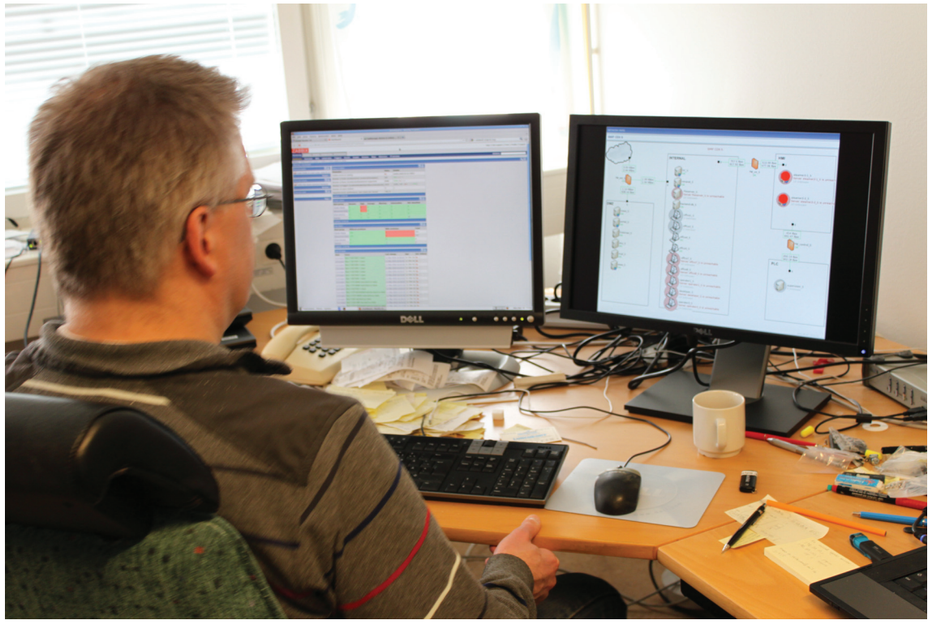
Detektering av IT-attacker

Intrångsdetekteringssystem och systemadministratörens roll

TEODOR SOMMESTAD, KRISTOFFER LUNDHOLM



FOI
Totalförsvarets forskningsinstitut
164 90 Stockholm

Tel: 08-55 50 30 00
Fax: 08-55 50 31 00

www.foi.se

FOI-R--3419--SE
ISSN 1650-1942

Mars 2012

Teodor Sommestad, Kristoffer Lundholm

Detektering av IT-attacker

Intrångsdetekteringssystem och systemadministratörens roll

Titel	Detektering av IT-attacker: Intrångsdetekteringssystem och systemadministratörens roll
Title	Detection of IT attacks: Intrusion detection systems and the role of the system administrator
Rapportnr/Report no	FOI-R--3419--SE
Månad/Month	Mars/March
Utgivningsår/Year	2012
Antal sidor/Pages	40 p
ISSN	1650-1942
Kund/Customer	Försvarsmakten
FoT område	Ledning och MSI
Projektnr/Project no	E36005
Godkänd av/Approved by	Lars Höstbeck
Ansvarig avdelning	Informations- och aerosystem

Sammanfattning

Intrångsdetekteringssystem övervakar datorsystem i syfte att upptäcka attacker som görs mot dem. I driftsatta installationer genereras vanligtvis alarm för misstänkta händelser, och en systemadministratör inspekterar denna alarmlista för att skapa sig en överblick över annalkande hot och pågående attacker. Ett problem med dagens intrångsdetekteringssystem är den stora mängd falsklarm som de genererar. Erfarenheter från praktisk användning av intrångsdetekteringssystem pekar på att systemadministratören har en viktig roll att spela när intrångsdetekteringssystem används. Erfarenheten är att systemadministratören behövs för att filtrera bort falsklarm genom att korrelera olika alarm och tolka dem med hjälp sin expertkunskap.

Denna rapport beskriver ett experiment som gjorts under 2011 med syftena att (1) undersöka detektionsförmågan i en intrångsdetekteringslösning som motsvarar industristandard och (2) att analysera systemadministratörens roll i detektionsprocessen. Experimentet bekräftar tidigare forskning som pekat på problem med en stor andel falsklarm. Experimentet visar också att andelen falsklarm kan minskas utan att i större uträckning påverka andelen upptäckta attacker om alarmen filtreras baserat på expertkunskap om det attackerade datornätet, datornät i allmänhet, attacker i allmänhet och hur hotbilden ser ut.

Nyckelord: IDS, Intrångsdetektering, Intrångsdetekteringssystem, IT-attacker, IT-försvar, Systemadministratör, Experiment

Summary

Intrusion detection systems monitor computer systems in order to detect attacks performed against them. In operational installations they typically produce alarms for suspicions events, and a system administrator inspects these alarms to obtain situational awareness concerning immediate threats and ongoing attacks. An issue with contemporary intrusion detection systems is the large amount of false alarms they produce. Experiences from practical applications of intrusion detection systems point to the important role that the system administrator plays. The experience is that the administrator is needed to filter the alarm-list by correlating alarms and interpret them with the help of his/her expert judgment.

This report describes an experiment performed in 2011 with the purposes to (1) investigate the detection rate of "state-of-practice" solutions, and (2) to analyze the role of the system administrator in the detection process. The experiment confirms earlier results in the field concerning a high percentage of false alarms. The experiment also shows that the percentage of false alarms can be decreased without a great impact on the detection rate if the alarm-list is filtered using expertise concerning: the attacked computer network, computer networks in general, attacks in general, and expertise concerning the existing threat environment.

Keywords: IDS, Intrusion detection, Intrusion detection system, IT attack, IT defense, Experiment

Innehållsförteckning

1 Inledning

Denna rapport presenterar analysresultat och erfarenheter från ett experiment genomfört inom ramen för projektet ”Spaning och motmedel på informationsarenan” (SMIA). Projektet är en del i Försvarsmaktens samlingsbeställning inom forskning och teknikutveckling (FoT).

1.1 Bakgrund

Att veta när ett system är utsatt för IT-attacker är väsentligt för att kunna försvara systemet. Ett sätt att upptäcka att ett system är utsatt för en attack är att använda ett intrångsdetekteringssystem. Ett intrångsdetekteringssystem övervakar händelser och tillstånd i ett IT system med hjälp av sensorer och avger alarm då attacker eller felanvändning misstänks. Intrångsdetekteringssystem beskrivs ofta som ett komplement till preventiva åtgärder och skydd. En stor mängd forskningsartiklar har producerats inom detta område de senaste åren. Den absoluta majoriteten av dessa är av teoretisk natur och presenterar huvudsakligen nya algoritmer för att detektera attacker, men testar inte dessa i någon realistisk miljö. Överlag finns det få empiriska resultat som indikerar om, eller när, intrångsdetekteringssystem är effektiva som säkerhetsåtgärd.

Under 2011 har FOI kombinerat ett antal mogna programvaror för intrångsdetektering med programvara för aggregering och visualisering av alarm. Denna kombination av programvaror har använts vid ett experiment där övervakade nät attackerades varpå empiriska resultat erhölls. Syftet var att undersöka potentialen i vanliga intrångsdetekteringslösningar och få erfarenheter som är till nytta då området utforskas vidare i kommande års experiment. Denna rapport presenterar resultat från experimentet i form av en analys av insamlad data samt en beskrivning av erfarenheter från användandet av intrångsdetekteringssystem för att övervaka system under attack.

1.2 Rapportens innehåll

Syftet med denna rapport är att presentera resultaten från de experiment som gjorts och av de data som samlats in samt att diskutera lämpliga förbättringar inför nästa experiment. Rapporten är indelad i fyra delar.

Kapitel 2 introducerar läsaren till vad ett intrångsdeteketringssystem gör samt hur den fungerar. Kapitlet presenterar vilka kvaliteter som kan mätas för att avgöra hur bra ett intrångsdetekteringssystem fungerar samt vilka faktorer som påverkar dessa.

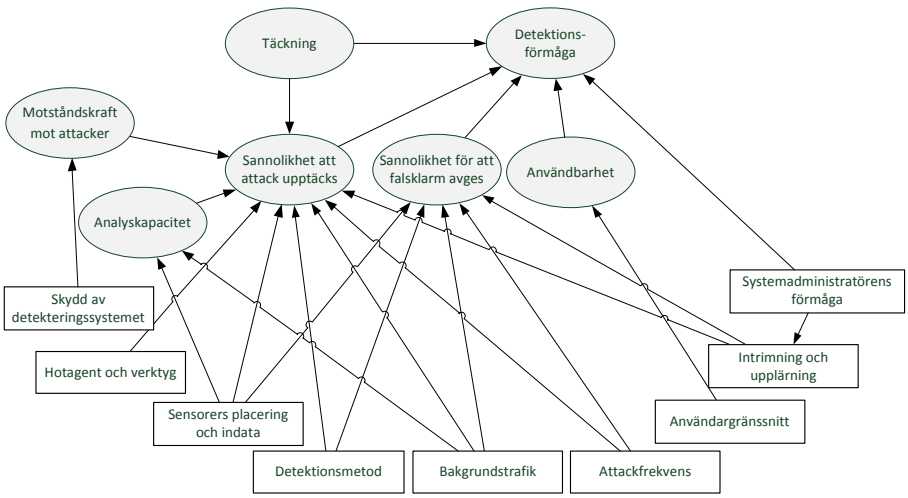
Kapitel 3 presenterar systemet som användes i experimentet och beskriver hur olika faktorer hanterades vid experimentet.

Kapitel 4 innehåller beskrivningar av de analyser som gjorts baserat på data som samlades in under experimentet.

Kapitel 5 presenterar de slutsatser som framkommit samt möjliga förändringar och förbättringar inför experimentet 2012.

2 Intrångsdetektering

Detta kapitel är baserat uteslutande på tidigare forskning och syftar till att ge en översikt över området. Kapitlet beskriver en uppsättning utvärderingskriterier (eller kvalitetsegenskaper) som en intrångsdetekteringslösning kan utvärderas utifrån. Utöver detta beskrivs även ett antal faktorer som tros ha betydande påverkan på utvärderingskriterierna men som sällan betraktas som sådana kriterier. En översikt över olika utvärderingskriterier och faktorer samt deras påverkan på varandra visas i Figur 1. Inom detta projekt har en delmängd av dessa studerats.



Figur 1: En översikt över utvärderingskriterier (ellipser) och faktorer (rektanglar) samt deras påverkan på varandra.

2.1 Utvärderingskriterier

Intrångsdetekteringssystem övervakar händelser som sker i datorsystem och analyserar dessa i syfte att upptäcka möjliga incidenter som är brott mot säkerhetspolicys, stundande brott mot säkerhetspolicys, brott mot policys för acceptabelt användande eller brott mot etablerade säkerhetsrutiner [1]. Det finns ett flertal sammanställningar över utvärderingskriterier som ett intrångsdetekteringssystem kan utvärderas mot. I [2] beskrivs olika typer av prestandamått och hur dessa bör testas. I [3] utvärderas intrångsdetekteringssystems förmåga utifrån fyra kriterier och i [4] beskrivs tre viktiga testmål. I [5] beskrivs ett stort antal utvärderingselement (eng. ”Evaluation Elements”). En delmängd av dessa passar in i denna jämförelse

(övriga ”element” är av beskrivande karaktär snarare än utvärderande). Tester har även gjorts på specifika områden, exempelvis utvärderas gränssnitt mot operatören i [6]. I Tabell 1 listas de benämningar på utvärderingskriterier som används i denna rapport tillsammans med de begrepp i annan litteratur som dessa försöker omfatta.

Utvärderingskriterium	**Benämningar som omfattas**
Detektionsförmåga	Security [7]; Accuracy [8] [6]; Intrusion Detection Capability [9]; Bayesian detection rate [10]
Antal falsklarm per dag	Probability of false alarms [2]; Probability of false positive [11]; False alarm rate [10]; False alarms per day [3]
Sannolikhet att attack upptäcks	Probability of detection [2] [11]; Detection rate [3][10]; How well they detect intrusions [8]
Täckning	Coverage [2]; Broad detection range [4]; Ability to detect never before seen attacks [2]; Completeness and coverage of: Known attacks, Masquerade attacks, Denial of service, Malicious use, Leakage, Attempted break-ins, Penetration of security control system [5]
Motståndkraft mot attacker	Attack resistance [5]; Resistance to attacks directed at the IDS [2]
Analyskapacitet	Ability to handle high bandwidth traffic [2]; Scalability (background traffic) [3]; Performance [7]; Resilience to stress [4]; Economy in resource usage [4]; Operational impact [8]; Capacity verification for NIDS [2]; Overhead [5]; Detection time [5], Detection delay [3]
Användbarhet	Ability to identify an attack [2]; Diagnose an attack [8]; Ability to determine attack success [2]; Countermeasure activities [5]; Other measurements [2]; Confidence [6]; Accuracy (operator) [6]; Time on task [6]; Rating of interfaces [6];Total number of commands used [6]; Complexity[3]; Type/amount/origin of data [5];Other measurements [2]; Mining actionable alarms [12]; Ability to correlate events [2];

Tabell 1: Utvärderingskriterier för intrångsdetekteringssystem.

Utvärderingskriterierna i Tabell 1 beskrivs i avsnitt 2.1.1–2.1.7 nedan.

2.1.1 Detektionsförmåga

Det som i Tabell 1 benämns som detektionsförmåga är ett övergripande utvärderingskriterium som flera författare berör och förklarar som kombinationer av ett flertal faktorer. Få definierar dock detta utvärderingskriterium formellt. Ett undantag är [9] som med ett informationsteoretiskt angreppssätt försöker fånga flera utvärderingskriterier i en metrik. På ett liknande sätt är Axelssons Bayesianska detektionskvot [10] en sammanfattande metrik för intrångsdetekteringssystemets förmåga. Denna anger sannolikheten att ett

genererat alarm faktiskt beror på en attack, dvs. *P(Händelse är attack | Alarm avges)*[1]. Det finns även ansatser som angriper problemet genom att uttrycka nytta i monetära termer. Detta görs utifrån flera utvärderingskriterier och indata som kostnad för incidenthantering så väl som kostnad för förlorad data [11][12].

2.1.2 Sannolikhet att falsklarm avges

Om ett intrångsdetekteringssystem genererar en stor andel falsklarm är detta naturligtvis problematiskt. Sannolikheten att ett falsklarm avges, *P(Alarm avges |Händelse är ej attack)*, är ett utvärderingskriterium som fångar intrångsdetekteringssystemets tendens att generera falsklarm.

Mätning av denna sannolikhet kräver dock att det finns en definition av begreppet ”händelse” som beskriver den typ av enhet som intrångsdetekteringssystemet analyserar och bedömer. Vad en händelse ska vara, eller rättare sagt, vilken analysenhet som är lämplig att använda för att bestämma denna sannolikhet är inte klart från litteraturen. Vissa använder nätpaket som analysenhet, andra använder nät/användarsessioner, vissa använder logghändelser, och andra använder abstraktare analysenheter som exempelvis användaraktivitet. Ett annat problem är att det i driftsatta miljöer sker en stor mängd olika händelser under en dag, med olika frekvens och intensitet. För många intrångsdetekteringssystem är också ordningen händelser sker i av vikt. Sannolikheten *P(Alarm avges | Händelse är ej attack)* är med andra ord beroende av karakteristiken för bakgrundstrafiken. En sannolikhet uträknad för en viss bakrundstrafik är förmodligen inte representativ för en annan typ av bakgrundstrafik.

Samtidigt som det kan vara problematiskt att bestämma sannolikheten att falsklarm avges är det förhållandevis enkelt att räkna antalet falsklarm som genereras av ett intrångsdetekteringssystem i en viss driftsatt miljö. För en beslutsfattare som överväger nyttan och kostnaden av ett intrångsdetekteringssystem som redan är installerat i en viss miljö (snarare än intrångsdetekteringssystemets generella kvalitet) är antalet falsklarm per dag tillräcklig information.

2.1.3 Sannolikhet att attack upptäcks

Intrångsdetekteringssystem ska hjälpa till att detektera attacker som sker. Sannolikheten att ett alarm avges när en attack sker, *P(Alarm avges | Attackhändelser sker)*, är därmed en viktig storhet.

[1] Utläses sannolikheten för att en händelse är en attack givet att ett alarm avges

På ett liknande sätt som händelser som inte är attacker är problematiska att definiera är det svårt att definiera händelser som är attacker. Till exempel, om en attackerare ansluter till en tjänst, utnyttjar en sårbarhet i den för att skaffa ytterligare rättigheter på maskinen, och därefter använder rättigheterna. Vilken eller vilka delar i denna kedja är då att betrakta som attacker? I en situation där hotbilden är kontrollerad (som i ett experiment) kan sannolikheten erhållas genom att matcha händelser som definierats som attacker mot alarmlistan. Kvoten mellan antalet attack-händelser som genererat alarm och det totala antalet attack-händelser motsvarar sannolikheten att en attack upptäcks.

Till skillnad från sannolikheten för att falsklarm avges är inte sannolikheten att en attack upptäcks beroende av bakgrundstrafiken i någon större utsträckning. Sannolikheten att attack upptäcks är framför allt påverkad av hotbilden och den förmåga som hotagenten har (se 2.2.2). Detta gör att en sannolikhet för att attacker upptäckts som tagits fram i ett experiment med en viss typ av hotagent utan större problem kan generaliseras även till andra driftsatta miljöer med samma intrångsdetekteringssystem och samma intrimning, givet att dessa är utsatta för hotagenter med likvärdiga kunskaper och likartat tillvägagångssätt samt att övriga påverkande faktorer är rimligt likartade.

2.1.4 Täckning

Med detta utvärderingskriterium avses vilka typer av attacker som intrångsdetekteringssystemet kan detektera. Vanligtvis beskrivs täckning utifrån en taxonomi över attacker, eller delar av attacker. En vanlig uppdelning är också mellan tidigare kända och tidigare okända attacker. Var gränsen går för vad som är tidigare känt och okänt är dock inte helt tydligt. Till exempel, om en ny attack komponeras av flera tidigare kända delar och tekniker, är det då en tidigare okänd attack trots att delarnas signaturer är bekanta? De attacker ett intrångsdetekteringssystem faktiskt kan upptäcka framgår ofta av dess specifikation. Till exempel anger signaturerna i ett signaturbaserat system vilka attacker det kan upptäcka.

2.1.5 Motståndskraft mot attacker

Ett intrångsdetekteringssystem kan självt bli attackerat på flera olika sätt. Det kan till exempel utsättas för attacker som introducerar störande alarm, vilka förvillar administratören. Om en stor mängd alarm kan genereras för till synes ofarliga händelser kommer dessa att tolkas som falsklarm av administratören. Eventuellt stängs också denna typ av alarm av, vilket leder till att efterföljande (riktiga) attacker inte upptäcks.

Ett mer rättframt sätt är att använda verktyg för att överlasta intrångsdetekteringssystemet med trafik. Beroende på hur intrångsdetekteringssystemet har installerats (se 2.1.6) kommer detta antingen att

skapa en överbelastningsattack mot det övervakade systemet eller betyda att intrångsdetekteringssystemet inte hinner inspektera all trafik vilket i sin tur gör att vissa händelser undgår inspektion. En vidare översikt över metoder som kan användas för att undgå detektion finns i [13].

2.1.6 Analyskapacitet

Ett intrångsdetekteringssystems analyskapacitet är den mängd av händelser som det kan analysera per tidsenhet. Intrångsdetekteringssystems primära syfte är att upptäcka pågående attacker så att dessa kan stoppas eller skadan begränsas. Intrångsdetekteringssystem kan också användas som ett logg-verktyg för att analysera attacker och skeden i efterhand, men sådan analys är varken deras primära syfte eller fokus för denna rapport. Om ett intrångsdetekteringssystem ska kunna bidra till att attacker stoppas eller att skadan från attacker begränsas behöver det analysera indata och ge alarm inom den tidsrymd som attacken pågår. En kort svarstid är alltså att föredra. Ett stort antal händelser sker i datornnät och i datorer. För att ett intrångsdetekteringssystem ska kunna se attacker som pågår behöver det därför kunna analysera stora datamängder i realtid. Ett intrångsdetekteringssystems analyskapacitet kan ses som en funktion av prestanda på dess analysmotor och den mängd datorkraft den har tillgång till. Ett intrångsdetekteringssystem med god prestanda kan således hantera en viss mängd händelser med mindre datorkraft än ett intrångsdetekteringssystem med dålig prestanda.

2.1.7 Användbarhet

En viktig del för användbarheten för intrångsdetekteringssystem är det beslutsstöd som ges till systemadministratörer och andra som använder systemet. Exempel på sådant stöd är förmågan att kunna:

- identifiera vilken systemdel som attackeras
- identifiera varifrån attacken görs
- identifiera vilken attack som görs samt vilken sårbarhet som utnyttjas
- ange välkalibrerade osäkerhetsmått på informationen
- göra allvarlighetsbedömningar av attacken
- ge information om attacken lyckats eller misslyckats.

Användandet av ett intrångsdetekteringssystem är också förknippat med kostnader. Dessa inkluderar bland annat:

- avgifter för licenser till mjukvara och uppdateringar
- kostnader för hårdvara
- kostnader för administratörer som installerar, konfigurerar och övervakar systemets larmlista

- kostnader för ”utryckningar” på falsklarm.

Mer formella sammanställningar över kostnader (och nyttor) går att finna i [11] och [12].

2.2 Faktorer som påverkar utvärderingskriterierna

I detta avsnitt ges en överblick över vilka faktorer som i litteraturen anses påverka utvärderingskriterierna och därmed vara viktiga för att skapa ett intrångsdetekteringssystem av god kvalitet. Läsaren bör beakta att genomgången som ges här inte gör anspråk på att vara komplett i bemärkelsen att alla faktorer som litteraturen beskrivit som potentiellt viktiga finns med. Mängden litteratur inom området är för omfattande för att en komplett sammanställning ska kunna göras inom detta projekt [2]. Inte heller görs anspråk på att vara definitivt korrekt. Vad olika faktorer påverkar och hur stor denna påverkan är, är okänd. Orsaken är framför allt de svårigheter som kringgärdar formella tester av intrångsdetekteringssystem [2][14]. Det som redovisas i detta avsnitt är de faktorer som påverkar de utvärderingskriterier som beskrivits i 2.1. En grov uppdelning i följande 8 faktorer görs:

- skydd av detektionssystemet
- hotagent och verktyg
- sensorers placering och indata
- detektionsmetod
- bakgrundstrafik
- användargränssnitt
- intrimning/upplärning
- systemadministratörens förmåga
- attackfrekvens

Faktorerna beskrivs i avsnitt 2.2.1 till 2.2.9.

2.2.1 Skydd av detekteringssystemet

En rad skydd kan användas för att skydda mot hotagenter (se avsnitt 2.2.2) som attackerar intrångsdetekteringssystemets funktionalitet. Exempel på sådana attacker är överbelastning, attacker som ger hotagenten tillgång till intrångsdetekteringssystemet och attacker som gör att hotagenten kan radera loggade händelser och alarm. God analyskapacitet, d.v.s. antalet händelser som

[2] Som exempel kan nämnas att en sökning på ”intrusion detection system” i titel, sammanfattning eller nyckelord i den vetenskapliga referensdatabasen Scopus (www.scopus.com) renderar över 4000 artiklar publicerade efter 2004.

kan analyseras per tidsenhet (se avsnitt 2.1.6), är ett motmedel mot överbelastningsattacker. Även sedvanlig systemsäkerhet för intrångsdetekteringssystemets komponenter (sensorer, servrar, användar-terminaler och kommunikationslänkar) är av vikt för dess motståndskraft mot attacker [1].

2.2.2 Hotagent och verktyg

En hotagent genomför attacker och är den som avgör vilka metoder som ska användas, vilka tekniker som ska brukas och vilka verktyg attacker ska utföras med. Eftersom dessa val påverkar vilka händelser som sker påverkar de även sannolikheten att attacker upptäcks av intrångsdetekteringssystemet.

En hotagent kan undgå upptäckt i signaturbaserade system genom att utföra attacker som det saknas signaturer för. Signaturer saknas exempelvis om en hotagent använder sårbarheter som inte är allmänt kända. En hotagent kan även försöka smälta in bakgrundstrafiken för att undgå upptäckt. Exempelvis kan en hotagent försöka smälta in i bakgrundstrafiken genom att dela upp attacker i flera steg utspridda över tid för att enbart generera alarm som tycks vara orelaterade och slumparatade (som alarm från bakgrundstrafik ofta är).

2.2.3 Sensorers placering och indata

Ett intrångsdetekteringssystem kan använda olika typer av data för att upptäcka intrång och det kan samla in data via sensorer placerade på olika platser. Förutom detektionsförmågan påverkar detta vilken täckning systemet har, det vill säga vilka typer av händelser/attacker det kan detektera.

De två vanligaste typerna av data som samlas in är rå nättrafik respektive operativsystemhändelser. Utöver detta är det även vanligt att loggar från antivirusprogram, datorers mjukvaruapplikationer och loggar från nätutrustning såsom brandväggar inkluderas [15]. Vilka av dessa loggar som ett intrångsdetekteringssystem kan komma åt beror på var sensorerna är placerade. Placeringen av sensorer avgör även om intrångsdetekteringssystemet kan se attacker mot en viss mjukvara, maskin eller nät. En vanlig uppdelning i litteraturen är att skilja mellan värdbaserade (eng. ”host based”) och nätbaserade (eng. "network based”) sensorer [8][16]. Inom båda dessa ryms varianter [16]:

- En värdbaserad sensor kan vara en del av systemet den ska granska (t.ex. en del av operativsystemet) eller utanför detta (t.ex. en separat applikation i operativsystemet).
- En nätbaserad sensor kan sitta på samma fysiska nät som det ska övervaka (t.ex. vid en switch) eller på det stomnät som sammankopplar segmenten (t.ex. vid en router).

- En nätbaserad sensor kan vara i linje med trafiken (eng. ”in-line”) och kräva att all trafik går igenom den eller vara bredvid och bara lyssna på trafiken (t.ex. via en separat lyssningsport på någon del av nätutrustningen).

Ju närmare sensorerna är placerade datorerna som ska övervakas, desto mer information kan samlas in. Till exempel kan en värdbaserad sensor se både händelser som sker internt på en maskin och den nättrafik som en nätbaserad sensor kan se. Å andra sidan krävs fler sensorer för att ge en överblick över all nättrafik.

Den sista punkten handlar framför allt om att kompromissa mellan prestanda, driftkostnad och detektionsförmåga. En lösning där all trafik går igenom sensorn gör att sensorn ”ser” allt, vilket ökar sannolikheten att attacker upptäcks. Dock kan det bromsa upp trafik eller medföra borttappad trafik om sensorns prestanda är otillräcklig. En sensor som bara lyssnar på trafiken kan inte bli ett sådant problem. Om sensorns prestanda är för dålig kan den kan å andra sidan släppa förbi trafik som innehåller attackhändelser utan att hinna undersöka trafiken.

2.2.4 Detektionsmetod

Intrångsdetekteringssystem kan använda olika tekniker för att detektera attacker. Ett flertal förslag finns på hur dessa kan delas in i olika grupper [1][4][5][15][17–19]. En grov uppdelning är den som görs i [15]:

- **Programmerad signaturbaserad:** En uppsättning allmänna regler (signaturer) används och matchas mot händelser i systemet. Om en regel matchar en händelse genereras ett alarm.
- **Självlärd anomalibaserad:** Intrångsdetekteringssystemet ”lärs upp” genom att det först utsätts för normal trafik som är fri från elakartad aktivitet. Ett stort antal nätparametrar mäts kontinuerligt och om dessa avviker allt för mycket från normalvärdena genereras ett alarm.
- **Programmerad anomalibaserad:** Utifrån specifikationer och/eller manuella inmatningar beskrivs för intrångsdetekteringssystemet vad som är godtagbar/normal aktivitet. Om avvikelser från detta sker genereras alarm.

Kombinationer av dessa tre i en och samma lösning är fullt möjliga.

Det finns ett stort antal förslag på algoritmer och tekniker för intrångsdetekteringssystem, framför allt för de anomalibaserade intrångsdetekteringssystemen med motiveringen att dessa också kan upptäcka okända attacker [1][5][15][20].

Förmågan hos signaturbaserade lösningar begränsas av de signaturer som systemet är programmerat att reagera på. Om ingen del av en attack matchar någon av de programmerade signaturerna kommer den signaturbaserade lösningen inte att avge ett alarm. Trots denna svaghet och de stora

forskningsinsatser som riktats mot det anomalibaserade området dominerar signaturbaserade lösningar i dagsläget marknaden och tillämpningar i praktiken.

2.2.5 Bakgrundstrafik

Det kanske största problemet med dagens intrångsdetekteringssystem är det stora antalet falsklarm som genereras av händelser som är ofarliga och harmlösa. Då det typiskt sker ett stort antal normala/ofarliga händelser i datornät kan även en liten sannolikhet att falsklarm avges leda till ett stort antal falsklarm per dag. I riktiga installationer kan andelen falsklarm av den totala mängden alarm vara 99 procent [21].

Den trafik som normalt finns i ett nät kallas ofta för bakgrundstrafik. Det är när händelser ur bakgrundstrafiken genererar ett alarm som ett falsklarm uppstår. Hur mycket bakgrundstrafik som finns och hur problematisk denna är skiljer sig mellan olika miljöer. Ju mer bakgrundstrafik, desto fler falsklarm och ju högre sannolikhet att falsklarm genereras för en händelse, desto fler falsklarm. Som förklaras i [2] finns det inget ”standardnät” att utgå från när ett intrångsdetekteringssystem ska utvärderas, vilket gör generaliseringar utifrån tester svåra. Överlag kommer en miljö med mycket bakgrundstrafik samt stor variation i bakgrundstrafikens innehåll och frekvens att skapa större problem än en relativt tyst och statisk miljö.

2.2.6 Användargränssnitt

Det är önskvärt att meddelanden som intrångsdetekteringssystemet genererar vid alarm är informativa och hjälper administratören att agera korrekt. Det finns gott om idéer på hur alarm och händelser ska presenteras för en administratör (se t.ex. [23][24]) samt ett fåtal tester på hur effektiva olika metoder är (se t.ex. [6]).

2.2.7 Intrimning/upplärning av intrångsdetekteringssystemet

Oavsett om intrångsdetekteringssystemet är anomalibaserat eller signaturbaserat, programmerat eller självlärt, så behöver det installeras och trimmas in i sin miljö.

För självlärande, anomalibaserade lösningar görs intrimning/upplärning genom att utsätta sensorerna för ”normal”, godartad, trafikdata och applicera den självlärande metoden på dessa data. Som nämns i avsnitt 2.2.4 finns ett stort antal förslag på sådana lösningar, till exempel Markovbaserade algoritmer, multivariata metoder och Bayesianska nätverk [24]. För programmerade, anomalibaserade lösningar är upplärningen en fråga om att specificera det normala för detektionssystemet. Att specificera vad som är normalt kan kräva en betydande insats i miljöer som är dynamiska eller komplexa. Självupplärning av intrångsdetekteringssystem lider av problemet att man vid upplärningsfasen

behöver ett system utan elakartat beteende, annars kan det elakartade accepteras som ”normalt”. Man behöver dessutom gott om ”normal” trafik för att det inte ska larmas för nya typer av ”normal” trafik i skarp användning.

De signaturbaserade systemen är uteslutande programmerade. Signaturerna ska programmeras så att de beskriver de elakartade händelserna. Signaturbaserade lösningar innehåller typiskt en databas med grundläggande signaturer som kontinuerligt uppdateras. Vanligtvis behövs dock ett betydande jobb göras med att anpassa dessa signaturer till miljön som systemet ska fungera i. Det kan också skrivas signaturer som är speciellt anpassade för miljön, till exempel för egenutvecklade protokoll eller mjukvara. Anpassningarna av de grundläggande signaturerna görs framför allt för att reducera antalet falsklarm som systemet genererar; tilläggen görs framför allt för att öka andelen attacker som upptäcks. Hur dessa anpassningar och tillägg görs bäst är i stort ett outforskat område. Intervjuer med systemadministratörer [25–30] har gett en grov bild av vilken typ av kunskap som krävs för att trimma in eller lära upp ett system. Denna kunskap är allmän kunskap om datornät och informationsteknik, allmän kunskap om säkerhet och datorintrång och en ansenlig mängd situationsberoende kunskap om det övervakade systemet (dvs. kunskap om miljön där systemet befinner sig).

2.2.8 Systemadministratörens förmåga

Systemadministratören (eller administratören) som övervakar intrångsdetekteringssystemet är den som ska avgöra om ett alarm ska ignoreras, om det ska undersökas vidare, eller om det krävs direkt handling (och i så fall hur). En skicklig administratör filtrerar bort falsklarm och agerar på korrekta alarm. Precis som vid intrimning och upplärande av ett intrångsdetekteringssystem måste systemadministratören vid användandet av intrångsdetekteringssystemet besitta samma typer av kunskap för att filtrera de alarm som genereras som beskrivs i avsnitt 2.2.7:

Utöver detta behöver administratören ha en kognitiv förmåga för att bedöma alla alarm som kommer in för att kunna se alla attacker som sker. Om intrångsdetekteringssystemet faktiskt ska bidra till att attacker avvisas behöver administratören också vara utrustad med förmågan att avgöra vem som ska kontaktas eller vad som ska göras [1][31].

2.2.9 Attackfrekvens

Andelen elakartad trafik av den totala mängden trafik spelar roll för ett intrångsdetekteringssystems förmåga. Alltså är den frekvens med vilken attacker sker av vikt. Ett system som uteslutande utsätts för attacker kan med enkelhet konstrueras för att ge en felfri detektionsförmåga genom att markera all trafik som attacker. På motsvarande sätt kan ett intrångsdetekteringssystem som inte

alls utsätts för attacker med enkelhet konstrueras för att ge en felfri detektionsförmåga genom att markera all trafik som ofarlig.

I driftsatta miljöer är attacker blandade med en stor mängd bakgrundstrafik (som är ofarlig). Att trimma in intrångsdetekteringssystemet så att det genererar alarm vid händelser som är attacker men samtidig inte genererar en stor mängd falsklarm för händelser ur bakgrundstrafiken är svårt när händelser från bakgrundstrafik är tiopotenser vanligare än händelser som är attacker [32]. Ju fler attacker som sker per dag desto lättare är det att uppnå en god sannolikhet att attack upptäcks samtidigt som antalet falsklarm per dag hålls på en rimlig nivå. Antalet attacker som sker är med andra ord av betydelse.

3 Experimentuppställning

Denna del av rapporten beskriver experimentet som genomfördes inom projektet under hösten 2011. Beskrivningen görs utifrån de begrepp som introducerades i kapitel 2. Först presenteras syftet med experimentet, därefter presenteras hur de olika faktorerna behandlades under experimentet.

3.1 Experimentets syfte och omfattning

Tidigare forskning inom intrångsdetektering har i stor utsträckning varit av teoretisk karaktär och handlat om utveckling av ny ”state of the art”. Få empiriska studier har överhuvudtaget gjorts av de lösningar som är vanliga idag. De flesta empiriska tester som gjorts använder data från DARPAs tester från millennieskiftet [33] trots att flera välkända problem finns med dessa data [14]. Med anledning av bristen på empiriska data behövs indikativa, aktuella resultat inom intrångsdetekteringsområdet, baserade på andra dataset.

Syftet med det genomförda experimentet var att i en realistisk miljö undersöka vilket stöd ett intrångsdetekteringssystem kan ge säkerhetsarbetet i driftsatta miljöer. För att undersöka detta inriktades experimentet på att testa ”state of the practice” snarare än ”state of the art”. En utgångspunkt för experimentet var att systemadministratören har en viktig roll att spela när intrångsdetekteringssystem används i praktiken. Experimentet som presenteras i denna rapport testade följande två hypoteser:

Hypotes 1: *En kompetent administratör kommer tillsammans med sitt intrångsdetekteringssystem kommer att ha en högre detektionsförmåga än enbart intrångsdetekteringssystemet.*

Hypotes 2: *Även en kompetent administratör kommer tillsammans med sitt intrångsdetekteringssystem att upptäcka färre attacker än enbart intrångsdetekteringssystemet.*

Experimentet använder två utvärderingskriterier för att testa dessa hypoteser: *Detektionsförmåga* (i form av Bayesiansk detektionskvot) och *Sannolikhet att attack upptäcks*.

Den första hypotesen är relaterad till det välkända problemet med falsklarm. Ett huvudresultat av de kvalitativa studier som gjorts i [25–30] är att systemadministratören spelar en viktig roll i hanteringen av dessa falsklarm. Detta gör systemadministratörer både genom att trimma in systemet så att falsklarm undviks och genom att filtrera larmlistan genom okulär inspektion. Den första hypotesen syftar till att testa detta.

Den andra hypotesen syftar till att testa om detektionsförmågan påverkas negativt av en administratörs filtrering. Då administratören filtrerar listan med alarmen

finns naturligtvis en risk att korrekta alarm ignoreras som falsklarm. Det är därför rimligt att tro att en administratör sänker antalet attacker som upptäcks jämfört med om alla alarm skulle leda till en åtgärd. Detta är dock inte säkert då administratören kan upptäcka attacker som intrångsdetekteringssystemet inte genererat alarm för. Till exempel, kan administratören göra detta genom att härleda tidigare lyckade attacker utifrån de alarm som har genererats, eller genom att manuellt övervaka maskiner.

Som beskrivs i kapitel 2.2 finns ett stort antal faktorer som påverkar detektionsförmågan och sannolikheten att attacker upptäcks. Hur dessa faktorer behandlades i experimentet är beskrivet i avsnitt 3.2–3.9. Resultatet av hypotestesterna beskrivs i avsnitt 4.1 och 4.2.

3.2 Detekteringsmetod

Intrångsdetekteringssystemet som användes hade både programmerade signaturbaserade och programmerade anomalibaserade detektionsmetoder, men med ett starkt fokus på signaturbaserad detektion. Lösningen utvecklades vid FOI och kallas Panorama och bestod av två intrångsdetekteringssystem (Snort och OSSEC) och ett driftövervakningsverktyg (Zabbix). För en mer ingående beskrivning av Panorama samt den miljö i vilket experimentet genomfördes se [34].

Snort analyserar datapaket i nättrafik och genererar alarm om dessa stämmer överrens med någon regel i dess databas. Databasen innehåller cirka 10 000 regler. Två exempel på regler är nummer 2951 som ska ge alarm om flera förfrågningar görs direkt efter varandra i ett Windowsnät, samt nummer 3018 som ska ge alarm om skadlig kod skickas till nättjänsten Samba. Under experimentet användes, med ett fåtal modifikationer, Snorts ursprungliga signaturer (se avsnitt 3.4).

På motsvarande sätt som Snort analyserar nättrafik analyserar OSSEC händelser (loggar) på maskiner och genererar alarm ifall de överensstämmer med någon av dess regler. Exempel på regler är nummer 40107 som ska larma om en ”heap overflow”-attack görs mot tjänsten cachefsd i Solaris-maskiner. Förutom dessa regler har OSSEC även programmerade anomalibaserade detektionsmetoder i form av att inspektera filers integritet, registrets integritet (i Windows-maskiner), och anomalidetektion av så kallade ”root-kit” i Unix-baserade maskiner. Inga ändringar gjordes av de inbyggda delarna i OSSEC, och inga nya regler skapades vid genomförande av experimentet.

Zabbix är ett driftövervakningsverktyg. Det kan konfigureras för att generera alarm vid situationer som när en maskin har för hög belastning på processorn, eller när en nätuppkoppling belastas över en fördefinierad nivå. Sådana regler användes i experimentet, främst för att övervaka tillgängligheten till maskiner.

Zabbix grafiska gränssnitt användes också för att presentera alarm från intrångsdetekteringssystemen Snort och OSSEC. Detta är beskrivet under användargränssnitt (avsnitt 3.9).

3.3 Sensorers placering

Två typer av sensorer användes i experimentet. Snort använde nätbaserade sensorer och OSSEC använde värdbaserade sensorer. Driftövervaknings-verktyget Zabbix använder också värdbaserade sensorer.

Snorts nätsensorer placerades vid portar i experimentnätets switchar så att det mottog kopior på alla paket som skickades i de övervakade näten (se Figur 2 för en beskrivning av de övervakade näten). För att förhindra prestandaproblem användes totalt tre sensorer på de fem övervakade näten. OSSEC och Zabbix installerades på samtliga maskiner i de övervakade näten. I och med att även maskinerna som körde brandväggarna hade Zabbix installerat kunde en bild över aktuella trafikbelastningar erhållas. Både Snorts, OSSECs och Zabbix sensorer skickade sina alarm och observationer vidare till centrala system, som presenterade dessa i ett webbaserat användargränssnitt.

3.4 Intrimning

Som beskrevs i avsnitt 3.2 gjordes endast smärre ändringar på intrångsdetekteringssystemens grundutförande. OSSEC (i experimentet användes OSSEC version 2.5.1) behölls i sitt grundutförande och alla typer av alarm användes. De ändringar som gjordes var i Snort (i experimentet användes Snort version 2.9.0.5) och kan summeras som:

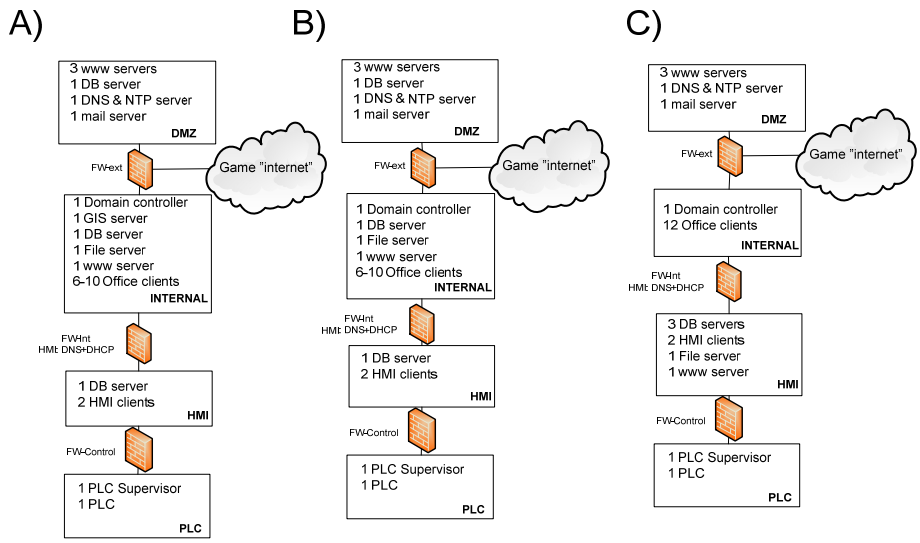
- Alarm på nivå 0 (”priortering saknas”) och nivå 4 (”udda trafik”) i Snort ignorerades med motiveringen att reglerna som genererar dessa inte är direkt relaterade till säkerhet eller intrång.
- Ett antal regler som producerade störande brus togs bort. Dessa var alarm nummer 4 om felaktiga tidsstämplar i paket, alarm nummer 12 om små IP-paket, alarm 399 om maskiner som försöker ansluta till servrar som inte är åtkomliga och nummer 3218 om Windows maskiner som försöker ansluta till en server.

Utöver dessa ändringar filterades vilka Snort-alarm som skulle presenteras för administratören, dvs. när något skulle skickas vidare till det grafiska gränssnittet. En separat modul bestämde när alarm skulle vidarebefordras till det centrala systemet. Om minst 50 alarm av prioritet 3 (låg) hade genererats i Snort under de senaste 5 minuterna betonades detta med ett särskilt alarm i användargränssnittet; om minst 10 alarm av prioritet 2 (medium) hade genererats av Snort de senaste 10 minuterna betonades detta med ett särskilt alarm i användargränssnittet; om

ett alarm av prioritet 1 (hög) hade genererats under de senaste 20 minuterna betonades detta med ett särskilt alarm i användargränssnittet. Dessa tröskelvärden valdes med omsorg av administratören för att begränsa antalet falsklarm och för att förhindra en ohanterlig mängd alarm vid aktivitet som är högljudd[3] och genererar många likartade alarm i Snort (som t.ex. då vissa kartläggningsverktyg används).

3.5 Bakgrundstrafik

I experimentet övervakades fem datornät konstruerade som i Figur 2. Dessa nät innehöll maskiner med olika mjukvaruversioner och olika skydd. Det fanns även användare med olika namn och kontouppgifter i de fem datornäten. De fem näten sammankopplades via ett fiktivt internet, med flera internetleverantörer och ett stort antal fiktiva webbsidor.



Figur 2: Nätarkitekturer i målnäten. Ett målnät hade en arkitektur som i A); ett målnät hade en arkitektur som i C); resterande målnät hade en arkitektur som i B).

För att skapa realism installerades användaragenter på klientdatorer i målnäten. Agenterna implementerades med automatiseringsverktyget AutoIt[4] och kördes på Windows-maskiner som var konfigurerade att automatiskt logga in en kontorsanvändare och direkt vid inloggning exekvera ett program som

[3] Det vill säga trafik som genererar många händelser av suspekt karaktär.
[4] Se http://www.autoitscript.com/

genererade knapptryckningar via maskinens användargränssnitt. Programmet valde med 1-2 minuters mellanrum bland följande saker att göra:

- 35 % sannolikhet: Öppna Internet Explorer och surfa till en slumpvis vald hemsida i experimentnätet. Väl på denna hemsida följs länkar slumpvis ett antal gånger.
- 15 % sannolikhet: Öppna en nyligen ändrad fil i mappen ”Mina dokument”.
- 10 % sannolikhet: Använda Outlook för att skicka e-post till en slumpvis vald mottagare i experimentnätet. E postens innehåll består av ett slumpat utklipp från en längre text.
- 10 % sannolikhet: Öppna Outlook och läs det senaste e-postmeddelandet som inkommit. Eventuella bilagor sparas i mappen ”Mina dokument”. Då användaragenterna inte skickar bilagor till varandra är denna funktion till för att ge angriparna en möjlighet att ta över system genom att skicka skadlig kod i bilagor.
- 10 % sannolikhet: Kopiera filer mellan ”Mina dokument” och en delad mapp i nätet.
- 20 % sannolikhet: Gör ingenting.

Dessa sannolikheter bedömdes som rimliga för en kontorsmiljö men det har inte gjorts några empiriska studier där typisk användaraktivitet kartlagts. Inte heller varierar aktivitetsprofilen under dagen eller mellan användare, vilket vore rimligt att förvänta sig för bakgrundstrafik i vanliga organisationer.

3.6 Hotagent och verktyg

Hotagenten i detta experiment utgjordes av Försvarsmaktens IT-förband. Mindre än en handfull individer var involverade i angreppen som skedde under experimentet[5]. Som primärt verktyg användes Backtrack 5, ett operativsystem (med tillhörande mjukvara) skapat för penetrationstester och liknande. Utöver denna plattform användes en uppsättning egenproducerade skript, exempelvis för att kartlägga tjänster, men inga andra verktyg än de som kommer med Backtrack 5 användes. Det angivna målet för hotagenten att bevisa att den tekniska infrastrukturen var otillförlitlig. Det var alltså ett högnivåmål som kunde uppnås på ett stort antal sätt.

Eftersom de exakta aktiviteterna var ett utfall i experimentet snarare än en kontrollerad faktor lades energi på att dokumentera dessa aktiviteter. De viktigaste och mest lättillgängliga data som samlades in var de loggar som deltagarna i experimentet förde. Dessa fördes utifrån förutbestämda mallar för att ge en tydlig bild av vad angriparna gjorde vid olika tidpunkter. Loggen var i

[5] Detaljer kring denna grupps sammansättning och exakt antal individer som agerade inom experimentet är okänt.

förhand uppdelad på huvudgrupperna kartläggning och attack. Ett utklipp från loggen av kartläggningar visas i Figur 3.

	Provide stating time (hh:mm and ending time if applicable (hh:mm)		Provide the machine from which the scan is performed	Provide the machine/port/IP-address or IP range which is scanned	Add comments here, for instance, if something goes wrong, flags used etc.
Day (yymmdd)	**Time-start**	**Time-end**	**Performed from (machnine name or IP-address)**	**Performed against machine/port/IP/IP-range/network**	**Notes**
111011	08:05		.233	172.16.0.0/12	045. nmap -sn
111011	08:07		.165	172.18.1.5	046. http://
111011	08:13		.235	172.17.1.5,172.19.1.5	047. http://
111011	08:51		.150	10.100.0.0	048. Traceroute
111011	08:30	08:50	.235	172.21.1.0/24	049. nmap -A
111011	10:02		.235	172.21.1.4	050. Nessus webapp-scan
111011	09:25	10:03	.235	172.18.1.0/24	051. nmap -A
111011	10:20		.141	172.21.1.2,5,6	052. FTP
111011	10:37		.150	10.110.64.0/24	053. nmap -sn
111011	10:43		.150	10.110.64.10	054. nmap -A

Figur 3: Utklipp från hotagentens logg.

Utöver dessa översiktliga beskrivningar sparade hotagenten resultaten av sina aktiviteter i separata filer. I den sista kolumnen i Figur 3 anger siffrorna före kartläggningsaktiviteten namnet på den fil där resultatet från aktiviteten sparades.

3.7 Skydd av detekteringssystemet

Intrångsdetekteringssystemet installerades på relativt säkra maskiner i de övervakade näten. Det vidtogs inga särskilda åtgärder för att separera intrångsdetekteringssystemet från de övervakade systemen. De använde samma datornät som övrig trafik och alarmen aggregerades till ett centralt system placerat i det fiktiva internet som ingick i experimentet. Det var tillåtet för hotagenten att försöka manipulera, slå ut eller på annat sätt attackera intrångsdetekteringssystemet för att undvika detektion. Sådana aktioner skulle dock ske i slutet av experimentet och skulle dokumenteras lika noggrant som andra offensiva handlingar.

3.8 Systemadministratörens förmåga och kunskap

Systemadministratören i detta experiment var samma person som trimmade in och designade lösningen. Att det är samma person som både trimmar in och övervakar intrångsdetekteringssystemet är vanligt i driftsatta tillämpningar [28] [29]. Det är också vanligt att det, som i detta experiment, är en person med god kännedom om den övervakade miljön som använder och administrerar systemet. Systemadministratörens expertis är relevant eftersom den påverkar både hur

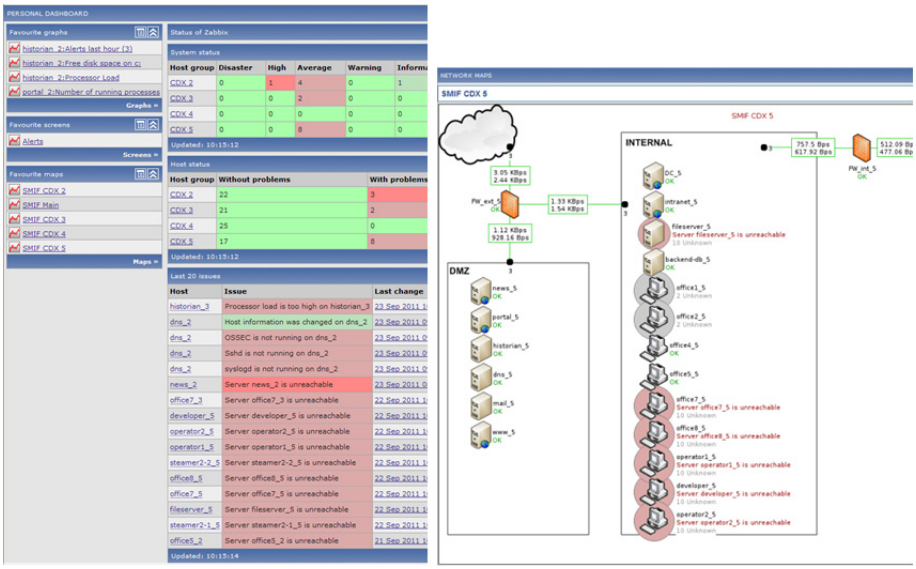
intrimning av lösningen görs och hur alarmen filtreras. I detta fall var det en erfaren forskare inom IT-säkerhet som i förhållande till det som anges som viktigt i litteraturen [25–30] hade:

- mycket god allmän kunskap om datornät och informationsteknologi
- mycket god allmän kunskap om säkerhet, datorintrång och de avtryck som olika verktyg lämnar
- mycket god kunskap om det övervakade systemet (dvs. kunskap om miljön där systemet befinner sig) och om den bakgrundstrafik som fanns i systemet.

En mer ovanlig form av kunskap som denna systemadministratör hade var den om hotbilden. Systemadministratören visste med säkerhet att attacker skulle ske under en viss tidsperiod, något som sällan är känt i driftsatta miljöer. Systemadministratören hade dessutom kännedom om vem hotagenten var och vilka verktyg de kunde tänkas använda.

3.9 Användargränssnitt

En viktig del av detta experiment var användargränssnittet. Eftersom flera typer av sensorer användes och ett relativt stort antal system övervakades behövdes ett samlat användargränssnitt för alarm från alla sensorer. En av idéerna var att möjligheten att få en översikt över alla alarm i ett enda användagränssnitt skulle göra det möjligt för administratören att korrelera olika händelser och därmed erhålla en förbättrad förståelse för läget. Gränssnittet implementerades i Zabbix html-baserade användargränssnitt. Figur 4 visar två översiktsvyer från detta gränssnitt.



Figur 4: Översiktsvyer från användagränssnittet i experimentet. Till vänster visas ett av de attackerade näten i tabellform och till höger visas det i form av en nätskiss.

Mer detaljerad information kunde fås på flera sätt genom att navigera till enskilda maskiner eller alarm utifrån dessa översiktsvyer. Exempelvis kunde användargränssnittet producera vyer där endast alarmen som genererats för en speciell maskin visades, och vyer som endast visade alarmen som genererats av en viss sensor (t.ex. Snort) för en viss dator.

4 Analys och resultat

I detta kapitel presenteras analysen och resultaten av experimentet. Analysen har gjorts med hjälp av de loggar som hotagenten, systemadministratören och intrångsdetekteringssystemet producerat och med hjälp av intervjuer med systemadministratören. Resultatet är i huvudsak en jämförelse mellan den detektionsförmåga som systemet har på egen hand och den detektionsförmåga administratören har tillsammans med systemet. Sist beskrivs den information och analysmetod som administratören använt för att filtrera larmlistan.

4.1 Sannolikhet att attack upptäcks

Totalt gjordes under experimentveckan 63 handlingar som av hotagenten betraktades som attacker. Majoriteten av dessa (49 stycken) var olika former av kartläggning, t.ex. nätavsökning. Bland attackerna fanns 14 försök att ta över maskiner genom att utnyttja mjukvarusårbarheter, gissning av lösenord online och extrahering av lösenordsfiler för off-line-knäckning. I Tabell 2 redovisas att av de totalt 63 handlingarna upptäckte intrångsdetekteringssystemet 44 (19 missades) och systemadministratören upptäckte 37 (26 missades). I analysen betraktades en attack som upptäckt om åtminstone ett alarm hade sitt ursprung i en handling som hotagenten klassat som en attack. Att upptäcka en attack i denna definition behöver med andra ord inte betyda att detaljerad och exakt information finns tillhanda om hotagentens handlingar. Inte heller behöver alla moment i en attack upptäckas för att den ska betraktas som upptäckt. T.ex., om ett helt nät scannas räcker det att det upptäcks att en maskin i detta nät scannas för att attacken ska betraktas som upptäckt.

	Systemet	Systemadministratören
Upptäckta attacker	44	37
Missade attacker	19	26
Sannolikhet att attack upptäcks	69 %	58 %

Tabell 2: Attacker utförda och upptäckta av system respektive administratör.

Resultatet tyder på att det är en större sannolikhet att attacker upptäcks om man litar blint på systemet än om man låter en administratör tolka resultatet först. Sett till alla attacker är sannolikheten att systemet detekterar en attack 69 procent medan kombinationen system och administratör ger 58 procent. Detta utgör dock

inte någon statistiskt signifikant skillnad[6]. Med andra ord, utifrån detta experiment går det inte att statistiskt säkerställa att det är en skillnad mellan systemet och systemet med administratör med avseende på sannolikheten att attacker upptäcks. Samtidigt bör det poängteras att detta experiment gjordes under förhållanden där systemadministratören kände till att attacker skulle utföras, vilket sällan är fallet i driftsatta miljöer. Detta scenario bör rimligen ha gjort administratören mer alert och lett till att alarm oftare bedömdes som en attack än i scenarion då det är osäkert ifall attacker är att vänta. Det är alltså rimligt att andelen attacker som en administratör vanligtvis ser i verkliga driftsatta miljöer är lägre än andelen som uppmättes i detta experiment.

4.2 Systemets och systemadministratörens detektionsförmåga

Att upptäcka attacker är en viktig förmåga för intrångsdetekteringssystem. Det är också viktigt att det inte genereras för mycket falsklarm. Studier av övervakning inom andra områden pekar på att om mindre än hälften av alarmen är korrekta så förlorar systemadministratören förtroendet för detektionsmetoden och börjar ignorera alarmen som den genererar [32]. Den Bayesianska detektionskvoten fångar detta. Den kan utryckas som sannolikheten att det är en attack givet att ett alarm avges: *P(Attack sker | Ett alarm avges)*. Som Tabell 3 visar avger systemet en stor andel falsklarm. Den Bayesianska detektionskvoten är 11 procent för intrångsdetekteringssystemet och 57 procent administratören tillsammans med systemet. Med andra ord: när systemet larmar är det en attack i vart tionde fall. När administratören larmar är det en attack i mer än vartannat fall. I detta fall är skillnaden statistisk signifikant[7].

	Intrångsdetekteringssystemet	Systemadministratören tillsammans med intrångsdetekteringssystemet
Alarm	2107	70
Alarm kopplade till en attack	233	40
Bayesiansk detektionskvot	11%	57%

Tabell 3: Alarm och falsklarm från intrångsdetekteringssystem respektive systemadministratören tillsammans med intrångsdetekteringssystem.

[6] Fishers exakta metod [40] testar om det kan uteslutas att skillnaden mellan två binära fördelningar beror på slumpen. I detta fall är fördelningarna för missade attacker samt upptäckta attacker som analyserats. Testet visar att det är 27 % sannolikhet att skillnaden beror på slumpen.

[7] Fishers exakta test visar på en signifikant skillnad mellan fördelningarna. Sannolikheten att skillnaden beror på slumpen är mindre än en promille.

I detta experiment har endast loggar från den tid experimentet pågått används, dvs. under kontorstid de fyra dagar då attacker utfördes. Som påpekats i avsnitt 4.1 ovan har detta sannolikt påverkat systemadministratörens detektionskvot positivt. Av samma skäl har det förmodligen också ökat antalet falsklarm som administratören genererat. System har inte denna situationskunskap och påverkas inte av att detta är en vecka då attacker är att vänta. Antalet falsklarm som detta system genererar är en funktion av bakgrundstrafiken. Inga djupare studier har gjorts för att säkerställa att bakgrundstrafiken är representativt för ett särskilt scenario, men antalet falsklarm som genereras under en arbetsdag (cirka 500) för dessa relativt små system, pekar på det orimliga i att agera på alla alarm som intrångsdetekteringssystemet avger. Detta gäller speciellt i miljöer som är stora och komplexa.

4.3 Information och analysmetod som systemadministratören använder

Som framgår av tabellerna i avsnitt 4.1 och 4.2 ovan var cirka nio av tio alarm som intrångsdetekteringssystemet gav inte kopplade till någon attack. Systemadministratören som granskade och analyserade dessa alarm tycks dock ha en god förmåga att filtrera bort falsklarm och mer än femdubblade den Bayesianska detektionskvoten. En intervju gjordes med systemadministratören för att identifiera hur denna resonerade och vilken typ av information som användes i analysen. Tidigare forskning på området har identifierat att expertis inom följande områden bidrar till en effektiv analys [25–30]:

- allmän kunskap om datornät och informationsteknologi,
- allmän kunskap om säkerhet och datorintrång
- situationsberoende kunskap om det övervakade systemet (dvs. kunskap om miljön där systemet befinner sig).

Intervjun utgick från tidpunkter i experimentet och bekräftade i stort att dessa former av expertis användes. Tidpunkter valdes för att representera tillfällen då administratören fattat korrekta beslut och inkorrekta beslut med olika gott stöd från intrångsdetekteringssystemet. Med korrekt beslut menas att administratören avfärdat falsklarm som just falsklarm eller korrekt markerat händelser som en attack. Ett gott stöd från intrångsdetekteringssystemet fås genom att det genereras korrekta alarm som relateras till attacken eller inte avger alarm på grund av bakgrundstrafik; ett dåligt stöd från intrångsdetekteringssystemet fås när det inte genererar tydliga alarm under en attack eller genererar alarm på grund av händelser i bakgrundstrafiken. De olika scenariotyperna som undersöktes sammanfattas i Tabell 4.

	Intrångsdetekteringssystemets reaktion	Systemadministratörens reaktion
A	Genererar ett stort antal korrekta alarm	Identifierar korrekt att det är en attack
B	Genererar ett stort antal korrekta alarm	Identifierar felaktigt att det är falsklarm
C	Genererar ett stort antal falska alarm	Identifierar korrekt att det är falsklarm
D	Genererar ett litet antal korrekta alarm	Identifierar korrekt att det är en attack
E	Genererar ett litet antal falska alarm	Identifierar felaktigt att det är en attack

Tabell 4: Scenariotyper som diskuterades under intervjun.

Scenariotyp A (med många korrekta alarm) kan tyckas förhållandevis oproblematisk för en administratör. Vid de tidpunkter som diskuterades var det stora antalet alarm som intrångsdetekteringssystemets genererade en viktig del av beslutet att anteckna det som en attack. Att alarmen dessutom passade väl in mot typiska attack-aktiviteter (t.ex. sårbarhetsavsökningar) och att externa nätadresser var involverade bidrog också till beslutet.

Intressantare är kanske skillnaden mot scenariotyp B och C, då administratören inkorrekt respektive korrekt betraktar den stora mängden alarm som falska. Tre tillfällen av scenariotyp B analyserades. Bakgrunden till att administratören felaktigt avfärdade alarmen var i två fall att alarmen helt enkelt missades på grund av arbetsbelastningen, och i ett fall att administratören betraktade systemet som attacken utfördes mot som redan förlorat (dvs. att hotagenten redan hade komprometterat systemet).

Fyra fall av scenariotyp C diskuterades. I dessa avfärdas alarmen korrekt som falsklarm och skälen till detta var något eller flera av följande:

- det är alarm som bottnade i typiskt beteende hos Windowsmaskiner
- det var en attack som hotagenten rimligen inte hade förmåga att genomföra som skapar sådana alarm
- det var en attack som hotagenten orimligen kunde utföra eftersom den inte kommit så djupt in i datornätet (ännu).

En del av systemadministratörens analys utgick med andra ord från kunskap om hotagentens allmänna förmåga och en annan del utgick från vilka privilegier de hittills kan ha tillskansat sig. Det förstnämnda (andra punkten i listan) är en typ av expertis som inte passar in bland de typer av expertis som i [25–30] identifierats som användbara. Möjligheten att bruka denna expertis är till viss del en effekt av experimentets uppställning, där administratören hade viss kännedom om vem hotagenten var. Men det är också en typ av expertis som väl underrättade administratörer bör kunna bruka i driftsatta miljöer. Kunskapen om hotagentens nuvarande privilegier (tredje punkten i listan) kan ses som en del av den situationsbaserade expertis som en beskrivs i [25–30], även om kunskap om hotagentens nuvarande privilegier är mer temporär.

Ett antal scenarion av typ D och E diskuterades också. I de fall administratören korrekt identifierade att det var en attack utifrån ett fåtal antal alarm var orsakerna: att alarmen hade hög prioritet (vilket är ovanligt), de skapades av en nätadress som är extern, det var ovanlig intern trafik, och/eller det var ovanligt mycket belastning på berörda systemdelar. Två scenarion av typ E identifierades. I dessa bedömde administratören felaktigt att en attack hade skett baserat på ett litet antal (falska) alarm. I ena fallet var orsaken att det var ett högprioritetslarm och i det andra var orsaken att det upplevdes som rimligt att hotagenten skulle försöka med en sådan attack givet dåvarande privilegier och deras förmodade mål.

Sammanfattningsvis bekräftas alltså den teori som ges i litteraturen. En administratör använder såväl allmän kunskap om datorsystem (som då typiskt Windows-beteende identifierades i scenariotyp C), allmän kunskap om datorsäkerhet (som då alarm identifierades som spår av sårbarhetsavsökningar i scenariotyp A) och situationsbaserad kunskap (som då ovanliga uppkopplingar mellan interna maskiner identifierades i scenariotyp D). Utöver dessa typer av expertis använde administratören också expertis om vad en hotagent av denna typ rimligen kan utföra för attack och situationsberoende kunskap om hotagentens potentiella mål och nuvarande privilegier.

5 Diskussion och slutsatser

Detta kapitel sammanfattar de slutsatser som kan dras angående intrångsdetekteringssystems potential och diskuterar förändringar som kan göras i experimentuppställningen inför kommande experiment.

5.1 Intrångsdetekteringssystems potential

Mjukvarorna som kombinerades i experimentet är vanligt förekommande i driftmiljöer som använder intrångsdetekteringssystem och den typ av intrimning som gjorts är lik den som typiskt görs på intrångsdetekteringssystem. Likaså är attackerna som utförs troligen representativa för de typer av attacker som skulle utföras av en mindre resursstark hotagent. Å andra sidan är detta experiment begränsat till ett visst intrångsdetekteringssystem, konfigurerad på ett visst sätt, och testad i en viss miljö med en bakgrundstrafik som eventuellt inte är representativ. Av förklarliga skäl går det inte att göra vida generaliseringar utifrån detta experiment angående intrångsdetekteringssystems allmänna potential. Experimentet bör snarare ses som en antydan till vad potentialen i ”state of the practice” är idag.

Testet pekar på tydliga problem med falsklarm, och detta trots en förhållandevis homogen uppsättning maskiner och förhållandevis lite dynamik i denna uppsättning. Å andra sidan tycks det som att alarmen utgör en god grund för bedömningar gjorda av systemadministratörer. Den administratör som övervakade näten med hjälp av ett intrångsdetekteringssystem i detta experiment lyckades nästan upptäckta lika många attacker som intrångsdetekteringssystemet (58 kontra 69 procent) och avge en betydligt högre andel korrekta alarm (57 kontra 11 procent). Administratören i detta experiment resonerade som administratörer gör i allmänhet och använde flera olika typer av expertis för att filtrera larmen. Administratören använde bland annat expertis om det övervakade nätet (t.ex. vilka datorer som är tillförlitliga), expertis om datornät (t.ex. hur Windows-maskiner typiskt beter sig), expertis om datorattacker och detektionssystem (t.ex. vilka spår en typisk nätavsökning lämnar och expertis om vad hotbilden är (t.ex. vilka medel hotagenterna har att tillgå). Resultatet pekar på att när sådan expertis vägs in i analysen kan antalet falsklarm minskas utan att dramatiskt minska sannolikheten att attacker upptäcks. Resultatet visar samtidigt på begränsningar i administratörens förmåga att filtrera alarm – när systemet gav alarm som administratören missade berodde detta i princip uteslutande på att administratören inte hann/orkade inspektera alla alarm. Förbättrade användagränssnitt är en potentiell lösning på sådana problem.

5.2 Möjliga förändringar och förbättringar

Under 2012 kommer ett ytterligare experiment att genomföras med det i denna rapport beskrivna experiment som grund. Inför det nya experimentet är flera förändringar tänkbara för att öka korrektheten och tillförlitligheten i resultatet och/eller för att studera andra utvärderingskriterier och faktorer.

Som nämnts i avsnitt 5.1 ovan var den intrångsdetekteringslösning som användes i experimentet inte ”state-of-art”, utan snarare ”state-of-practice”. Det finns alltså ett stort antal förändringar som kan göras för att för förbättra lösningen. Det finns också potential att förbättra realismen i den miljö som experimentet görs i. I princip finns förbättringspotential inom alla faktorer som nämns i kapitel 2.2, till exempel:

a) förändringar i hur alarm och status visas i administratörens grafiska gränssnitt
b) tydligare och mer ingående prioritering av alarm
c) inkludering av algoritmer för att bedöma konfidensnivån i intrångsdetekteringssystemets bedömningar
d) ytterligare intrimning av reglerna som intrångsdetekteringssystemet använder
e) ytterligare sensorer och datakällor i intrångsdetekteringssystemet, som anomalidetektion eller antivirusprogram
f) skydd av intrångsdetekteringssystemet genom att separera det från resterande nätet
g) fler administratörer, eventuellt med olika bakgrund
h) en komplexare miljö och bättre anpassad bakgrundstrafik
i) implementation av automatiska reaktioner på alarm (”intrusion prevention”-funktionalitet).

Av resursskäl kommer inte alla möjliga förändringar att genomföras eller testas. Projektet kommer att sikta in sig på att till experimentet 2012 genomföra förbättringar inom områdena a), d) och h). Hur dessa förbättringar är tänkta att införas beskrivs i avsnitt 5.2.1–5.2.3 nedan. Utöver dessa kommer också förbättringar att göras i datornäten med avseende på nätkapacitet för att erbjuda en stabilare testmiljö.

5.2.1 Förändringar av larm och status i användgränssnittet

Lösningen som användes i experimentet visade alarm för administratören under 20 minuter för att sedan arkiveras. Att alarm bara visades under 20 minuter ledde troligtvis till att flera korrekta alarm missades, till exempel då operatören varit borta på rast. Dessutom låg alarm som betraktades som falsklarm kvar och krävde onödig kognitiv kraft av administratören. Till nästa experiment kommer administratören att ges möjlighet att kvittera alarm (som då arkiveras) och alarm

kommer att synas under en längre tid än 20 minuter. Eventuellt kan också operatörens anteckningar/loggbok integreras med det grafiska gränssnittet för att ytterligare förbättra användbarheten. Dessa förändringar förväntas leda till en ökning i den Bayesianska detektionskvot som systemadministratören kan ge tillsammans med systemet.

5.2.2 Ytterligare intrimning av regler

Ett stort antal falsklarm genererades av intrångsdetekteringssystemet i experimentet. Intrimningen gjordes med hjälp av regler (”triggers” i Zabbix terminologi) som använde de alarm som genererades i sensorerna Snort och OSSEC som input. Dessa regler var förhållandevis grova och baserades enbart på antalet alarm som genererats den närmsta tiden. Om ett visst antal alarm hade inkommit från sensorerna genererades ett alarm av intrångsdetekteringssystemet som visades för administratören. Om enstaka alarm kom in ignorerades sensorernas alarm för intrångsdetekteringssystemet.

Inför kommande experiment kommer ett antal förfiningar att göras i filteringen som görs av intrångsdetekteringssystemet. Modeller kommer att skapas för olika sekvenser av sensor-alarm som genereras av olika former av attacker, på ett sätt som liknar de beskrivna i [35–39]. En sådan förändring innebär t.ex. att de sekvenser av alarm som vanligtvis genereras av en nätavsökning, en sårbarhetsavsökning och lösenordsgissning kommer att generera alarm i intrångsdetekteringssystemet. Om de alarm som kommit in från sensorer den närmsta tiden passar med en modell över en attack så avges ett alarm till administratören. Om de alarm som kommit in från sensorer den närmsta tiden inte matchar det mönster som attacker tros ha så presenteras inget alarm för administratören. Hypotesen är att även relativt grova modeller av detta slag kommer att minska antalet falsklarm utan att påverka andelen upptäckta attacker nämnvärt.

5.2.3 Komplexare miljö och representativ bakgrundstrafik

En viktig sak för intrångsdetekteringssystem är den bakgrundstrafik det utsätts för. Som illustreras i Figur 1 är detta en faktor som inverkar på sannolikhet att falsklarm avges, och falsklarm är ett av huvudproblemen med dagens intrångsdetekteringslösningar. Under experimentet genererades bakgrundstrafik av realistisk karaktär i den mening att de aktiviteter som utfördes är vanliga, att de utfördes på ett vanligt sätt (”riktiga” mjukvaror användes) och att de exekverades slumpmässigt i tid enligt en fördefinierad modell. Inför kommande experiment är det önskvärt om det kunde säkerställas att bakgrundstrafiken följer mönster som är vanliga i någon viss typ av miljöer. Det vore också önskvärt att ha en ännu mer komplex miljö att övervaka. Till exempel en miljö som

inkluderar fler användare, fler aktiva maskiner och en mer dynamisk extern miljö.

6 Referenser

[1] K. Scarfone and P. Mell, “Guide to intrusion detection and prevention systems,” SP 800-94,National Institute of Standards and Technology (NIST), Gaithersburg, MD, USA, 2007.

[2] P. Mell, V. Hu, R. Lippmann, J. Haines, and M. Zissman, “An overview of issues in testing intrusion detection systems,” NISTIR-7007, National Institute of Standards and Technology (NIST), Gaithersburg, MD, USA, 2003.

[3] A. Ashfaq, M. Robert, A. Mumtaz, M. Ali, A. Sajjad, and S. Khayam, “A comparative evaluation of anomaly detectors under portscan attacks,” in proceedings of RAID '08 Proceedings of the 11th international symposium on Recent Advances in Intrusion Detection, pp. 351–371, Cambridge, MA, USA, 2008.

[4] R. Barber and H. A. Chan, “Intrusion Detection Systems What is intrusion detection and why do we need it?,” Computer Fraud, vol. 2001, no. 6, pp. 9-12, Jun. 2001.

[5] E. Biermann, “A comparison of Intrusion Detection systems,” Computers & Security, vol. 20, no. 8, pp. 676-683, Dec. 2001.

[6] R. S. Thompson, E. M. Rantanen, W. Yurcik, and B. P. Bailey, “Command line or pretty lines?: comparing textual and visual interfaces for intrusion detection,” in Proceedings of the SIGCHI conference on Human factors in computing systems, 2007, p. 1205.

[7] C. Xenakis, C. Panos, and I. Stavrakakis, “A comparative evaluation of intrusion detection architectures for mobile ad hoc networks,” Computers & Security, vol. 30, no. 1, pp. 63-80, Nov. 2010.

[8] M. J. Ranum, “Experiences Benchmarking Intrusion Detection Systems,” NFR Security White Paper, NFR Security, Inc, 2001. Available at [2012-03-22]: http://bandwidthco.com/whitepapers/compforensics/ids/Benchmarking%20IDS.pdf .

[9] G. Gu, P. Fogla, D. Dagon, and W. Lee, “Measuring intrusion detection capability: An information-theoretic approach,” in proceedings of the 2006 ACM Symposium on Information, computer and communications security, Taipei, Taiwan, pp. 90-101, 2006.

[10] N. Stakhanova, S. Basu, and J. Wong, “A taxonomy of intrusion response systems,” International Journal of Information and Computer Security, vol. 1, no. 1/2, p. 169, 2007.

[11] H. Cavusoglu, B. Mishra, and S. Raghunathan, “The Value of Intrusion Detection Systems in Information Technology Security Architecture,” Information Systems Research, vol. 16, no. 1, pp. 28-46, Mar. 2005.

[12] C. Iheagwara, “The effect of intrusion detection management methods on the return on investment,” Computers and Security, vol. 23, pp. 213-228, 2004.

[13] The VeriSign IDefense Intelligence Team, “Intrusion Detection System (IDS) Evasion,” VeriSign, Virgina, USA, 2006. Available at [2012-03-22]: http://complianceandprivacy.com/WhitePapers/iDefense-IDS-Evasion/iDefense_IDSEvasion_20060510.pdf

[14] J. McHugh, “Testing Intrusion detection systems: a critique of the 1998 and 1999 DARPA intrusion detection system evaluations as performed by Lincoln Laboratory,” ACM Transactions on Information and System Security, vol. 3, no. 4, pp. 262-294, Nov. 2000.

[15] S. Axelsson, “Intrusion detection systems: A survey and taxonomy,” Technical report 99-15, Chalmers University of Technology, Dept. of Computer Engineering, Sweden, 2000.

[16] S. Shaikh, H. Chivers, P. Nobles, J. Clark, and H. Chen, “Characterising intrusion detection sensors,” Network Security, vol. 2008, no. 9, pp. 10-12, Sep. 2008.

[17] M. A. Faysel and S. S. Haque, “Towards Cyber Defense : Research in Intrusion Detection and Intrusion Prevention Systems,” Journal of Computer Science, vol. 10, no. 7, pp. 316-325, 2010.

[18] B. I. A. Barry and H. A. Chan, “Intrusion detection systems,” in Handbook of Information and Communication Security, vol. 2001, no. 6, P. Stavroulakis and M. Stamp, Eds. Springer, 2010, pp. 193-205.

[19] P. Kabiri and A. A. Ghorbani, “Research on intrusion detection and response: A survey,” International Journal of Network Security, vol. 1, no. 2, pp. 84–102, 2005.

[20] P. Garciateodoro, J. Diazverdejo, G. Maciafernandez, and E. Vazquez, “Anomaly-based network intrusion detection: Techniques, systems and challenges,” Computers & Security, vol. 28, no. 1-2, pp. 18-28, Feb. 2009.

[21] K. Julisch and M. Dacier, “Mining intrusion detection alarms for actionable knowledge,” in Proceedings of the eighth ACM SIGKDD international conference on Knowledge discovery and data mining, Edmonton, Alberta, Canada, 2002, pp. 366–375.

[22] J. R. Goodall and D. R. Tesone, “Visual Analytics for Network Flow Analysis,” 2009 Cybersecurity Applications & Technology Conference for Homeland Security, Washington, DC, USA, pp. 199-204, 2009.

[23] T. Itoh, H. Takakura, A. Sawada, and K. Koyamada, “Visualization of Network Intrusion Detection Data,” IEEE Computer Graphics and Applications, vol. 26, no. 2, pp. 40-47, 2006.

[24] V. Chandola, A. Banerjee, and V. Kumar, “Anomaly detection,” ACM Computing Surveys, vol. 41, no. 3, pp. 1-58, 2009.

[25] R. Werlinger, K. Hawkey, and K. Muldner, “The challenges of using an intrusion detection system: is it worth the effort?,” SOUPS ’08 Proceedings of the 4th symposium on Usable privacy and security, Pittsburgh, Pennsylvania, USA, pp. 107-108, 2008.

[26] R. Werlinger, K. Hawkey, and K. Beznosov, “An integrated view of human, organizational, and technological challenges of IT security management,” Information Management & Computer Security, vol. 17, no. 1, pp. 4-19, 2009.

[27] R. Werlinger, K. Muldner, K. Hawkey, and K. Beznosov, “Preparation, detection, and analysis: the diagnostic work of IT security incident response,” Information Management & Computer Security, vol. 18, no. 1, pp. 26–42, 2010.

[28] R. Werlinger, K. Muldner, K. Hawkey, and K. Beznosov, “Towards Understanding Diagnostic Work During the Detection and Investigation of Security Incidents,” in Proceedings of the Third International Symposium on Human Aspects of Information Security & Assurance (HAISA 2009), 2009, no. Haisa, p. 119.

[29] J. R. Goodall, W. G. Lutters, and A. Komlodi, “Developing expertise for network intrusion detection,” Information Technology & People, vol. 22, no. 2, pp. 92–108, 2009.

[30] J. R. Goodall, W. G. Lutters, and A. Komlodi, “I know my network: collaboration and expertise in intrusion detection,” in Proceedings of the 2004 ACM conference on Computer supported cooperative work, Chicago, Illinois, USA, pp. 342–345, 2004.

[31] S. Mitropoulos, D. Patsos, and C. Douligeris, “On Incident Handling and Response: A state-of-the-art approach,” Computers & Security, vol. 25, no. 5, pp. 351-370, 2006.

[32] S. Axelsson, “The base-rate fallacy and the difficulty of intrusion detection,” ACM Transactions on Information and System Security, vol. 3, no. 3, pp. 186-205, Aug. 2000.

[33] C.-F. Tsai, Y.-F. Hsu, C.-Y. Lin, and W.-Y. Lin, “Intrusion detection by machine learning: A review,” Expert Systems with Applications, vol. 36, no. 10, pp. 11994-12000, Dec. 2009.

[34] K. Lundholm, T. Sommestad, M. Persson, T. Gustafsson, and A. Hunstad, “Detektion av IT-attacker Övningsuppställning och insamlad data,” FOI-R-3342-SE, FOI, Linköping, Sweden, 2011.

[35] P. Ning and Y. Cui, “Constructing attack scenarios through correlation of intrusion alerts,” Proceedings of the 9th ACM conference on Computer and communications security, Washington, DC, USA, pp. 245-254, 2002.

[36] S. Eckmann and G. Vigna, “STATL: An attack language for state-based intrusion detection,” Journal of Computer Security, vol. 10, no. 1-2, pp. 71-103, 2002.

[37] F. Cuppens and R. Ortalo, “LAMBDA: A Language to Model a Database for Detection of Attacks,” in Recent Advances in Intrusion Detection, Springer Berlin / Heidelberg, pp. 197-216, 2000.

[38] O. Dain and R. Cunningham, “Fusing a heterogeneous alert stream into scenarios,” in proceedings of the ACM Workshop on Data Mining for Security Applications, Philadelphia, PA, USA, pp. 1-13, 2001.

[39] H. Debar and A. Wespi, “Aggregation and correlation of intrusion-detection alerts,” Proceedings of the 4th International Symposium on Recent Advances in Intrusion Detection, Springer-Verlag, London, UK, UK, pp. 85-103, 2001.

[40] R. A. Fisher, “On the interpretation of χ 2 from contingency tables, and the calculation of P,” Journal of the Royal Statistical Society, vol. 85, no. 1, pp. 87–94, 1922.