

IT-revisjon

Med fokus på sikkerhetsrevisjon

Versjon 1.0
15.oktober 2002

KITH Rapport 22/02
ISBN 82-7846-147-3

KITH-rapport

TITTEL

IT-revisjon

Med fokus på sikkerhetsrevisjon

KITH**Kompetansesenter for
IT i helsevesenet AS**

Postadresse

**Sukkerhuset
7489 Trondheim**

Besøksadresse

Sverresgt 15, inng G

Telefon

73 59 86 00

Telefaks

73 59 86 11

e-post

firmapost@kith.no

Foretaksnummer

959 925 496

Forfatter(e)

Bjarte Aksnes, Heine Kolltveit

Oppdragsgiver(e)

Sosial- og helsedirektoratet

Rapportnummer

R 22/02

URL

<http://www.kith.no/rapportarkiv/>

Prosjektkode

S-SV

ISBN

82-7846-147-3

Dato

15.oktober 2002

Antall sider

34

Kvalitetssikret av

Arnstein Vestad

Gradering

Godkjent av

Jacob Hygen
Adm. direktør

Sammendrag

En IT-sikkerhetsrevisjon er en intern revisjon med spesiell fokus på sikkerhet i tilknytning til IT-ressursene. Det kan være mange årsaker til at en virksomhet gjennomfører IT-revisjoner. En IT-revisjon kan være et verktøy for å hjelpe ledelsen med å ha styring og kontroll over IT-aktivitetene i virksomheten, for eksempel ved å få en uavhengig gjennomgang og kvalitetssikring av et IT-system for å forsøke å avdekke svakheter, sårbarheter eller feil. Vi foreslår derfor at IT-revisjon bør få en sentral plass i funksjoner for intern revisjon i helseforetakene.

I denne rapporten presenterer vi et enkelt rammeverk for gjennomføring av en IT-revisjon. Rammeverket består av de tre hoveddelene planlegging, gjennomføring og rapportering. Det vil også være oppfølgingsaktiviteter i etterkant av en revisjon (intern oppfølging), men dette har vi definert som utenfor selve revisjonen. Rammeverket er uavhengig av om revisjonen utføres av interne eller eksterne ressurser.

I forbindelse med IT-revisjoner vil det i en del tilfeller være aktuelt å gjøre ulike former for sikkerhetstester, f.eks. for å avsløre svakheter i nettverket eller i barrierer. Vi vil i denne rapporten også se på noen metoder og verktøy for å gjøre slike tester, og gi anbefalinger for hvordan slike tester kan brukes i forbindelse med IT-revisjoner.

Innhold

Innhold.....	2
Kapittel 1 Bakgrunn.....	4
1.1 Hvorfor gjennomføre IT-revisjoner?	4
1.2 Datatilsynet om sikkerhetsrevisjon	5
1.3 Eksempel på revisjonsobjekter	5
1.4 Risikovurdering og IT-revisjon	6
Kapittel 2 Revisjon - begreper og ulike former	7
2.1 Intern revisjon	7
2.2 IT-revisjon og IT-sikkerhetsrevisjon	8
2.3 Uavhengighet	9
2.4 Kompetanse	10
2.5 Rammeverk for IT-revisjon	10
Kapittel 3 Planlegging	12
3.1 Sette opp målsetning	12
3.2 Bakgrunnsinformasjon	12
3.3 Vurdere ressursbehovet	12
3.4 Lage revisjonsprogram	13
3.5 Informasjonsplan	13
3.6 Godkjenning av revisjonsplanen	13
Kapittel 4 Gjennomføring av IT-revisjon.....	14
4.1 Gjennomgang av dokumentasjon	14
4.2 Testing	14
4.3 Stikkprøver	14
4.4 Observasjoner	15
4.5 Intervjuer	15
4.6 Dataverktøy	15
4.7 Rammeverk for revisjon	15
NS/ISO IEC 17799	16
4.8 Gjennomføring av operasjonell revisjon (rådgivning)	16
Kapittel 5 Rapportering av funn	17
5.1 Skriftlig rapportering	17
5.2 Muntlig rapportering	18
5.3 Avslutningsmøte	18
5.4 Distribusjon	18
Kapittel 6 Egen funksjon for IT-revisjon i helseforetak	19
6.1 Helseforetakenes rolle	19
6.2 Mandat	20
6.3 Identifisere risikoområder	20
Kapittel 7 - Sikkerhetstesting.....	22
7.1 Hvorfor teste sikkerheten?	22

7.2 Ulike teknikker for sikkerhetstesting	22
7.2.1 Kartlegging av nettverket	22
7.2.2 Skanning etter sårbarheter.....	23
7.2.3 Penetreringstesting	24
7.2.4 Passord-revisjon.....	25
7.2.5 SikkerhetsTesting og -Evaluering (ST&E)	26
7.3 Prioritering	27
7.4 Verktøy	28
Vedlegg A - Noen verktøy for sikkerhetstesting	30
A.1 Nmap	30
A.2 SuperScan	30
A.3 Nessus Security Scanner	31
A.4 ISS Internet Scanner	31
A.5 Microsoft Baseline Security Analyzer	31
A.6 Netcat	32
A.7 L0pthCrack	33
A.8 John the Ripper	34

Kapittel 1 Bakgrunn

KITH har gjennom samarbeid med ulike aktører i helsesektoren erfart at IT-revisjon er et område som helsevirksomheter bør ta fatt i for å få en tilfredsstillende kvalitet på informasjonssikkerhetsarbeidet sitt, samt for å leve opp til de krav til revisjon som stilles i Datatilsynets regelverk (personopplysningsloven med tilhørende forskrift). Per i dag er det så vidt vi vet svært få helsevirksomheter som har noen praktisk erfaring i å gjennomføre IT-revisjoner, men vi tror at behovet for dette vil komme for fullt i de kommende år. Det vil være aktuelt at eksterne aktører gjennomfører IT-revisjoner, men det kan også etter hvert bli aktuelt å bygge opp egne funksjoner for IT-revisjon innenfor både de regionale og lokale helseforetakene. Det kan nevnes at de fleste banker, forsikringsselskaper, oljeselskaper og andre større selskaper etter hvert har etablert egne interne funksjoner for IT-revisjon.

Det er en del grunnleggende prinsipper som må legges til grunn ved gjennomføring av IT-revisjoner, og vi ønsker å nedfelle en del av disse i denne veiledning på området. I forbindelse med IT-revisjoner vil det i en del tilfeller være aktuelt å gjøre ulike former for sikkerhetstester, f.eks. for å avsløre svakheter i nettverket eller i barrierer. Vi vil i denne rapporten også se på noen metoder og verktøy for å gjøre slike tester, og gi anbefalinger for hvordan slike tester kan brukes i forbindelse med IT-revisjoner.

1.1 Hvorfor gjennomføre IT-revisjoner?

Det kan være mange årsaker til at en virksomhet gjennomfører IT-revisjoner. Som regel vil en IT-revisjon være et verktøy for å hjelpe ledelsen med å ha styring og kontroll over IT-aktivitetene i virksomheten. En revisjon vil ofte gjennomføres for å sjekke om virksomhetens målsetninger, policyer, retningslinjer og prosedyrer for IT-relaterte aktiviteter følges. I tillegg kan IT-revisor gi råd til ledelsen om hvilke korrigerende tiltak som bør iverksettes. Noen årsaker til at en virksomhet ønsker å gjennomføre IT-revisjoner kan være at de ønsker:

- en uavhengig gjennomgang av spesifikke IT-systemer eller forhold
- å gjøre sikkerhetsrevisjon på IT-området iht. Datatilsynets krav (jfr §2.5 i personopplysningsforskriften)
- kvalitetssikring av spesifikke IT-systemer eller IT-prosesser
- å følge opp av tidligere avdekkede svakheter eller problemområder
- å sjekke om ulike aktører følger sikkerhetspolicyen og prosedyrer
- å vurdere om leverandør(er) følger opp avtaler

1.2 Datatilsynet om sikkerhetsrevisjon

I § 2-5 i forskriften til personopplysningsloven stilles det følgende krav til sikkerhetsrevisjon:

§ 2-5 Sikkerhetsrevisjon

Sikkerhetsrevisjon av bruk av informasjonssystemet skal gjennomføres jevnlig.

Sikkerhetsrevisjon skal omfatte vurdering av organisering, sikkerhetstiltak og bruk av kommunikasjonspartner og leverandører.

Dersom sikkerhetsrevisjonen avdekker bruk av informasjonssystemet som ikke er forutsatt, skal dette behandles som avvik, jf. § 2-6.

Resultatet fra sikkerhetsrevisjon skal dokumenteres.

I kommentarene til den samme forskriften sies følgende:

til § 2-5 Sikkerhetsrevisjon

Bestemmelsen pålegger den behandlingsansvarlige jevnlig, eksempelvis årlig, å etterprøve sikkerhetsarbeidet for å verifisere at de sikkerhetstiltak som er besluttet etablert, faktisk er iverksatt og fungerer etter sin hensikt. Ved sikkerhetsrevisjon sammenlignes faktisk bruk av informasjonssystemet med de retningslinjer for slik bruk som er besluttet.

Dette viser at personopplysningsloven og Datatilsynet legger stor vekt på sikkerhetsrevisjon som en del av arbeidet med å oppnå en tilfredsstillende informasjonssikkerhet. De vektlegger spesielt samsvarsrevisjon, dvs. å kontrollere om virksomheten handler i tråd med relevante lover, standarder, policyer og prosedyrer.

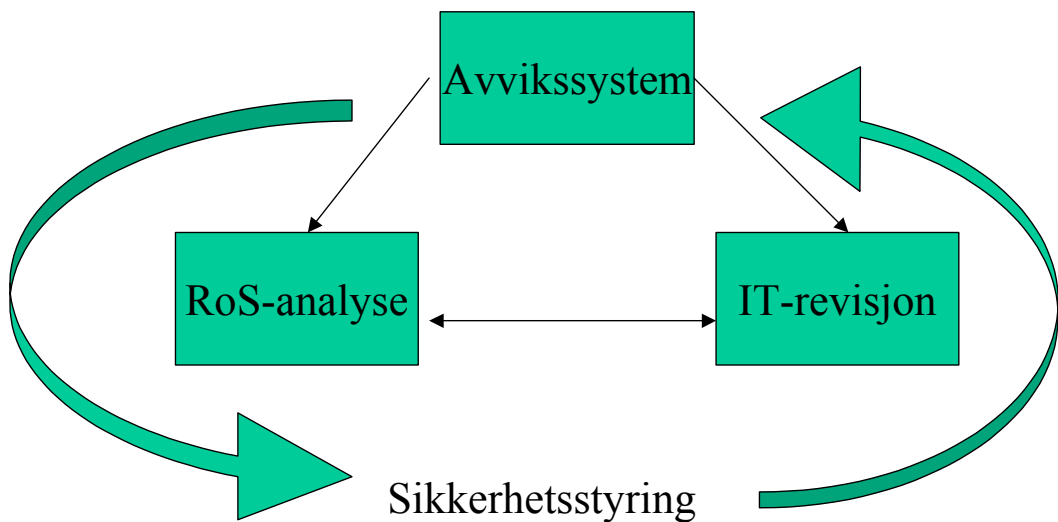
1.3 Eksempel på revisjonsobjekter

Noen eksempler på objekter som kan være gjenstand for IT-revisjon:

- Ledelse og organisering av IT-funksjonen
- Teknisk infrastruktur (operativsystemer, nettverk, databaser)
- Informasjonsressurser (logisk tilgangskontroll, viruskontroll, kryptering, digitale signaturer, brannmurer, IDS-systemer, fjerntilgang)
- Fysisk sikring av informasjonsressurser (innbrudd, brann, vann, klima, strøm, telekomlinjer)
- Sikkerhetspolicyer og prosedyrer
- Beredskaps- og kontinuitetsplaner (backupløsninger, kriseløsninger, avtaler med leverandører)
- Systemutvikling og innføring (utviklingsmetodikk, endringskontroll, prosjektstyring, metoder/verktøy, testing, implementering, vedlikehold)
- Risikostyring (risikovurdering)
- Applikasjonskontroller (autorisering, feilhåndtering, validering, integritetskontroll)

1.4 Risikovurdering og IT-revisjon

I løpet av de siste par årene har helsevirksomheter for alvor begynt å få opp øynene for Risiko- og sårbarhetsanalyser (RoS-analyser) som nyttige verktøy i forbindelse med informasjonssikkerhetsarbeidet. Etter hvert som virksomhetene har gjennomført noen analyser, har en del begynt å se seg om etter metoder for å sikre at resultatene fra ROS-analysene følges opp videre. En mulighet er å gjennomføre IT-revisjoner av de objektene som tidligere har vært gjenstand for en risikoanalyse, og IT-revisjon blir dermed et element i sikkerhetsstyringsløyfen (se figuren).



Figur 1 Elementer i sikkerhetsstyringen for en helsevirksomhet

Gjennomføring av risikovurderinger av IT-systemer og IT-revisjoner er aktiviteter som har mange fellestrekk og som også kan supplere hverandre. Risikovurderinger og –analyser er i større grad proaktive, ved at de forsøker å anslå hvilken risiko man utsetter seg for i ulike sammenhenger. En tradisjonell IT-revisjon vil vanligvis være mer reaktiv ved at den peker på avvik og problemsituasjoner som har oppstått. Men som vi også skal komme tilbake til i de neste kapitlene er det ikke noe absolutt skille mellom disse.

Ved gjennomføringen av IT-revisjoner vil også risikovurderinger benyttes, m.a. for å plukke ut revisjonsobjekter som er utsatt for høy risiko. For å utnytte knappe revisjonsressurser best mulig legges det ofte opp til en risikoorientert revisjonstilnærming, slik at man reviderer de objektene som (man tror) er utsatt for høyest risiko. En fare med en slik tilnærming er at man overser forhold som man i utgangspunktet ikke vurderer som risikoutsatt. I tillegg må IT-revisor ha et bevisst forhold til risikofaktorer ved selve revisjonen, f.eks. risikoen for å trekke feil konklusjoner eller for å ikke oppdage vesentlige feil eller mangler gjennom revisjonen.

Kapittel 2 Revisjon - begreper og ulike former

I utredningen ”Om revisjon og revisorer (NOU 1997:9)” er det m.a. sagt følgende om ulike former for revisjon:

”I Norge forekommer flere former for revisjon, - både lovbestemte og ikke lovbestemte. Disse skilles primært ved at formålene med revisjonsarbeidet kombineres forskjellig. Innledningsvis er det derfor aktuelt å sortere ut prinsipielt forskjellige formål som ligger til grunn for revisjon:

- 1. Bekrefte at årsoppgjør og andre regnskapsoppstillinger er oppsatt i henhold til gitte kriterier (AUDITS OF FINANCIAL STATEMENTS/FINANSIELL REVISJON)*
- 2. Bekrefte at organisasjon og rutiner er etablert og fungerer effektivt mot foretakets mål (OPERATIONAL AUDITS/OPERASJONELL REVISJON)*
- 3. Bekrefte at foretaket følger etablert lov, forskrifter, vedtekter og retningslinjer fastsatt av ledelsen (COMPLIANCE AUDITS/FORVALTNINGSREVISJON)*

For å gjennomføre revisjon av selskapenes årsoppgjør (se punkt 1 over) stilles det i revisorloven bestemte krav til utdanning og faglig bakgrunn. De som tilfredstiller kravene kan kalle seg ”registrert revisor” eller ”statsautorisert revisor”, som er lovbeskyttede titler. Det finnes også andre utdanninger for revisorer, og ofte vil revisorbetegnelsen knyttes til spesialfunksjoner som IT-revisor, kvalitetsrevisor o.l.

Innenfor helsevesenet vil Statens helsetilsyn og fylkeslegene gjennomføre systemrevisjoner av virksomhetenes internkontrollsystemer. En systemrevisjon er en systematisk undersøkelse for å fastslå om aktiviteter og tilhørende resultater er i samsvar med krav fastsatt i medhold av lov eller forskrift. Ved en revisjon av et internkontrollsystem er et avvik mangel på oppfyllelse av myndighetenes krav.

2.1 Intern revisjon

Den internasjonale foreningen for internrevisorer benytter følgende definisjon (The Institute of Internal Auditors Inc, 1999):

Internal auditing is an independent, objective assurance and consulting activity designed to add value to an organization's operations. It helps an organization accomplish its objectives by bringing a systematic, disciplined approach to evaluate and improve the effectiveness of risk management, control and governance processes.

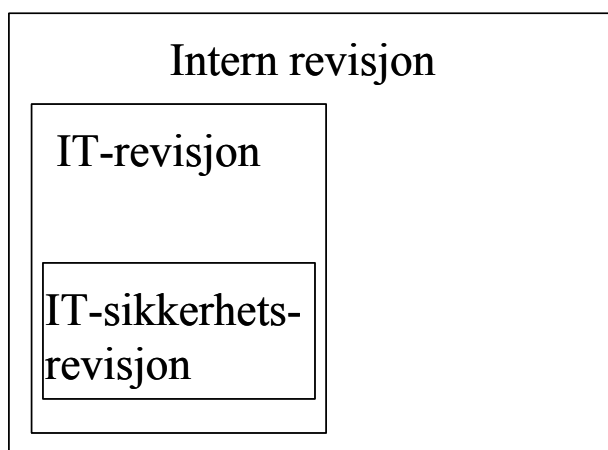
En intern revisjon er altså en uavhengig og objektiv aktivitet, som i form av kontroller, vurderinger og rådgivning skal tilføre verdi og forbedre en virksomhets aktiviteter. Intern revisjon er en del av virksomhetens ledelses- og kontrollstruk-

tur. IT-revisjon er å betrakte som en intern revisjon, selv om den utføres av personer utenfor egen virksomhet. Finansiell revisjon er en ekstern revisjon, og vil ikke bli behandlet videre i denne rapporten.

Det finnes to hovedtyper av intern revisjon; operasjonell revisjon og samsvarsrevisjon.

- **Operasjonell revisjon** har som mål å bidra til at virksomheten når sine mål, m.a. gjennom å vurdere hvor effektivt den benytter sine ressurser for å nå sine mål. Operasjonell revisjon er i hovedsak en rådgivende og *proaktiv* aktivitet.
- **Samsvarsrevisjon** (compliance) har som mål å kontrollere om virksomheten handler i tråd med relevante lover, standarder, policyer og prosedyrer. Dette er i hovedsak en vurderende og *reaktiv* aktivitet.

2.2 IT-revisjon og IT-sikkerhetsrevisjon



Figur 2 Sammenhengen mellom IT-revisjon og IT-sikkerhetsrevisjon

Figuren illustrerer at IT-revisjon er en del av internrevisjonen, og at IT-sikkerhetsrevisjon er en del av feltet IT-revisjon. Dette betyr ikke at det nødvendigvis må være på plass egne funksjoner for hhv. intern revisjon og IT-revisjon for å gjennomføre en IT-sikkerhetsrevisjon, man kan derimot ha stor nytte av å gjennomføre IT-sikkerhetsrevisjoner selv om de andre funksjonene ikke er etablert.

Det kan være ulike årsaker til at man ønsker å gjøre en IT-revisjon, en av de vanligste årsakene er at man ønsker en uavhengig gjennomgang og kvalitetssikring av et IT-system for å forsøke å avdekke svakheter, sårbarheter eller feil.

Det meste av det som beskrives i denne rapporten er relevant for gjennomføring av alle typer IT-revisjoner, inkludert spesialtilfellet IT-sikkerhetsrevisjon. Begrepet IT-revisor og IT-revisjon vil derfor i regelen brukes selv om det dreier seg om en IT-sikkerhetsrevisjon.

Noen begreper i tilknytning til IT-revisjon:

IT-revisor – en person med nødvendig kompetanse til å gjennomføre en intern revisjon med spesiell fokus på IT-ressurser

IT-revisjon – en intern revisjon med spesiell fokus på IT-ressurser

IT-sikkerhetsrevisjon – en intern revisjon med spesiell fokus på sikkerhet i tilknytning til IT-ressursene

Revisjonsobjekt – objektet som er gjenstand for revisjon, det kan for eksempel være et system, en funksjon/avdeling, en prosess, teknologi, infrastruktur, personell eller en kombinasjon av flere av disse elementene.

Kontroller – benyttes for å forhindre, oppdage/avdekke og korrigere uønskede aktiviteter eller hendelser

Ulike typer kontroller

Forebygge	<ul style="list-style-type: none">- Skal oppdage problemer før de oppstår- Forhindre at en (potensiell) sikkerhetstrussel får gjøre skade- Varsle om potensielle problemer før de gjør skade
Oppdage/avdekke	<ul style="list-style-type: none">- Skal oppdage at sikkerhetsbrudd har oppstått, slik at korrigerende eller forebyggende tiltak kan gjøres
Korrigere	<ul style="list-style-type: none">- Kontroller for å komme tilbake til en "normaltilstand" etter et sikkerhetsbrudd- Identifisere årsak til trusselen- Rette opp skader som har skjedd og minimalisere konsekvensene

2.3 Uavhengighet

En IT-revisor må være *uavhengig* og objektiv i forhold til det objektet han skal revidere. Det betyr mellom annet at han ikke bør jobbe i avdelingen som skal revideres, og han må heller ikke ha personlige interesser i forhold til det reviderte objektet. For virksomheter av noen størrelse er det ikke noe i veien for at IT-revisor er ansatt i virksomheten selv.

I den senere tid har det vært spesiell fokus på revisors uavhengighet i forhold til det å gjennomføre rådgivningsoppdrag for den samme aktøren som man utfører revisjon for (jf. Enron saken i USA). Det må her bemerkes at dette er *mest* problematisk i forbindelse med ekstern revisjon, og spesielt finansiell revisjon. I forbindelse med intern revisjon, som IT-revisjon, er det vanlig og ofte ønsket at IT-revisoren også skal kunne fungere som en rådgiver for virksomheten. Samtidig er det viktig at revisor alltid må tilstrebe uavhengighet og objektivitet i forhold til revisjonsobjektene.

Dersom revisoren har noen avhengigheter i forhold til revisjonsobjektet, eller det av andre grunner er grunn til å tro at han ikke er fullt ut objektiv, må dette gjøres

klart for oppdragsgiver (internt eller eksternt) før arbeidet startes, og også dokumenteres i rapporter og presentasjoner fra revisjonen.

2.4 Kompetanse

En IT-revisor må ha nødvendig fagkunnskap innen IT til å vurdere revisjonsobjektet, men trenger ikke nødvendigvis å kjenne alle tekniske detaljer. IT-revisoren må også ha kjennskap til prinsipper for revisjon og revisjonsmetodikk for å kunne gjennomføre en tilfredstillende revisjon. IT-revisoren vil samarbeide med de som til daglig jobber med revisjonsobjektet for å gjøre nødvendige vurderinger, men han må i tillegg være i stand til å foreta egne uavhengige vurderinger, m.a. for å unngå at de ansatte forsøker å dekke over forhold som de ikke ønsker at revisoren(e) skal få kjennskap til. Ofte vil revisjonene utføres av en gruppe som er sammensatt av personer med ulik kompetanse.

Det er mange typer kompetanse som kan være aktuell å inneha for personer som skal gjennomføre IT-revisjon, men spesielt relevant er en CISA-sertifisering. CISA står for ”Certified Information Systems Auditor”, gjerne kalt CISA sertifisert IT-revisor på norsk. En CISA-sertifisering betyr at man har dokumentert nødvendige kunnskaper gjennom å bestå en internasjonalt anerkjent eksamen arrangert av ISACA (Information Systems Audit and Control Association), samt at man har minimum 5 års relevant erfaring eller praksis innenfor området. Den norske revisorforeningen holder årlig kurs som forberedelse til denne eksamenen.

Selv om det ikke er et krav, slik som ved finansiell revisjon, vil det ofte være naturlig å knytte til seg kompetanse på IT-revisjon fra eksterne parter. Mange rådgivnings- og konsulentmiljø har nødvendig kompetanse og erfaring fra denne typen oppdrag. På sikt kan det dessuten være aktuelt for helsevirksomheter og helseforetakene å bygge opp egen kompetanse på gjennomføring av IT-revisjon (mer om dette i kapittel 6).

2.5 Rammeverk for IT-revisjon

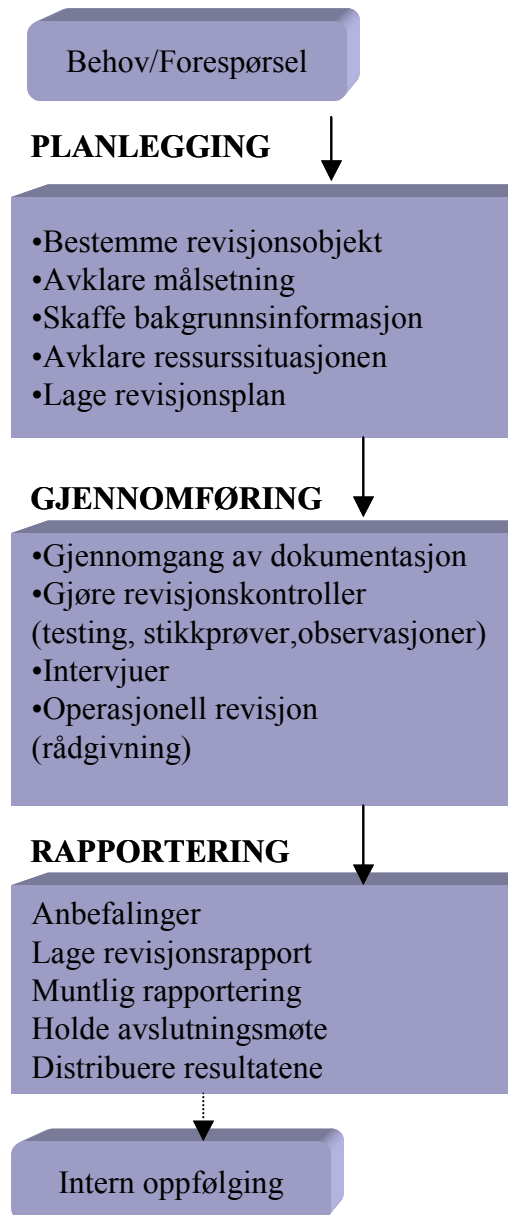
Når det gjelder rammeverk for å gjennomføre IT-revisjon finnes det en del å ta fatt i m.a. CISA sine egne rammeverk, herunder COBIT (Control Objectives for Information and Related Technology), samt Information Security Forum’s Standard of Good Practice.

I denne rapporten presenterer vi et enkelt rammeverk for gjennomføring av en IT-revisjon (se figuren). Rammeverket består av de tre hoveddelene planlegging, gjennomføring og rapportering. Det vil også være oppfølgingsaktiviteter i etterkant av en revisjon (intern oppfølging), men dette har vi definert som utenfor selve revisjonen. Rammeverket er uavhengig av om revisjonen utføres av interne eller eksterne ressurser.

I de neste tre kapitlene vil vi ta for hver av hoveddelene i rammeverket:

- I kapittel 3, Planlegging, ser vi på hva som bør gjøres av planlegging før man starter på selve gjennomføringen.
- I kapittel 4, Gjennomføring, ser vi på ulike metoder, verktøy og rammeverk som kan benyttes ved gjennomføring av IT-revisjon.

- I kapittel 5, Rapportering av funn, ser vi hva som bør inngå i rapporteringen fra en IT-revisjon.



Figur 3 Rammeverk for IT-revisjon

Kapittel 3 Planlegging

Planleggingen av en IT-revisjon er en viktig aktivitet som m.a. bør inkludere:

- sette opp målsetning for revisjonen
- sette seg inn i nødvendig bakgrunnsinformasjon
- vurdere ressursbehovet (personell, kostnader)
- informere alle som bør vite om revisjonen
- lage en plan for arbeidet (revisjonsprogram)
- finne ut hvordan og hvem som skal informeres om resultater fra revisjonen
- få planen godkjent hos oppdragsgiver eller hos ansvarlig internt

3.1 Sette opp målsetning

Det er vesentlig å avklare hva som er målsetningen med revisjonen. Før man setter opp målsetningen med revisjonen bør det avklares om oppdragsgiver (intern eller ekstern) ønsker en operasjonell revisjon eller en samsvarsrevisjon. Dersom ønsket er å utføre en operasjonell revisjon bør målsetningen fokusere på hvordan organisasjonen kan nå sine mål på en bedre måte. Er ønsket derimot å gjennomføre en samsvarsrevisjon bør man fokusere på om virksomheten handler i tråd med relevante lover, standarder, policyer og prosedyrer. Det er også mulig å tenke seg kombinasjoner av disse revisjonstypene.

3.2 Bakgrunnsinformasjon

I planleggingsfasen bør man forsøke å få en oversikt over hva som er tilgjengelig av bakgrunnsinformasjon om revisjonsobjektet.

Eksempler på bakgrunnsinformasjon kan være:

- Strategier (IT-strategi etc)
- Policyer, planer og prosedyrer
- Rapporter fra tidligere utførte revisjoner
- Rapporter fra utførte Risikoanalyser
- Organisasjonskart/stillingsbeskrivelser
- Prosedyrer og instruksjoner
- Systembeskrivelser/arkitektur og konfigurasjonskart
- Lovpålagte krav
- Intervjuer med sentrale medarbeidere

3.3 Vurdere ressursbehovet

Ut fra målsetningen for revisjonen og ressurssituasjonen må det settes opp et budsjett for revisjonen. Det må vurderes om det er tilgjengelig personalressurser med

nødvendig kompetanse i det aktuelle tidsrommet. Det må også vurderes om det er aktuelt å leie inn ekstra kompetanse fra andre. Spesielt må det legges vekt på at det både er tilgjengelig kompetanse om revisjonsmetodikk og teknisk fagkompetanse.

3.4 Lage revisjonsprogram

Revisjonsprogrammet er en slags prosjektplan for revisjonsarbeidet og bør inneholde en kort beskrivelse av de fleste emner fra dette kapittelet.

- Målsetning
- Bakgrunnsinformasjon
- Ressurser (budsjett, medarbeidere)
- Informasjonsplan/rapportering (se neste punkt)
- Metoder
- Tidsplan

3.5 Informasjonsplan

Informasjonsplanen kan være en del av revisjonsplanen, og bør inneholde en oversikt over hvem som bør informeres om revisjonen på ulike stadier som før oppstart, under arbeidet og når resultater foreligger. Det må også tenkes på hvordan denne informasjonen skal formidles, f.eks. i form av brev, e-post, møter eller personlig kontakt. Forslag til noen sentrale personer som bør informeres på de ulike stadiene:

Før oppstart: Ledelsen, IT-sikkerhetsansvarlige, ansvarlig for revisjon, ansattrepresentant, de som skal delta i revisjonen

Under arbeidet: Ledere og ansatte i berørte avdelinger, leverandører

Ved avslutning: Ledelsen, ledere av berørte avdelinger, IT-sikkerhetsansvarlig, eksterne parter (f.eks. leverandører)

3.6 Godkjenning av revisjonsplanen

Når revisjonsplanen er utarbeidet bør man sikre en formell godkjenning fra oppdragsgiver, dersom revisjonen utføres av eksterne, eller av ansvarlig for revisjonsaktiviteten internt. Dette er viktig for å sikre nødvendig forankring hos ledelsen. Dersom det finnes en intern revisjonsfunksjon kan denne få et mandat til å utføre revisjoner på eget initiativ. I andre tilfeller bør hver enkelt revisjon godkjennes av ledelsen. Manglende forankring kan føre til at det er vanskelig å få interne ressurser til å gi nødvendig informasjon eller å stille ressurser disponibelt til revisjonen.

Kapittel 4 Gjennomføring av IT-revisjon

I dette kapitlet vil vi kort beskrive hovedaktivitetene i gjennomføring av en IT-revisjon. Målsetningen med dette kapitlet er ikke å beskrive gjennomføringen av en IT-revisjon i detalj, men heller å peke på noen viktige metoder som bør vurderes.

En revisjon kan utføres på mange ulike måter. Framgangsmåten må tilpasses målsetningen for revisjonen og hvilke ressurser som er tilgjengelig, samt revisjonsobjektet og omfanget av dette. Er målsetningen å avdekke eventuelle brudd på retningslinjer vil metoden være en helt annen enn om man ønsker å finne svakheter i et system.

Det som er svært viktig er uansett at man fokuserer mest på de mest vesentlige forhold eller de som man antar utgjør størst effekt. Dette forutsetter at man vet hva som er de mest vesentlige forhold, noe som ikke alltid er tilfelle, og det vil derfor også være nødvendig å ta en bredere tilnærming for å forsøke å avdekke andre svakheter. Spesielt bør det legges vekt på kritiske komponenter der feilsituasjoner kan ha alvorlige konsekvenser.

Som regel vil det være hensiktsmessig å benytte en kombinasjon av ulike metoder.

Alle relevante funn som gjøres bør dokumenteres underveis (se også neste kapittel).

4.1 Gjennomgang av dokumentasjon

Dokumentasjon av IT-system eller prosesser kan gi mye nyttig informasjon, m.a. for å finne ut hva man bør fokusere på videre. Manglende dokumentasjon eller dårlig vedlikeholdt dokumentasjon er et viktig funn i seg selv og bør følges opp videre.

4.2 Testing

Systematisk testing av relevante forhold er en vanlig metode å bruke. Det kan noen ganger være aktuelt å benytte dataverktøy, men ofte vil testingen innebære å gå systematisk igjennom regler, policyer, prosedyrer og avtaler for å undersøke om disse er fulgt. Det vanligste er å teste for å få bekreftet at noe fungerer som forutsatt (samsvarstesting), men samtidig er det viktig å designe tester som finner eventuelle svakheter eller feil. I kapittel 7 sier vi mer om sikkerhetstesting ved bruk av dataverktøy.

4.3 Stikkprøver

Kompleksiteten i systemene eller datamengdene vil ofte være så store at man ikke har mulighet for å gjennomføre systematisk testing av alle relevante forhold. Da

kan man i stedet benytte seg av stikkprøver, der man basert på skjønn velger ut noen forhold som man ønsker å teste nærmere.

Det vil også være mulig å bruke ulike statistiske metoder som hypotesetesting og konfidensintervall når man har data som egner seg for en statistisk tilnærming.

4.4 Observasjoner

Observasjon av personer eller systemer kan være en måte å få nyttig informasjon på. For eksempel for å finne ut om resultatene er som forventet eller om prosedyrer følges. Ved observasjon av mennesker bør man være oppmerksom på at det at folk vet at de blir observert kan få dem til å endre atferden ("Hawthorne-effekten").

4.5 Intervjuer

Sentrale personer med ansvar for bruk eller drift av IT-systemer vil kunne frambringe mye relevant informasjon, og kan også peke på forhold som bør undersøkes nærmere. Det kan derfor være aktuelt å gjennomføre intervjuer med sentrale personer. Hvilke personer som er mest aktuelle vil være avhengig av revisjonsobjektet, men det kan f.eks. være:

- Strategisk ledelse
- IT-sjef/ansvarlig
- Nettverksansvarlig
- (IT)-Sikkerhetsansvarlig
- Driftspersonell
- Systemforvalter/-eier
- Superbrukere
- Vanlige brukere

4.6 Dataverktøy

Det finnes en del dataverktøy som kan benyttes i ulike deler av en revisjonsprosess, spesielt i forhold til spesifikke tekniske problemstillinger. Eksempel er verktøy som finner sikkerhetshull i brannmurer og servere, som avdekker "svake" passord, som finner mistenkelig programvare etc. (mer om dette i kapittel 7).

4.7 Rammeverk for revisjon

Til støtte i arbeidet med å gjennomføre en IT-revisjon finnes det ulike rammeverk eller retningslinjer som kan benyttes. Et eksempel på et slikt rammeverk er COBIT (Control Objectives for Information and Related Technology). COBIT er utarbeidet med støtte av ISACA (Information Systems Audit and Control Association) for å støtte gjennomføringen av IT-revisjoner. COBIT er delt inn i 34 IT-prosesser som er gruppert i fire hovedområder:

1. Planlegging og organisering
2. Anskaffelse og implementering
3. Leveranse og støtte

4. Overvåkning

For hver av prosessene er det satt opp overordnede og detaljerte kontrollmål samt retningslinjer for hvordan kontrollmålene kan revideres. COBIT er svært omfattende og alt er ikke like tilpasset norske forhold, men rammeverket kan likevel være svært nyttig ved at IT-revisor plukker ut de prosesser og kontrollmål som han finner mest hensiktsmessig i forhold til det aktuelle revisjonsobjektet. Mange virksomheter har laget sine egne rammeverk basert på COBIT eller andre modeller.

Det er også utarbeidet en rekke IT-revisjonsmanualer og sjekklister for spesifikke systemer og teknologier. Store konsulent- og revisjonsfirma har ofte utarbeidet sine egne rammeverk og manualer.

NS/ISO IEC 17799

NS/ISO IEC 17799 er en standard for informasjonssikkerhet i organisasjoner. Den bygger på en britisk standard (BS 7799) for etablering og håndtering av informasjonssikkerhet i organisasjoner. Del 1 av standarden gir en rekke anbefalinger for sikkerhetstiltak en organisasjon bør implementere. Disse tiltakene har sin bakgrunn i kjente svakheter som kan unngås ved å innføre tiltakene. Standarden kan brukes ved gjennomføring av en IT-revisjon for å sjekke om en organisasjon har håndtert ulike trusler og svakheter som er omtalt i denne.

Del 2 av standarden spesifiserer oppbygningen av et kvalitetssystem for informasjonssikkerhet, og setter krav til etableringen, vedlikeholdet og dokumenteringen av informasjonssikkerheten. Standarden beskriver seks trinn som må tas for å etablere dette systemet:

- Definerer av sikkerhetspolicy
- Bestemme hvilke deler av organisasjonen som skal inngå i systemet
- Gjennomføre risikoanalyse
- Håndtere risiko
- Velge ut hvilke elementer fra del 1 som er relevante for organisasjonen
- Begrunne valgene gjort i foregående trinn

4.8 Gjennomføring av operasjonell revisjon (rådgivning)

Operasjonell revisjon har som mål å bidra til at virksomheten når sine mål, og er i hovedsak en rådgivende og proaktiv aktivitet. Gjennomføringen av en operasjonell revisjon blir dermed av en annen karakter en tradisjonell samsvarsrevisjon. Det vil være vesentlig å være i en aktiv dialog med ledelsen og de ansatte for å finne ut hvordan virksomhetens mål kan nås på en best mulig måte. Avdekking av feil eller prosedyrer/regler som ikke følges er ikke noe mål i seg selv, hvis ikke det kan bidra til en bedre måloppnåelse. Slike revisjonsaktiviteter vil ofte være ad hoc preget basert på virksomhetens og ledelsens behov, og graden av formalisering i form av revisjonsplaner og rapporter vil være varierende. Revisoren vil i slike tilfeller ofte ha mer en rolle som prosessleder enn ekspert.

Kapittel 5 Rapportering av funn

I all revisjon står det sentralt å dokumentere de funn som gjøres. Vanligvis vil det utarbeides en revisjonsrapport, men rapporteringer trenger ikke bare å være skriftlig. Ofte kan muntlig fremføring av de viktigste funnene, med mulighet for spørsmål/diskusjon være vel så effektivt. Valg av rapporteringsform må gjøres i forhold til behovene til målgruppen og omfanget av revisjonen. Uansett er det sentralt at det skal være mulig å etterprøve de funn som er gjort.

Målsetningen med rapporteringen er å presentere de funn og anbefalinger man har kommet frem til gjennom revisjonsarbeidet.

5.1 Skriftlig rapportering

Som regel vil det utarbeides en skriftlig revisjonsrapport. Rapporten bør inneholde følgende:

- Oppsummering
- Målsetning for revisjonen (fra revisjonsplanen, eventuelt revidert målsetning)
- Beskrivelse av revisjonsobjektet og omfang
- Forutsetninger eller begrensninger
- Metoder benyttet
- Viktigste funn (mulig årsak og risiko)
- Anbefalinger for korrektive tiltak
- Forslag til videre arbeid
- Konklusjon

I tillegg bør man under revisjonen dokumentere det arbeidet som blir gjort, f.eks. i form av notater fra intervjuer, inspeksjoner og tester. Det samme gjelder dokumentasjon fra sikkerhetstester, logger el. som gjøres underveis. Relevante deler av denne informasjonen kan legges ved som vedlegg til rapporten, eller i det minste gjøres tilgjengelig på forespørsel.

Det ligger i revisjonens natur at det vil være størst fokus på negative funn. Positive funn bør også omtales, men forholdsvis kort. Negative funn må derimot omtales med beskrivelse av årsak og konsekvenser, som gir et grunnlag for å komme med anbefalinger for korrektive tiltak.

Revisjonsrapporten vil ofte inneholde konfidensiell/fortrolig eller sensitiv informasjon iht. Sikkerhetsloven eller Personopplysningsloven, eller informasjon som er følsom for virksomheten og som man derfor må ha kontroll med. Alt materialet må derfor behandles med stor forsiktighet av IT-revisoren og andre som er i besittelse med dette.

IT-revisoren er normalt pålagt taushet av oppdragsgiver om alle forhold som han får kjennskap til gjennom revisjonen. Dette vil ikke gjelde ved lovpålagt revisjon,

der revisor er pliktig til å rapportere eventuelle funn til relevante myndigheter. Revisjonsrapporten bør undertegnes av den ansvarlige for revisjonen, og rapporten må merkes med begrensninger på hvem den skal distribueres til.

5.2 Muntlig rapportering

Som regel vil det i tillegg til den skriftlige rapporteringen være fornuftig å rapportere muntlig til oppdragsgiver og andre berørte parter. På denne måten får man anledning til å kort presentere, gjerne i form av presentasjon med lysark, de viktigste funnene, samt mulighet for å avklare uklarheter og diskutere hva som kan gjøres for å foreta forbedringer. Ofte vil dette være en god måte å informere ledelsen og gi dem et eierskap til den revisjonen som er utført.

For enkelte mindre omfattende revisjoner, bør det ikke være nødvendig å lage en omfattende revisjonsrapport. Det kan klare seg med et kort notat, samt en muntlig rapportering.

For større revisjoner eller revisjoner som går over lengre tid, kan det være naturlig å også rapportere underveis, både skriftlig og muntlig. Ikke minst gjelder dette dersom det gjøres vesentlige funn. Dersom alvorlige forhold oppdages, bør de ansvarlige informeres straks. Unntaket fra dette er dersom det er mistanke om at vedkommendes som er ansvarlig selv har skyld i forholdet, og dermed kan forsøke å dekke over dette.

5.3 Avslutningsmøte

Viktige funn bør diskuteres med de reviderte og de ansvarlige før man gir ut den endelige revisjonsrapporten. Et avslutningsmøte gir de reviderte en anledning til å kommentere viktige funn i revisjonen, og reduserer faren for at man glemmer viktige forhold. Dersom det er uenighet om viktige forhold, bør dette beskrives i rapporten.

5.4 Distribusjon

Revisjonsrapporten bør distribueres til alle med vesentlige interesser i revisjonen. Dette gjelder m.a.:

- oppdragsgiver (dersom en annen enn de andre punktene)
- de som er ansvarlig for å gjennomføre korrektive tiltak
- de som har ansvaret (ledelsen) for revisjonsobjektet
- ansvarlig for intern revisjon og/eller IT-sikkerhetsansvarlig

Kapittel 6 Egen funksjon for IT-revisjon i helseforetak

Som nevnt i første kapittel er det mange virksomheter som har etablert egne funksjoner for intern revisjon. Utgangspunktet for opprettingen av slike funksjoner er gjerne at ledelsen ønsker en bedre styring med økonomiske forhold, men det har også vist seg at slike funksjoner er nyttige styrings- og kontrollverktøy for å sikre at virksomhetene når sine mål. IT-systemene får stadig en mer sentral plass i dagens virksomheter, så også innen helsevesenet. IT-revisjon bør derfor få en nødvendig plass i funksjoner for intern revisjon i helseforetakene.

6.1 Helseforetakenes rolle

Som kjent er ”helse-Norge” nå delt inn i 5 regionale helseforetak (RHF) som hver har et antall helseforetak (HF) under seg. En av utfordringene for de regionale helseforetakene er å etablere gode styrings- og kontrollsystemer for å sikre at de når sine mål på en hensiktsmessig måte.

Hver av de 5 helseregionene har i dag et samarbeid om helsenett innenfor regionen. Noen steder er dette arbeidet organisert som et eget selskap, mens det andre steder er direkte underlagt det regionale helseforetaket. Det er også en del andre samarbeidsprosjekter på IT-siden i de ulike regionene. De regionale foretakene vil legge rammene for IT-arbeidet innenfor regionen, m.a. ved å utarbeide og følge opp IT-strategi og styre IT-investeringer i regionen. Etablering av styrings- og kontrollstrukturer, f.eks. innenfor IT-revisjon vil også være en del av RHF sine oppgaver.

Selv om de regionale helseforetakene står ansvarlig for å etablere slike strukturer, trenger ikke det å bety at personer som arbeider med IT-revisjon må være direkte ansatt i RHF. Vi kan godt tenke oss en løsning der ulike personer som driver med IT-revisjon geografisk og personalmessig er tilknyttet et eller flere av helseforetakene i regionen. Det er viktig å bygge på kompetansemiljøer som finnes fra før på revisjonsområdet, for eksempel innen økonomi, kvalitet og HMS (Helse, Miljø og Sikkerhet). I tillegg vil det være behov for kompetanse på IT-systemene som benyttes. Imidlertid er det viktig å sikre at personer som skal utføre IT-revisjon er (mest mulig) uavhengige og objektive i forhold til det objektet han skal revidere (jfr. kapittel 2).

Det er imidlertid viktig at noen i det regionale helseforetaket har en overordnet ansvar for IT-revisjonsfunksjonen. Det regionale helseforetaket bør ha ansvar for følgende:

- sette av ressurser og legge rammene for aktiviteter på IT-revisjon
- plukke ut (eller ansette) ressurspersoner som skal delta i arbeidet
- være med på å plukke ut revisjonsobjekter
- være oppdragsgiver for revisjoner
- sørge for at resultater følges opp i ettertid

Det vil også være aktuelt å benytte ressurser fra eventuelle helsenettorganisasjoner og eksterne aktører/konsulenter for å gjennomføre de faktiske revisjonene. Helseforetakene må vurdere hva de selv skal bygge opp kompetanse på og hva de ønsker å kjøpe hos andre.

For å få en større grad av uavhengighet kan personer tilknyttet et helseforetak gjennomføre eller delta i revisjonen av et annet helseforetak. Det kan også være hensiktsmessig at ulike helseforetak spesialisere sin revisjonskompetanse på ulike områder, for eksempel kan ett foretak spesialisere seg på å revidere EPJ-systemet (spesielt aktuelt dersom det samme systemet benyttes i hele regionen). Enkelte mindre foretak har kanskje ikke egne ressurser til å foreta revisjoner i det hele tatt, og det kan da være nyttig å bruke ressurser fra andre foretak.

Fra Norges Interne Revisorers forening:

HVORFOR ETABLERER STØRRE FORETAK SIN EGEN INTERNREVISJON

Foretakene blir stadig større og mer komplekse og det er vanskelig for ledelsen selv å ha nødvendig kontroll og oversikt over viktige aktiviteter i de ulike nivåer i organisasjonen. Det vil derfor være behov for å etablere et internt styrings- og kontrollsystem som er tilpasset det risikobilde foretaket står overfor i forhold til å realisere sine planer og nå sine mål. Styret og den daglige leders ansvar for forvaltning av selskapet ble også betydelig skjerpet gjennom den nye aksjeloven som trådte i kraft 1. januar 1999. Gjennom å etablere en internrevisjon vil ledelsen ha et hjelpemiddel til å ivareta den kontrollfunksjonen de etter aksjelovens § 6.12 er forpliktet til å ivareta. Ved å foreta risikoanalyser, identifisere mulige risikoområder samt anbefale passende tiltak, vil internrevisjonen fungere som en konstruktiv bidragsyter og medspiller i foretaket. Styret og ledelsen kan også benytte internrevisjonen til å få tilbakemelding om det etablerte styrings- og kontrollsystem er effektivt og hensiktsmessig samt følges opp og fungerer som forutsatt.

6.2 Mandat

For helseforetak som vil utføre IT-revisjoner, må ledelsen for helseforetaket gi revisjonsfunksjonen, eller den personen som er utpekt til å utføre dette, et mandat til å utføre IT-revisjoner. Mandatet må inneholde beskrivelse av ledelsens mål for IT-revisjon, samt hvilket ansvar og myndighet som gis til IT-revisjonen.

Når det benyttes eksterne aktører for å gjennomføre revisjoner, bør det etableres en egen avtale for dette, som sier noe om de samme forholdene.

6.3 Identifisere risikoområder

En viktig oppgave for IT-revisjonsfunksjonen, enten den er lokalisert til et regionalt eller et lokal helseforetak, er å identifisere risikoområder i tilknytning til IT-systemene. En best mulig utnyttning av knappe revisjonsressurser tilsier at ressursene bør benyttes på de områdene som er utsatt for størst risiko. I denne sammenheng bør man ikke begrense seg til å se på risikovurderinger som er utført ift. IT-systemer, fordi manuelle systemer/rutiner også kan påvirke risikobildet til IT-systemene. Tidligere utførte risikovurderinger eller ROS-analyser kan gi nyttige innspill til denne prosessen. Men IT-revisjonsfunksjonen må også gjøre en totalvurdering av helsevirksomhetens aktiviteter og avhengigheten og risikobildet ift. bruk av IT-systemer. Dette kan gi innspill til nye områder som bør være gjenstand for IT-revisjon. Dersom det ikke tidligere er utført risikovurderinger innenfor området, kan IT-revisjonen gjerne starte med å utføre en enkel risikovurdering for å dokumentere risikoforholdene bedre.

Dersom man har etablert en egen IT-revisjonsfunksjonen må denne med jevne mellomrom vurdere og planlegge hvilke IT-revisjoner som bør utføres i nærmeste fremtid. Samtidig må funksjonen ha før en kontinuerlig oversikt over endringer i virksomhetens systemer og risikobildet og være beredt til å foreta endringer i planer og kompetanse basert på disse endringene.

Kapittel 7 - Sikkerhetstesting

I forbindelse med gjennomføringen av sikkerhetsrevisjoner vil det ofte være behov for å utføre tester for å forsøke å avdekke svakheter med dagens løsninger. I dette kapitlet vil vi gå mer inn på ulike metoder for testing, samt komme med noen eksempler på hvordan slike tester kan utføres og verktøy som kan benyttes.

7.1 Hvorfor teste sikkerheten?

Den fremste grunnen for å teste sikkerheten er for å identifisere potensielle sårbarheter og for å reparere disse. Dette gjøres for at de ikke skal kunne utnyttes av ondsinnede hackere (crackere) som ønsker tilgang til systemene. Hackerne benytter seg ofte av det faktum at organisasjoner ikke retter opp slike sårbarheter. Det store flertallet av Internett-angrepene er muliggjort av et lite antall programfeil. Hackere utnytter ofte disse kjente feilene med tilgjengelige angrepsverktøy for å slippe å gjøre det ekstra arbeidet det er å finne nye feil. De satser altså på at organisasjonene ikke fikser problemene etter hvert som de blir funnet og dokumentert på nettet.

Testing er en sikkerhetsaktivitet som brukes for å vurdere sikkerheten til et system i forhold til en organisasjons sikkerhetskrav og sikkerhetspolicy. Bruk av testing gir også organisasjonen muligheten til å se nettverket sitt fra en annen synsvinkel enn vanlig, noe som gir ekstra innsikt i eget nettverk og en fordel i forhold til sikkerhetsavgjørelser i fremtiden.

7.2 Ulike teknikker for sikkerhetstesting

Det er mange forskjellige typer av sikkerhetstesting. Flere av disse metodene kan kombineres for å oppnå en mer utfyllende vurdering av sikkerheten. Et første trinn i en slik sikkerhetstesting kan være å kartlegge nettverket.

7.2.1 Kartlegging av nettverket

Her bruker man en portskanner for å identifisere aktive vertsmaskiner som er koblet til nettverket. Så finner man hvilke porter som er åpne på hver vert og da også den sannsynlige tjenesten som tilbys (f.eks: port 21(FTP)/port 22(SSH)/port 80(HTTP)/osv...). Det er også mulig å identifisere programvaren som tilbyr tjenesten bak den åpne porten (f.eks IIS/Apache for HTTP, osv). Ved å se på resultatet (en liste med aktive verter, åpne porter og programvare som kjøres) av denne testen kan man blant annet si hvilket operativsystem som brukes. Denne informasjonen kan senere brukes av uautoriserte personer til å trenge seg inn i systemet. En begrensning ved portskannere er at selv om de kartlegger og identifiserer aktive vertsmaskiner, åpne porter, tjenester, programvare og operativsystemer, så sier de ingenting om hvilke sårbarheter systemet kan ha.

Organisasjoner bør kartlegge nettverket mellom annet for å:

- Finne uautoriserte verter koblet til organisasjonens nettverk.
- Identifisere sårbare tjenester

- Identifisere avvik fra tillatte tjenester definert i sikkerhetspolicyen.
- Forberede sårbarhetstesting- eller penetreringstesting (se dette)

Etter en kartlegging av nettverket bør man dokumentere resultatene og rette opp eventuelle avvik ved å:

- Koble fra uautoriserte verter
- Slå av eller fjerne unødvendige og sårbare tjenester
- Modifisere sårbare verter ved å begrense tilgangen til sårbare tjenester til et begrenset subsett av verter som må ha tilgang.
- Modifisere virksomhetens brannmur til å begrense tilgangen til kjente sårbare tjenester

7.2.2 Skanning etter sårbarheter

Sårbarhetsskannere tar nettverkskartlegging et steg lenger. Den identifiserer ikke bare verter og åpne porter, men bruker resultatet til automatisk å finne tilhørende sårbarheter i stedet for å bruke menneskelig tolkning. Dette gjøres ved å sammenligne med kjente sårbarheter for de identifiserte operativsystemer og applikasjoner. Sårbarhetsskannere kan konfigureres til å automatisk rette opp feil og fikse enkelte oppdagede sårbarheter. Dessverre har sårbarhetsskannere ofte en høy positiv feilrate (rapporterer ikke-eksisterende feil) og i en test utført av Network Computing ble de vanligste sårbarhetsskannerne testet på 17 kjente sårbarheter på 5 forskjellige operativsystemer. *Ingen* av dem fant *alle* feilene. Dette viser oss at det vi ikke kan stole 100% på noen av verktøyene.

Sårbarhetsskannere gir følgende muligheter:

- Identifisere aktive verter på nettverket
- Identifisere aktive og sårbare tjenere/porter på verter
- Identifisere applikasjoner og operativsystem
- Identifisere sårbarheter i forbindelse med de aktuelle operativsystemer og applikasjoner.
- Teste hvor godt verts-applikasjoner fungerer i forhold til sikkerhetspolicyen
- Etablere et grunnlag for penetreringstesting.

Sårbarhetsskanning krever en relativ stor arbeidsinnsats med høy grad av menneskelig involvering med tolkning av resultater. Etter gjennomført kartlegging bør følgende gjøres:

1. Dokumentere resultatene og rette opp eventuelle avvik.
2. Oppgrader og reparer sårbare systemer for å tette identifiserte sikkerhetshull.
3. Minimer sikkerhetshullene hvis de ikke kan lappes helt (f.eks hvis en applikasjon ikke vil kjøre hvis man oppdaterer operativsystemet).

4. Stram inn konfigurasjonsstyring (program og prosedyrer) for å sikre regelmessig oppdatering av systemet.
5. Endre organisasjonens sikkerhetspolicy, arkitektur eller annen dokumentasjon for å sikre at sikkerhetsmetodene inkluderer de riktige oppdateringene.

7.2.3 Penetreringstesting

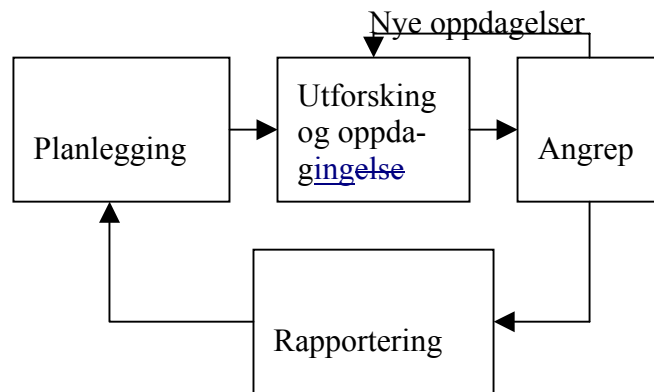
Penetreringstesting gjennomføres ved bruk av vanlige verktøy og teknikker som ofte er utviklet av hackere. Målet med denne testingen er å identifisere metoder for å få tilgang til et system. Det anbefales spesielt for komplekse eller kritiske systemer (det vil si de fleste organisasjoners nettverk). Det eksisterer en viss mulighet for at testingen kan føre til at systemer blir skadet. Denne risikoen kan minskes ved å bruke erfarne penetreringstestere, men elimineres aldri helt.

Penetreringstesting simulerer et virkelig angrep og bruker diverse verktøy og teknikker som kan være ulovlige i henhold til eksisterende lover, vedtekter og organisatoriske policyer. Derfor er det absolutt nødvendig å hente inn skriftlig tillatelse for testing før man setter i gang. Denne tillatelsen bør inneholde:

- Hvilke IP-adresser som skal sjekkes/ikke sjekkes
- Hvilke testeteknikker og verktøy som kan brukes
- Tidspunkt da testingen kan utføres
- Berøringspunkt for både penetrasjonstestingsgruppen, målsystemene og nettverkene.
- Forholdsregler for å unngå at eksterne (f.eks. politi) blir tilkalt dersom angrepet oppdages
- Håndtering av informasjonen som testgruppen samler.

Penetrasjonstesting kan foregå åpenlyst (Blue Teaming) og/eller skjult (Red Teaming). Den første måten innebærer at IT-staben vet om og samtykker til testingen, mens den andre foregår uten at IT-staben vet om det på forhånd, men med *full* viten og tillatelse fra den øvre ledelsen. Red Teaming kan bli utført både med og uten forvarsel.

Det er to nivå av penetreringstesting; intern og ekstern. Ekstern testing foregår fra utsiden av brannmurene og uten noen annen reell informasjon om målsystemet enn IP-adresser, og man må forsøke å skaffe mer informasjon før selve angrepet. Ved intern testing får man som regel tildelt en bruker og prøver å opparbeide seg mer rettigheter ved å utnytte svakheter i systemet. Man må ikke sitte fysisk på innsiden for å foreta en intern test. Det er nok å logge seg på en maskin på det lokale nettverket og dette kan godt foregå fra utsiden av brannmuren.



Figur 4 - De fire fasene i penetreringstesting

Penetreringstesting kan deles opp i faser:

1. I *planleggingsfasen* blir grunnarbeidet med identifisering av regler, godkjenning fra ledelsen og bestemmelse av målene for testen gjennomført. Først i den neste fasen begynner selve testingen.
2. Neste fase er *utforsking og oppdagelse*. Den starter med kartlegging av nettverket før man bruker sårbarhetsanalyse på resultatene. Sårbarhetsanalysen kan enten gjennomføres automatisk, med en sårbarhetsskanner, eller manuelt.
3. Selve *angrepet* er hjertet i enhver penetreringstest. Hvis et angrep er vellykket er sårbarheten verifisert og beskyttelsestiltak må identifiseres og vurderes implementeres for å lappe sikkerhetshullet. Ofte fører ikke et forsøk på å utnytte en sårbarhet frem, men man lærer likevel mer om mål-systemet. Det er årsaken til tilbakekoblingen til den andre fasen.
4. *Rapporteringsfasen* skjer parallelt med de andre fasene. Man dokumenterer hva som gjøres hele veien, og etter testen er fullført skrives en overordnet testrapport som beskriver de identifiserte sårbarhetene, gir en risikoevaluering og kommer med veiledning om hvordan de oppdagede sårbarhetene skal minskes/fjernes.

Å utføre regelmessige penetreringstester er vesentlig for å holde oversikten over hvor sårbar nettverket er og størrelsen på skadene som kan oppstå er. Etter hver test må korrigerende tiltak vurderes iverksatt, f.eks. :

- Stenging av utnyttede sårbarheter
- Modifisere en organisasjons sikkerhetspolicy og prosedyrer
- Avholde sikkerhetskurs for personell.

7.2.4 Passord-revisjon

Mange brukere velger passord som er lette å knekke fordi de består av ord som finnes i ordlister, eventuelt med et par tegn i tillegg, eller fordi de består av få

tegn. Denne typen passord kaller vi svake passord, og det blir derfor viktig å kjøre en passord-revisjon for å finne ut om virksomheten har for svake passord.

Passord blir generelt sett lagret og oversendt i en kryptert form som man kaller hash. Dette er resultatet av en enveis ikke-reversibel funksjon og kan ikke knekkes. Disse hashene blir så brukt til å sammenligne med hasher programmet lager. Det er mye lettere å knekke et passord om man har fått tak i en hash, enn om man må kjøre gjennom passordsjekk-funksjonen hver gang. Funksjonen har vanligvis mottiltak for å forsinke og stoppe angrep ved f.eks å legge inn en tidsforsinkelse for hver gang og å kun tillate 3 forsøk før man blir stengt ute. Det er tre forskjellige hovedangrepsmåter for å finne svake passord:

- Ordbok angrep - Man bruker en spesiell ordbok man finner på nettet eller annet sted til å teste alle vanlige ord i de aktuelle språk
- Hybrid angrep - Denne bruker ordbokmodellen, men bygger på den ved å legge tall og symboler til vanlige ord. (p@ssord, h4ckmeg, passord2, passord&%, osv)
- Fullskala angrep – Systematisk testing av alle kombinasjoner blir generert og testet. Dette tar lang tid, men som regel tar det kortere tid enn den tiden de fleste sikkerhetspolicyer spesifiserer for passordbytte.

Ved å spre søket etter rett passordhash over flere maskiner kan hackere og penetreringstester redusere søketiden etter et passord kraftig.

7.2.5 SikkerhetsTesting og -Evaluering (ST&E)

ST&E er en inspeksjon eller analyse av sikkerhetsoppsettet som er brukt av et system når det er fullstendig integrert og operasjonelt. Målene med denne typen gjennomgang er:

- Avdekke design, implementasjon og operasjonelle feil som kan forårsake brudd på sikkerhetspolicyen
- Avgjøre om sikkerhetsmekanismer og lignende er tilstrekkelige for å håndheve sikkerhetspolicyen
- Vurder konsistensen mellom systemdokumentasjonen og implementasjonen

ST&E planen spenner over følgende sikkerhetsemner; data (internett), kommunikasjon, trådløs, fysisk, personell, administrativ og operasjon. De som er aktuelle å ta opp i dette dokumentet er de to første.

Datasikkerhet omfatter m.a. de tiltak og foranstaltninger som beskytter systemet mot DoS (Denial of Service) angrep og uberettiget lesing, modifikasjon eller sletting av data i systemet. Dette kan deles opp i konfigurasjonstesting (sammenligne den godkjente konfigurasjonen med den som er installert) og driftstesting (eksekvere predefinerte tester i et operasjonelt miljø for å avgjøre om sikkerhetspolicyen blir håndhevd).

Kommunikasjonssikkerhet består av de tiltak og foranstaltninger som skal forhindre uberettiget aksess til nettverket. Man sjekker her at kommunikasjonslinjene

er beskyttet på et nivå som er i samsvar med sensitivitetsnivået på dataene som blir overført.

7.3 Prioritering

Sikkerhetstesting bør bli utført i implementasjons-, drifts- og vedlikeholdsfasene av systemets livssyklus. Typen tester som bør utføres, og hvor ofte er avhengig av livssyklusen, kostnaden og innvirkning på driften hvis sårbarheter blir utnyttet. Dette gjør at vi må dele opp testkravene i to deler; minimum obligatorisk testing og omfattende sikkerhetstesting.

Minimum obligatorisk testing er de testene som bør bli utført som et absolutt minimum. Det er viktig at alle organisasjoner, til og med de med begrensede ressurser, holder et visst nivå på sikkerheten. Minimum testing i implementasjonsfasen involverer gjennomføring av ST&E, mens i drift og vedlikeholdsfasene involverer det de fleste testeteknikkene.

Omfattende sikkerhetstesting betyr å utføre *alle* typer tester regelmessig. Dette kan være både dyrt og tidkrevende og en del organisasjoner vil ikke ha ressurser til å gjennomføre dette. Omfattende testing i implementasjonsfasen involverer ST&E og penetreringstesting, mens i de to andre fasene skal alle tester utføres med en viss frekvens.

På grunn av forskjellig tilgang på ressurser og ulikt behov for sikkerhet må man prioritere forskjellig for hver enkel organisasjon og system. En prioriteringsprosess som kan brukes er denne:

- Identifisere og rangere system sensibilitet og kritikalitet
- Organisasjonens systemer må kartlegges og rangeres etter hvor viktige de er
- Utføre konsekvensanalyse
- Konsekvensanalysen vurderer hvor alvorlig en skade på systemet er hvis en identifisert sårbarhet blir utnyttet, og så vurdere hvor mye det vil skade organisasjonens drift.
- Kostnadsanalyse

Kostnaden blir bestemt av flere faktorer:

- Størrelsen på systemet som skal testes (WAN/LAN/enkel database/stor applikasjon)
- Kompleksiteten på systemet som skal testes (å teste et nettverk for en stor organisasjon med heterogent operativsystem vil koste mer enn et lite nettverk med kun en type operativsystem)
- Mengden med menneskelig interaksjon som kreves for hver test
- Om det er mulig å velge ut noen stikkprøver å teste på (vanskelig ved kartlegging av nettverket, men ved penetreringstesting er det mulig)
- Identifisere fordeler av hver testtype per system

- For å sikre at kostnadene for testing ikke overskrider verdien av testingen bør fordelene identifiseres og kvantifiseres mest mulig. Kostnaden må vurderes mot verdien av ny kunnskap om systemet og minsket sjanse for vellykket inntrengning eller sammenbrudd i systemet.
- Prioritere systemer for testing. Resultatene av de foregående stegene bør evalueres og rangeres for å prioritere systemene for testing. Resultatet av denne analysen bør gi ei liste av system og tester, i synkende orden, sortert etter kritikalitet, innvirkning og gevinster. Listen vil også inneholde hvor mange ressurser (kostnader) de forskjellige testene krever. Så må kostnadene sammenlignes med de tilgjengelige ressursene. Hvis ikke det er nok ressurser tilgjengelig for å kunne utføre minimum obligatorisk testing for kritiske system, bør ekstra ressurser skaffes for å sikre at en slik test kan gjennomføres. Resultatet av dette siste steget er en prioritert liste med system som skal testes med tilhørende testeteknikk og frekvens.

Det må også vurderes om det skal brukes interne eller eksterne ressurser til å utføre testene. Fordelen med å bruke interne er at de har større kunnskap om systemet og kan utnytte denne i testingen. På den andre siden kan de eksterne ha større kompetanse og lengre erfaring på området og de er også uavhengige. Hvis en som har ansvaret for den daglige driften eller sikkerheten i et system skal utføre en test, så vil ofte resultatet bli påvirket av dette og dermed ikke bli uavhengig.

7.4 Verktøy

Det finnes flere forskjellige verktøy for å utføre og støtte de forskjellige testene. Noen eksempler på slike verktøy er:

Kartlegging av nettverket

- Nmap – portskanner (Linux / Win)
- SuperScan – portskanner (Win)

Skanning etter sårbarheter

- Nessus Security Scanner – sårbarhetsskanner (Linux / Win (kun klient))
- ISS Internet Scanner – sårbarhetsskanner (Win)
- Microsoft Baseline Security Analyzer (Win)

Penetreringstesting

- Netcat – leser og skriver TCP/UDP (Linux / Win)

Passord-revisjon

- L0phtCrack – passordcracker (Win)
- John the Ripper – passordcracker (Win)

Denne listen er ikke utfyllende, og gir heller ingen anbefalinger for hvilke verktøy som bør benyttes. Den er kun ment som eksempler på noen verktøy som vi har testet, mer om disse verktøyene i vedlegg A. Uansett hvilke verktøy som benyttes må det vises forsiktighet og personene som utfører tester må ha relevant kompetanse for å gjennomføre tester og vurdere resultater.

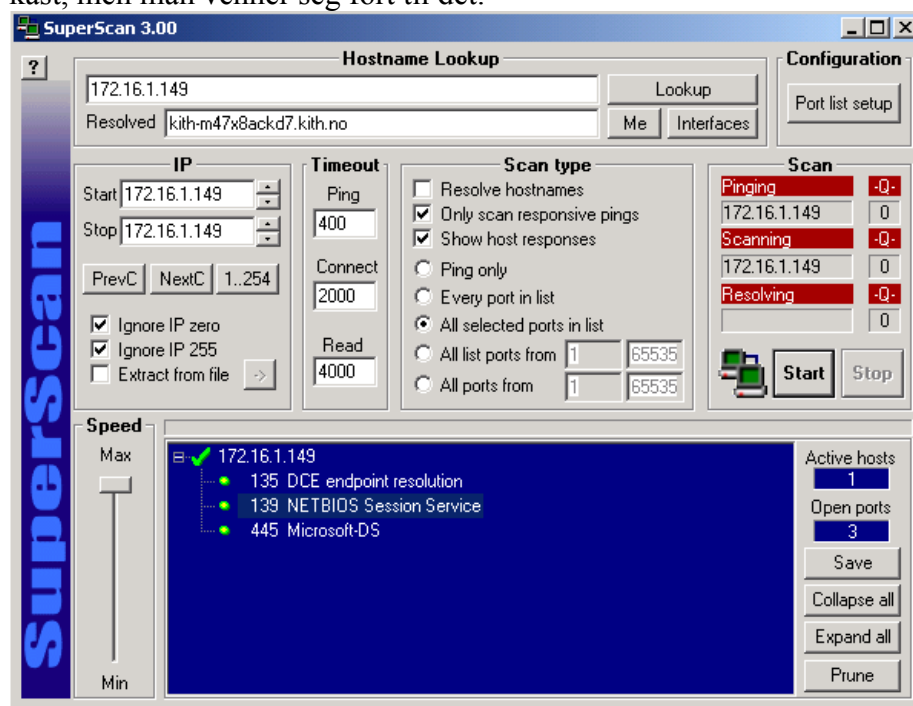
Vedlegg A - Noen verktøy for sikkerhetstesting

A.1 Nmap

Dette er den desidert mest fleksible og brukte portskanneren som er tilgjengelig. Dessverre er den kommandobasert og er derfor litt treg å lære seg. Den finnes i to versjoner; nmap (Linux) og nmapNT (Win). Disse to skal ha akkurat samme funksjonalitet så kun nmap blir drøftet her.

A.2 SuperScan

SuperScan er laget for Windows. Den utfører kun vanlig ping og TCP connect() portskanning og er dermed mye mer begrenset i sin bruk enn nmap. Den har et grafisk grensesnitt (se figur 5) som kan virke ganske komplekst ved første øyekast, men man venner seg fort til det.



Figur 5 - SuperScan med resultater

I "Hostname Lookup" boksen skriver man inn ip-adressen eller maskinnavnet til den man vil skanne. For å velge flere ip-er kan man legge inn i "IP" boksen. Hvor mange millisekund det skal gå før det blir en timeout velges i "Timeout" og skannemetoden velges i "Scan type". Når man skanner vises det hvor man ligger an i prosessen rett over det blå vinduet og hva som skjer i "Scan"-boksen. I "Port list setup" kan man sette opp nøyaktig hvilke porter som skal sjekkes, lagre og åpne

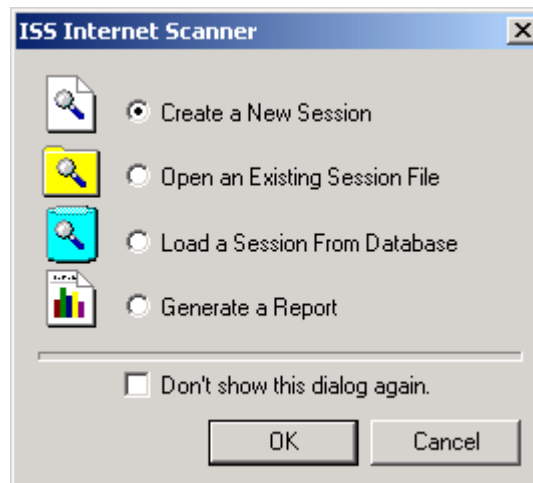
portlister og sette opp informasjon om hver port. Resultatet kommer opp i det blå vinduet og inneholder kun hvilke porter som er åpne på hver vert og hvilken tjeneste som sannsynligvis ligger bak.

A.3 Nessus Security Scanner

Dette er en sårbarhetsskanner som bruker klient/server modellen. Dette gjør at en sentral server kan ta seg av all skanning, mens klientene kun overvåker og gjennomgår resultatene på distribuerte maskiner. Serveren finnes kun på Linux, mens klientene også kan kjøres på Windows. Hele programmet har åpen kildekode i tillegg til at det er muligheter for å lage egne sårbarhetssjekker ved bruk av NASL (Nessus Attack Scripting Language).

A.4 ISS Internet Scanner

Dette er kommersiell sårbarhetsskanner for Windows.



Figur 6 - Meny ved oppstart av ISS Internet Skanner

Internet Scanner kommer med ei stor hjelpefil som beskriver enkelt og greit hvordan man skal gå frem for å sette opp ei økt, utføre den og få skrevet ut rapporter. Når man starter programmet får man opp en meny (se figur 6) med alternativer for hva man vil gjøre.

Internet Scanner utfører de valgte testene og undersøker de forskjellige kommunikasjonstjenestene, operativsystemene, hovedapplikasjoner og rutere. Når programmet skanner finner den de mest vanlige sårbarhetene og sikkerhetshullene og foreslår en løsning på hvordan disse skal tettes. Rapportene kan tilpasses målgruppen slik at de styrende kun får en overordnet sikkerhetsrapport, mens teknikerne får all tilgjengelig informasjon om systemet.

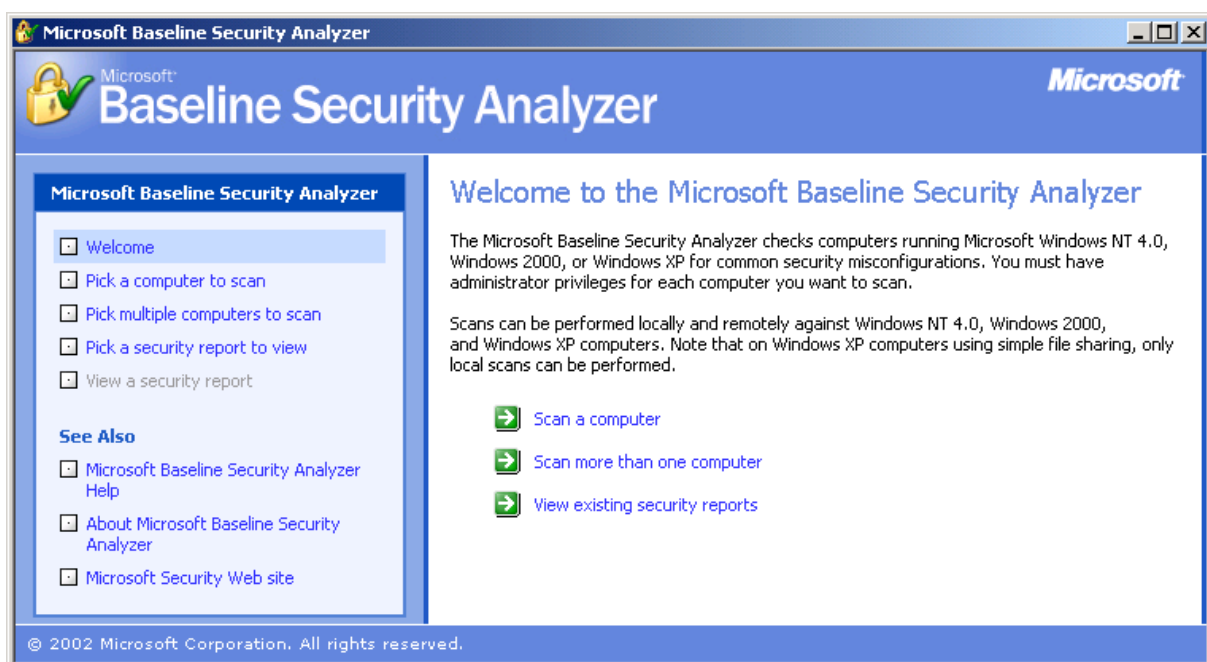
A.5 Microsoft Baseline Security Analyzer

Dette er Microsofts egen sårbarhetsskanner for Windows. Versjon 1 kjører på Windows 2000 og Windows XP. Den sjekker følgende Windows-produkt for sår-

- Noen verktøy for sikkerhetstesting

barheter: Windows NT 4.0, Windows 2000, Windows XP, Internet Information Server (IIS) 4.0 og 5.0, SQL Server 7.0 og 2000, Office 2000 og 2002 og Internet Explorer (IE) 5.01 og nyere. Dessverre er MBSA foreløpig bare tilgjengelig for engelsk Windows-versjon. Dersom den kjøres på norsk versjon kan den gi feilaktige rapporter på om at en sikkerhetsfiks mangler selv om den er installert.

Dette programmet er veldig intuitivt å bruke (se figur 7). Det eneste er at det virker som om det har problemer med å bruke maskinnavnet til å slå opp IP-adressen og at denne dermed må legges inn manuelt. (Ved å skrive ping <maskinnavn> i et kommandovindu får man tak i IP-adressen.) Det er også mulig å laste ned patcher på Microsoft sine sider under ”Windows Update”:



Figur 7 - MBSAs velkomstsvindu

<http://v4.windowsupdate.microsoft.com/no/default.asp>.

A.6 Netcat

Netcat for NT er et kommandobasert TCP/IP multiverktøy. De vanligste funksjonene er:

- Utgående eller inngående tilkobling, TCP eller UDP, til eller fra enhver port
- Bruke hvilken som helst lokal ut-port
- Innebygde muligheter for portskanning i tilfeldig rekkefølge
- Kan lese argumenter fra kommandolinjen fra standard input
- Sakte modus; sende en linje hvert N-te sekund
- Muligheter for å la et annet program betjene etablerte tilkoblinger
- Kan kjøre i bakgrunnen uten kommandovindu

- Muligheter for å restarte som en singeltråd-tjener for å håndtere en ny tilkobling

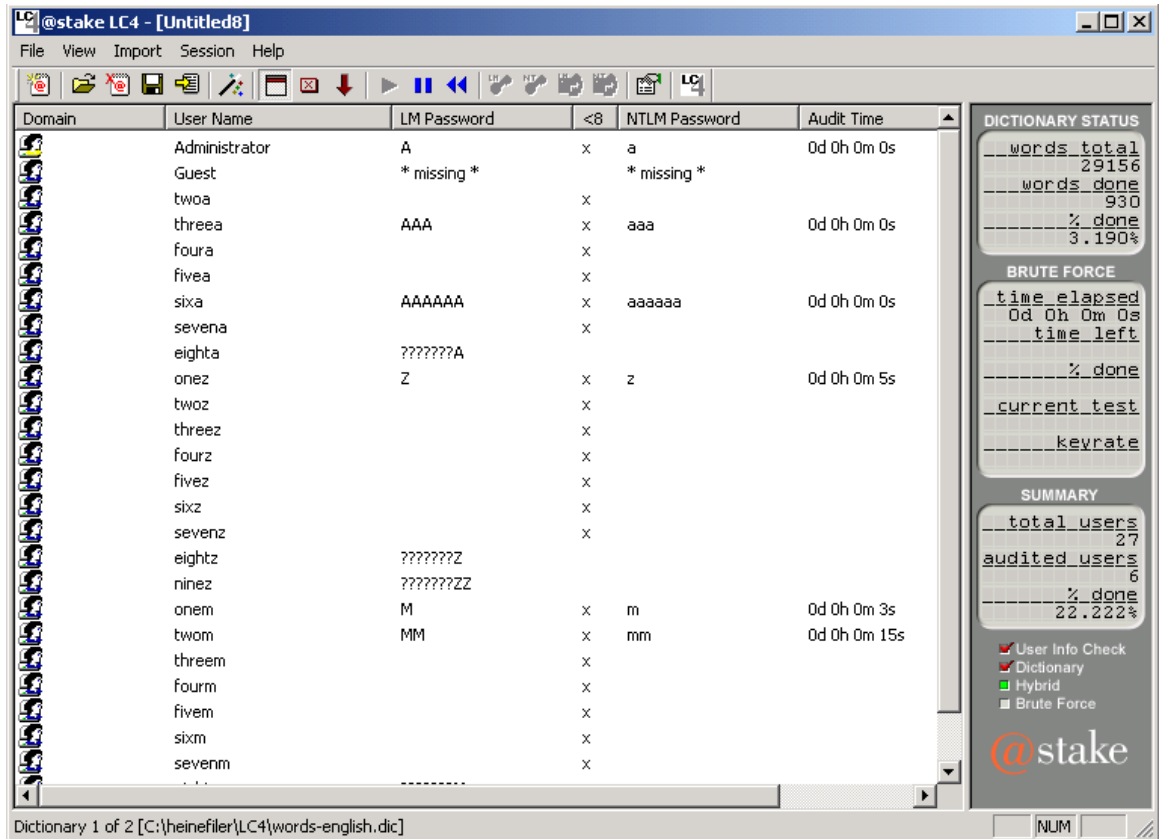
Den kan koble seg mot hvilken som helst port på hvilken som helst maskin med kommandoen `nc <www.website.com> <port>`. Man får da ofte ned et banner som sier noen om hvilken tjeneste som kjører bak (den åpne) porten. For å få opp mer informasjon om en tilkobling kan `-v` flagget brukes i tillegg.

Netcat kan også kjøres "foran" tjenester som allerede eksisterer på maskinen. Et eksempel er NETBIOS Session Service som kjøres på port 139 på NT maskiner med fildeling. Ved å binde til ei spesifikk kildeadresse gis Netcat prioritet over NETBIOS tjenesten fordi den er bundet til alle IP-adresser. Dette gjøres med `-s <kilde_ip>` noe som også fører til at fildelingen ikke vil fungere lenger.

A.7 L0pthCrack

L0pthCrack knekker Windows NT og Windows 2000 passord. Den kommer med grafisk grensesnitt (se figur 8) og en egen veiviser for uerfarne brukere (File - LC4 Wizard). Versjon 4 tilbyr flere måter å få tak i hasher av passord (Importmenyen). Henting av krypterte passord fra frittstående maskiner, tjenere, primære domene-kontrollere eller Active Directory, både med og uten SYSKEY installert, er støttet. Den kan også lytte til nettverkstrafikken og på den måten finne krypterte passord som blir sendt over nettverket når en maskin autentiserer seg for en annen. Støtte for de tre forskjellige hoved angrepsmåtene finnes (ordbok, hybrid og fullskala) og man kan også velge de ordbøkene man skal bruke (Session - Session Options...). Det kan eksporteres en tekstfil av resultatene som lett importeres i Excel og lignende verktøy. LC4 koster \$350 for hver vanlige maskinlisens, men hvis den skal brukes i konsulentvirksomhet er prisen \$1750 per konsulent.

- Noen verktøy for sikkerhetstesting



Figur 8 - Skjerm bilde av eksempelkjøring av L0phtCrack

A.8 John the Ripper

Dette er en kommandobasert passordcracker for både Windows og Linux passord. Den finnes på begge typer operativsystem. Man kan velge ordbok og angrepsmetode ved hjelp av argumenter. Det kan defineres egne fremgangsmåter å teste passord på og resultatene kan skrives til fil.

```
John the Ripper Version 1.6 Copyright (c) 1996-98 by solar Designer

Usage: john [OPTIONS] [PASSWORD-FILES]
-singl                    "single crack" mode
-wordfile:FILE -stdin    wordlist mode, read words from FILE or stdin
-rules                    enable rules for wordlist mode
-incremental[:MODE]      incremental mode [using section MODE]
-external:MODE           external mode or word filter
-stdout[:LENGTH]         no cracking, just write words to stdout
-restore[:FILE]          restore an interrupted session [from FILE]
-session:FILE            set session file name to FILE
-status[:FILE]           print status of a session [from FILE]
-makechars:FILE          make a charset, FILE will be overwritten
-show                    show cracked passwords
-test                    perform a benchmark
-users: [-]LOGIN|UID[,..] load this (these) user(s) only
-groups: [-]GID[,..]     load users of this (these) group(s) only
-shells: [-]SHELL[,..]  load users with this (these) shell(s) only
-salts: [-]COUNT       load salts with at least COUNT passwords only
-format:NAME             force ciphertext format NAME (DES/BSDI/MD5/BF/AFS/LM)
-savemem:LEVEL          enable memory saving, at LEVEL 1..3
```

Figur 9 - John the Rippers kommandoer. Vises hvis man skriver **john**.

Litteratur

- Information Systems Audit and Control Association, CISA review technical information manual, 2001
- Computer Security, DRAFT Guideline on Network Security Testing, Recommendations of the National Institute of Standards and Technology, John Wack & Miles Tracey, NIST Special Publication 800-42
- Open-Source Security Testing Methodology Manual, Peter Vincent Herzog, The Ideahamster Organization
- Vulnerability Assessment Scanners, Jeff Forristal & Greg Shipley, Network Computing
- Breaking into Computer Networks from the Internet, Roelof Temmingh, SensePost Ltd
- The Art of Port Scanning, Fyodor, Insecure.org
- Glossary of Vulnerability Testing Terminology, Department of Electrical and Information Engineering, University of Oulu, Finland
- Computer Security, Dieter Gollmann, John Wiley & Sons