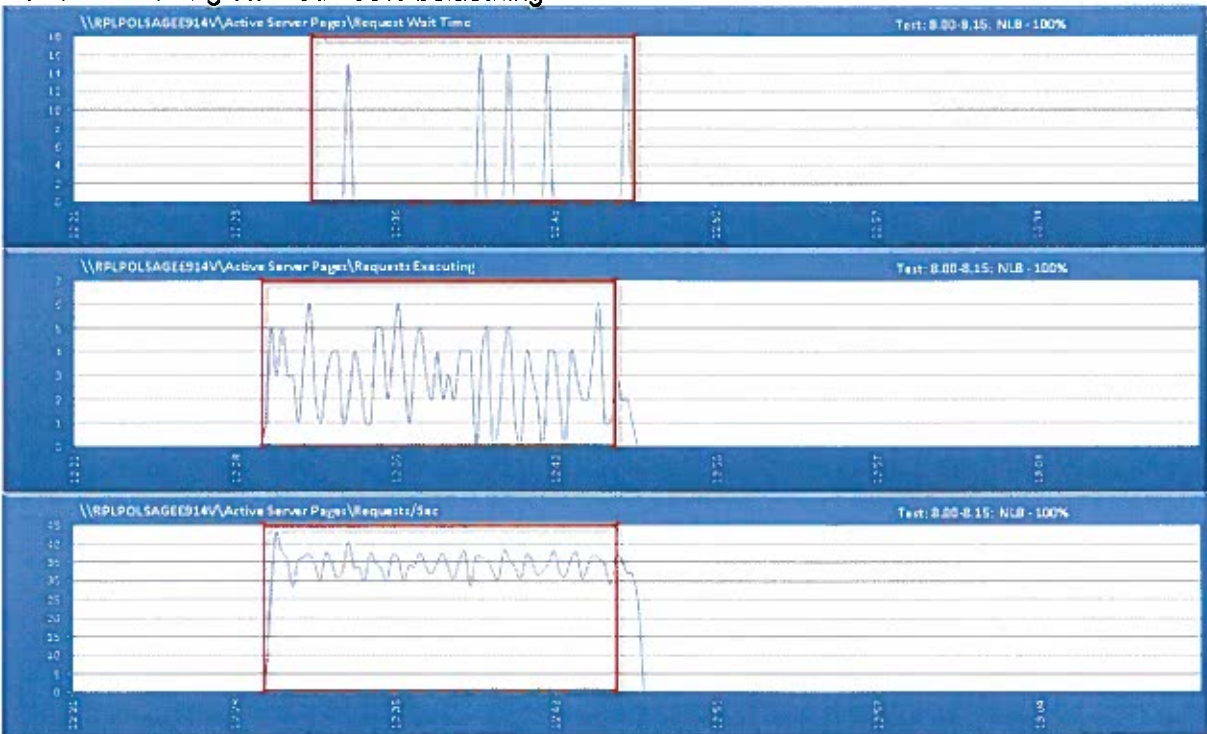


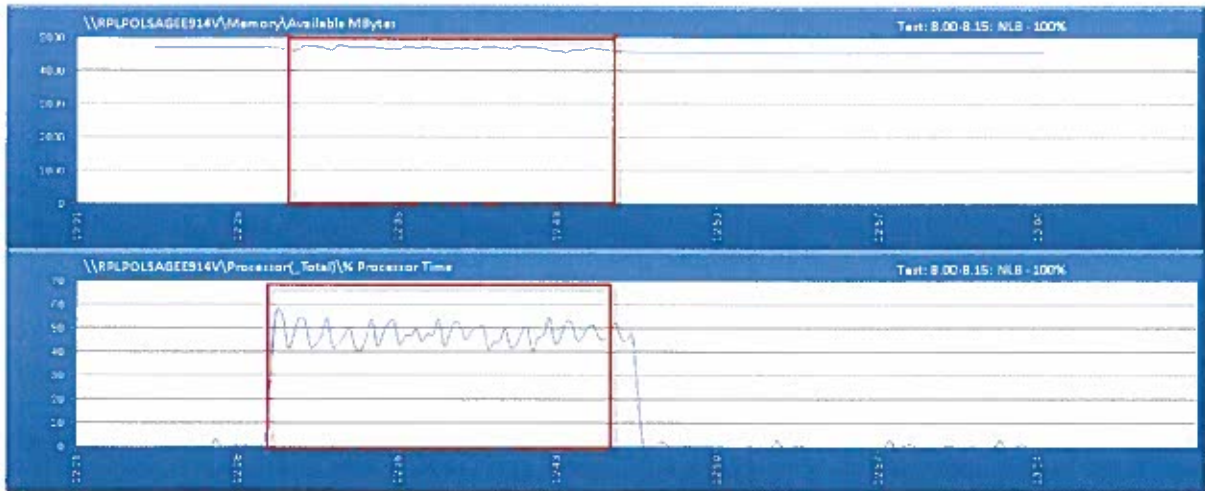
Performance nøgletal ved 66% belastning





Performance nøgletal ved 100% belastning





Stresstests

Stresstesten forsøger at presse POLSAG-miljøet ud til – og ideelt set noget over – dets grænser med henblik på at finde frem til "loftet" for systemets ydeevne. Eller rettere sagt de forskellige punkter i loftet, da der i reglen findes flere forskellige niveauer i systemets maksimale ydeevne.

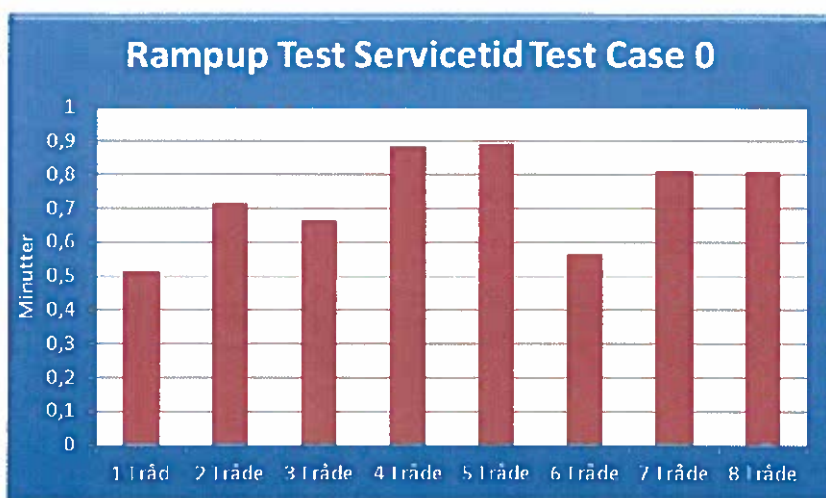
Stresstesten er baseret på de samme test cases som de øvrige tests.

Udover at påvise hvor loftet for POLSAG-miljøets ydeevne ligger, så er det også håbet at stresstesten kan tilføje information omkring systemets opførsel under høj belastning ud mod brugerne.

Stresstest med test case 0 (1 til 8 Tråde) mod NLB

I rampup-testen (som består af 8 pc-klienter, der kører op mod POLSAG-miljøet med et varierende antal tråde) med test case 0 har POLSAG-miljøet klaret sig rimeligt fint, selv ved de højere belastning. Alle iterationer er således kørt igennem uden problemer.

Der ses dog en væsentlig større variation i servicetiden end man skulle forvente.



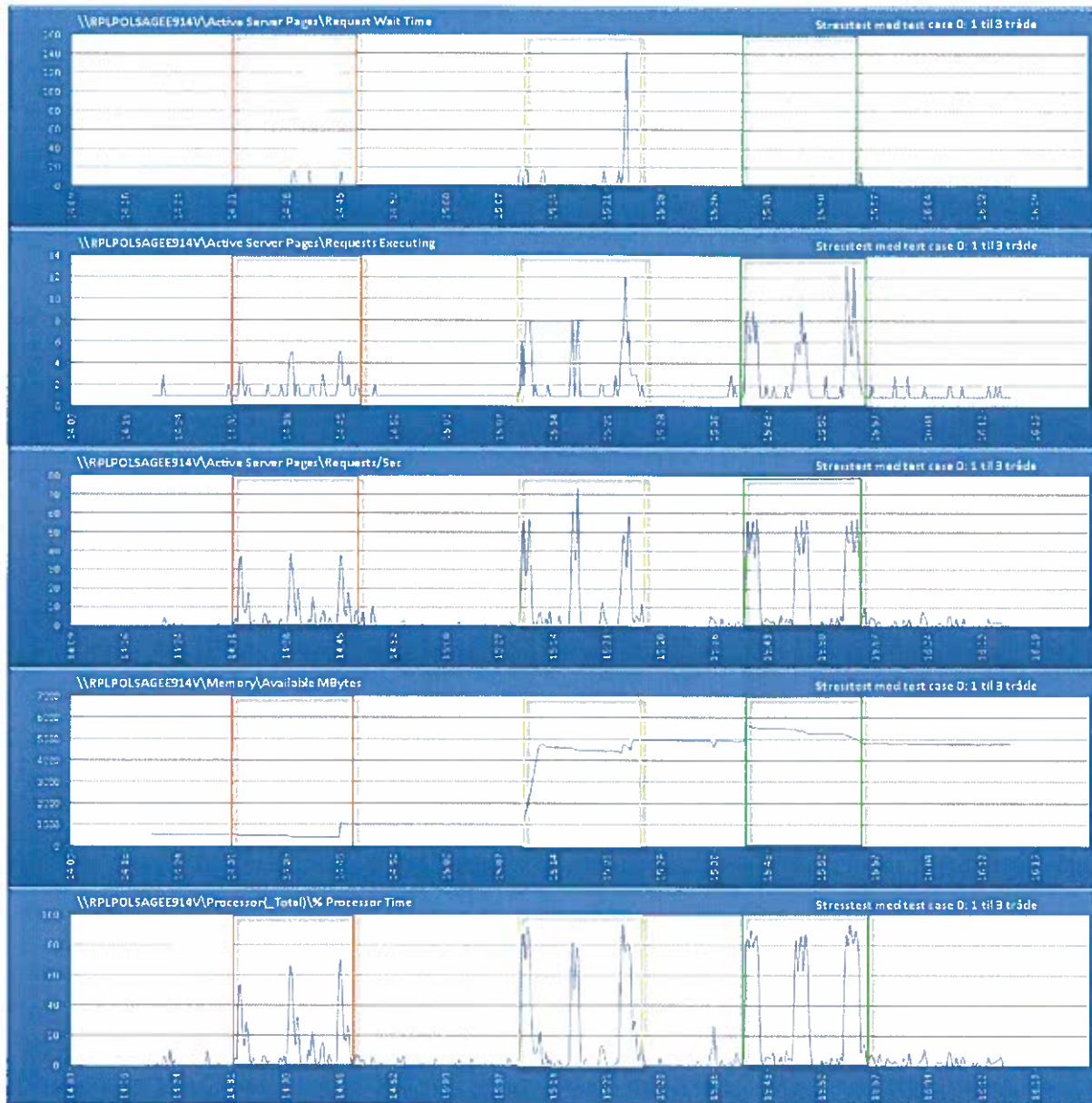
Antallet af gennemførte test case-iterationer ved de forskellige antal tråde er:

- 1 tråd: 51 iterationer
- 2 tråde: 96 iterationer
- 3 tråde: 153 iterationer
- 4 tråde: 288 iterationer
- 5 tråde: 360 iterationer
- 6 tråde: 432 iterationer
- 7 tråde: 504 iterationer
- 8 tråde: 576 iterationer

Nedenfor vises de registrerede performanceindikatorer på serverne.

Først vises testene med 1 til 3 tråde afmærket med farvede bokse, der indikerer, hvornår første afsendte og sidste modtagne succesfulde request er sendt:

- 1 tråd: Orange
- 2 tråde: Gul
- 3 tråde: Grøn



Nedenfor er vist testene med 5 til 8 tråde afmærket med farvede bokse, der indikerer, hvornår første afsendte og sidste modtagne succesfulde request er sendt.

- 4 tråde: Orange
- 5 tråde: Gul
- 6 tråde: Grøn
- 7 tråde: Rød