ANOMALIAPOHJAINEN NIDS-JÄRJESTELMÄ

Toiminnan tehostaminen uudelleenklusterointime-netelmällä

Jukka Sormunen

Opinnäytetyö
Toukokuu 2013

Ohjelmistotekniikan koulutusohjelma
Tekniikan ja liikenteen ala

JYVÄSKYLÄN AMMATTIKORKEAKOULU
JAMK UNIVERSITY OF APPLIED SCIENCES

JYVÄSKYLÄN AMMATTIKORKEAKOULU
JAMK UNIVERSITY OF APPLIED SCIENCES

OPINNÄYTETYÖN
KUVAILULEHTI

Tekijä(t) SORMUNEN, Jukka	Julkaisun laji Opinnäytetyö	Päivämäärä 09.05.2013
	Sivumäärä 42	Julkaisun kieli Suomi
	Luottamuksellisuus ( ) saakka	Verkkojulkaisulupa myönnetty (x)

Työn nimi
ANOMALIAPOHJAINEN NIDS JÄRJESTELMÄ
Toiminnan tehostaminen uudelleenklusterointimenetelmällä

Koulutusohjelma
Ohjelmistotekniikan koulutusohjelma

Työn ohjaaja(t)
SALMIKANGAS, Esa

Toimeksiantaja(t)
Relator Oy

Tiivistelmä

Tämän tutkimuksen tavoitteena oli esitellä ratkaisu anomaliapohjaisten IDS-järjestelmien ongelmaan. Anomaliapohjainen IDS-järjestelmä tuottaa suuren määrän vääriä positiivisia havaintoja ja tarkoituksena oli saada niiden määrä tasolle, joka on realistisesti ihmisen analysoitavaissa.

Havaittujen anomalioiden analysoinnista johtuvaa työmäärää saataisiin vähennettyä suoraan hylkäämällä niistä tietty osa. Tämä johtaa kuitenkin sattumanvaraisuuteen verkkohyökkäysten tunnistamisessa. Toinen vaihtoehto oli kehittää tapa, jolla IDS-järjestelmä voisi oppia tunnistamaan oikeat havainnot vääristä ja näin vähentää havaittujen anomalioiden määrää.

Havaittujen anomalioiden määrän vähentäminen toteutiin IDS-järjestelmään tehdyllä uudelleenklusteroinitimenetelmällä, jossa järjestelmä tutkii havaitsemiaan anomalioita niin kauan, kunnes haluttu lopputulos saavutetaan. Lopuksi testattiin toteutetun uudelleenklusteroinitimenetelmän vaikutusta IDS-järjestelmän kykyyn havaita verkkohyökkäykset.

Toteutetulla uudelleenklusterointimentelmällä saavutettiin hyviä tuloksia väärien ja oikeiden positiivisten havaintojen suhteeseen. Väärien positiivisten määrä väheni suhteessa oikeisiin positiivisiin havaintoihin. Verkkohyökkäyksiä sisältävällä materiaalilla tehdyt testit osoittivat menetelmän tehokuuden, kun hyökkäysten tunnistustarkkuus kasvoi aikaisemmasta noin 10 % tarkkuudesta aina 47 % asti.

Avainsanat (asiasanat)

Anomalia, IDS, NIDS, tunkeutumisen havaitsemisjäjestemä, verkkohyökkäys, verkkoliikenne

Muut tiedot

JYVÄSKYLÄN AMMATTIKORKEAKOULU
JAMK UNIVERSITY OF APPLIED SCIENCES

DESCRIPTION

Author(s)	Type of publication	Date
SORMUNEN, Jukka	Bachelor´s Thesis	09.05.2013
	Pages 42	Language Finnish
	Confidentiality ( )	Permission for web publication (x)

Title
ANOMALY-BASED NETWORK INTRUSION SYSTEM
Improving functionality with reclustering method

Degree Programme
Software Engineering

Tutor(s)
SALMIKANGAS, Esa

Assigned by
Relator Oy

Abstract

The aim of this thesis was to introduce a solution to the problem of Anomaly-based Network Intrusion System. The problem is the high false-positive rate of A-NIDS.

Some of the detected anomalies could be rejected and that way the workload of the network administrator could be reduced; however, this way the attack detection rate of the IDS would be randomized. Therefore, the second option was to develop a method where IDS would learn to distinguish false-positives from true-positives.

A method to reduce the number of false-positives was developed with reclustering. Reclustering-method makes the IDS inspect the detected anomalies until the given amount of anomalies remains. Lastly it was tested how the reclustering would affect the IDS's ability to detect network attacks.

Good results were achieved to the true-false positive rates with the developed reclustering method. The number of false positive rates decreased in contrast to true positives. The effect on the system's real capacity to detect network attacks became clear when it was tested with material containing real network attacks. The detection rate rose from 10 % to approximately 47 %.

Keywords
Anomaly, Intrusion Detection System, IDS, network attack, network traffic, NIDS,

Miscellaneous

SISÄLTÖ

KUVIOT

TAULUKOT

KÄSITTEET

Anomaly, anomalia	Poikkeus, epänormaalius. Poikkeama normaalista
DARPA	Defense Advanced Research Projects Agency. Yhdysvaltalojen asevoimien tutkimusorganisaatio.
DDoS	Distributed Denial of Service. Verkkohyökkäystapa, jolla estetään palvelimen toiminta kuorimittamalla palvelinta useilla yhtäaikaisille yhteyksillä
Ethernet	Pakettipohjainen verkkoratkaisu
FTP	File Transmission Protocol. TCP-protokollaa käyttävä tiedostonsiirtomenetelmä
HIDS	Host Intrusion Detection System. Tarkkailee tietokonejärjestelmän tilaa ja käyttäytymistä
HTTP	Hypertext Transfer Protocol. Protokolla, jota selaimet ja WWW-palvelimet käyttävät tiedonsiirtoon
ICMP	Internet Control Message Protocol. TCP/IP:ssä käytettävä kontrolliprotokolla
IDS	Intrusion Detection System. Järjestelmä tunkeutumisien havaitsemiseen
IP	Internet Protocol
MAC	Media Access Control
NIDS	Network Intrusion Detection System. Verkkohyökkäyksien havaitsemisjärjestelmä
P2P	Peer-to-peer, vertaisverkko. Verkko, jossa jokainen kone toimii palvelimena että asiakkaana verkon muille jäsenille.

SMTP	Simple Mail Transfer Protocol. Sähköpostissa käytettävä TCP-pohjainen protokolla
TCP	Transmission Control Protocol
TCP/IP	Transmission Control Protocol / Internet Protocol. Usean tietoverkkoprotokollan yhdistelmä
UDP	User Datagram Protocol

1. TYÖN LÄHTÖKOHDAT

1.1 Tutkimuksen tavoitteet

Tämän tutkimuksen tavoitteena on esitellä ratkaisu anomaliapohjaisten IDS-järjestelmien ongelmaan. Anomaliapohjainen IDS-järjestelmä tuottaa suuren määrän vääriä positiivisia havaintoja. Havaintojen määrä on tarkoitus saada tasolle, joka on realistisesti analysoitavissa. Havaittujen anomalioiden analysointiin tarvitaan ammattilainen.

Yksinkertaisesti havaittujen anomalioiden määrän vähentäminen onnistuisi suoraan hylkäämällä tietty osa havaituista anomalioista. Anomalioiden suora hylkääminen ilman niiden analysointia johtaisi sattumanvaraisuuteen verkkohyökkäysten tunnistamisessa. Toinen vaihtoehto on antaa IDS-järjestelmän tutkia havaitut anomaliat useamman kerran ja oppia tunnistamaan juuri ne oikeat anomaliat, jotka voivat johtua verkkohyökkäyksestä. Tätä toista lähestymistapaa kutsumme uudelleenklusteroinniksi, koska tässä työssä oleellisena osana oleva IDS-järjestelmä käyttää klusterointimenetelmiä verkkoliikenteen tutkimiseen.

Havaittujen anomaloiden määrä pitäisi saada sille tasolle, jolloin IDS-järjestelmää käyttävä henkilö voisi ne analysoida järkevän ajan puitteissa. Käyttäjälle toteutetaan mahdollisuus itse määritellä, kuinka monta anomaliahavaintoa IDS-järjestelmä antaa. Järjestelmän toteutus on esitelty luvussa 3.

Tavoitteena on myös tutkia kuinka toteutettava uudelleenklusterointimenetelmä vaikuttaa IDS-järjestelmän verkkohyökkäysten havainnointikykyyn. Tämän mittaamiseen käytämme juuri tähän tarkoitukseen luotua datasettiä. Testaaminen ja testitulokset esitellään luvuissa 4 ja 5, tutkimuksen lopputulokset luvussa 6.

1.2 Toimeksiantaja

Tämän tutkimuksen toimeksiantajana toimi Relator Oy, joka tarjoaa korkealaatuista toimittaja- ja teknologiariippumatonta asiantuntemusta. Relator Oy:n erikoisalaa ovat tietojärjestelmien hankintakonsultointi, tietoturva, tiedon- ja sisällönhallinta, ohjelmistokehitys sekä tietotekniikan konsultointi ja koulutus. Yrityksellä on vahvaa toimialapohjaista ICT-osaamista erityisesti kansallisen turvallisuuden, terveydenhuollon, pankki -ja rahoitusalan sekä eläkejärjestelmien parista. Yrityksen toimipaikka on Jyväskylässä.

2. TEKNIIKAT

2.1 Verkkoliikennetekniikat

Verkkokerrokset

Tässä luvussa pohjustetaan hieman verkkoliikenteen perusteita esittelemällä verkkoliikenteen kerrokset ja kuinka ne liittyvät verkkohyökkäyksien havaitsemiseen.

Verkon arkkitehtuuri voidaan esittää OSI–mallin (Open System Interconnection Reference Model) ja TCP/IP-viitemallin avulla.

OSI-malli kuvaa tiedonsiirtoprotokollien yhdistelmän seitsemässä kerroksessa. Jokainen kerroksista käyttää sen alapuolisen kerroksen tarjoamia palveluja ja tarjoaa palvelujaan yläpuoliselle kerrokselle. OSI–malli kehitettiin 1980–luvulla ja on ISO:n (International Organization for Standardization) kansainvälinen standardi. Taulukossa 1 on esitetty OSI-mallin kerrokset lyhyesti selityksineen.

TAULUKKO 1. OSI-malli

Sovelluskerros (Application Layer)	Sovellusten viestintään käyttämä kerros
Esitystapakerros (Presentation Layer)	Merkistökoodaus
Istuntokerros (Session Layer)	Istuntojen hallinta yhteyksissä
Kuljetuskerros (Transport Layer)	Pakettien toimittaminen ja vastaanottaminen
Verkkokerros (Network Layer)	Tarjoaa päästä-päähän yhteyden erilaisten verkko-ratkaisujen ylitse
Siirtokerros (Datalink Layer)	Kehystää ylemmän kerroksen tietoliikennepaketin fyysistä kerrosta varten
Fyysinen kerros (Physical Layer)	Määrittää tiedonsiirron fyysisellä tasolla, esimerkiksi kaapelia tai radioaaltoja pitkin

TCP/IP-viitemalli esittää internetverkon arkkitehtuuria. Sen nimi tulee mallin kahdesta pääprotokollasta: TCP:stä ja IP:stä. Taulukossa 2 on esitetty TCP/IP-viitemallin kerrokset ja niiden pääasiallisia protokollia.

TAULUKKO 2. TCP/IP-viitemalli

Sovelluskerros (Application Layer)	HTTP, FTP, SMTP
Kuljetuskerros (Transport Layer)	TCP, UDP
Verkkokerros (Internet Layer)	IP, ICMP
Peruskerros (Link Layer)	Ethernet

Peruskerros hoitaa datan välittämisen luotettavasti.

Verkkokerros reitittää paketit oikeisiin osoitteisiinsa verkon sisällä tai verkkojen välillä. Tämän kerroksen pääprotokolla on Internet Protocol (IP). IP-protokolla määrittää pakettien osoitetiedot.

Kuljetuskerros tarjoaa kommunikoinnin kahden eri isännän välillä. Yleisimmät protokollat tässä kerroksessa ovat Transmission Control Protocol (TCP) ja User Datagram Protocol (UDP). TCP tarjoaa yhteyden luotettavaan, kaksisuuntaiseen datavirtaan ja UDP tarjoaa yksinkertaisemman, mutta epäluotettavamman yksisuuntaisen yhteyden.

Sovelluskerros määrittää ohjelmien rajapinnan verkkoon. Tämän kerroksen tyypillisimpiä protokollia ovat HTTP, FTP ja SMTP. Tämä kerros antaa ohjelmille tavan kommunikoida keskenään käyttäen jotain kerroksen protokollista tai implementoimalla jonkin oman protokollan. Itse ohjelma ei kuulu tähän kerrokseen. Kerros vain tarjoaa ohjelmille palvelun kommunikointiin.

Yksinkertaistettuna IP-paketti voidaan esittää taulukon 3 mukaan. Paketti sisältää jatkuvan sarjan bittejä, jotka voidaan jakaa osiin paketin käyttämien protokollien mukaan. Esimerkin paketti on HTTP-komento Ethernet-verkossa. Ethernet-, IP- ja TCP-otsikot sisältävät paketin perus-, verkko- ja kuljetuskerroksen informaation. Sovelluskerroksen informaatio koostuu HTTP otsikosta ja komennosta.

TAULUKKO 3. Yksinkertaistettu HTTP-paketti käyttäen TCP/IP:tä

Peruskerros	Verkkokerros	Kuljetuskerros	Sovelluskerros	
Ethernet otsikko, MAC (14 bittiä)	IP otsikko (20 bittiä)	TCP otsikko (20 bittiä)	HTTP otsikko	HTTP toiminto

Peruskerros tarjoaa meille vain muutaman ominaisuuden verkkohyökkäysten havaitsemiseen. Ethernet II, joka on yleisin peruskerroksen toteutus, sisältää MAC-osoitteet (Media Access Control), tiedon seuraavan verkkokerroksen protokollasta, paketin ja CRC-tarkistussumman. (Paananen, 2010)

IP-protokolla

Internet Protocol (IP) huolehtii IP-tietoliikennepakettien toimittamisesta perille pakettikytkentäisessä verkossa. IP kuuluu TCP/IP-mallin verkkokerrokseen. IP-paketit toimitetaan perille käyttäen IP-osoitteita. (Internet Protocol, 2012)

TCP-protokolla

Transmission Control Protocol (TCP) on yksi tärkeimmistä tietoliikenneprotokollista TCP/IP-protokollaperheessä, jolla luodaan yhteyksiä tietokoneiden välille. TCP-protokolla huolehtii myös lähetettyjen pakettien saapumisesta oikeassa järjestyksessä ja tarvittaessa hävinnyt paketti voidaan lähettää uudestaan vastaanottajalle perustuen TCP:n vuonvalvonta- ja ruuhkanhallintamekanismeihin. (Transmission Control Protocol, 2012)

UDP-protokolla

User Datagram Protocol (UDP) on tiedonsiirtoprotokolla, jolla siirretään dataa IP-pohjaisen verkon yli. UDP eroaa TCP:stä siten, että protokolla ei suorita alkukättelyä, tarkista paketin eheyttä tai varmista paketin pääsyä vastaanottajalle. UDP on yhteydetön protokolla, koska isäntä – isäntä-yhteyttä ei muodosteta ollenkaan. UDP keskittyy vain datan lähettämiseen vastaanottajalle. Näin saadaan TCP:tä parempi suorityskyky tiedonsiirrossa, kun fyysinen verkko on luotettava. (User Datagram Protocol, 2012)

IP-, TCP- ja UDP-pakettien otsikot

Tässä luvussa esitellään IP, TCP ja UDP –pakettien otsikot. Tässä työssä käsitellään vain IPv4-protokollaa olevaa verkkoliikennettä, joten vain IPv4 esitellään ja IPv6 jätetään pois.

IP

Kuviossa 1 on esitelty IPv4-paketin rakenne.

0-3	4-7	8-15	16-18	19-31
Versio	IHL	Differiantiated Services	Kokonaispituus	
Identification			Liput	Fragment offset
TTL		Protokolla	Otsikon tarkistussumma	
Lähdeosoite				
Kohdeosoite				
Optiot ja Data				

KUVIO 1. IP-paketti (RFC Sourcebook 2012a).

Versio (4 bittiä): Paketin IP-versio.

IHL (Internet Header Length, 4 bittiä): IP-paketin otsikkotietojen koko.

Differentiated services (8 bittiä): Määrittää, kuinka pakettia täytyy verkossa käsitellä. Kentän käyttö määriteltiin uudelleen RFC 2474:ssä vuonna 1998. Tämän kentän bitit määrittävät, kuinka paketit ohjataan verkon solmukohdissa.

Kokonaispituus (16 bittiä): IP-paketin koko bitteinä.

Tunniste (16 bittiä): Lähde – kohde-parin kanssa käytettävä tieto, jolla paketti tunnistetaan. Käytetään yhdistämään useiden IP-pakettien kokonaisuuksia, kun data ei mahdu yhteen pakettin.

Liput (3 bittiä): Kolme bittiä, joilla hallinnoidaan pakettien hajoavaisuutta.

Fragment offset (13 bittiä): kertoo, minne tämä paketti kuuluu kun IP-paketit ovat pirstaloituneet.

TTL (Time to Live, 8 bittiä): Paketin elinaika määrittää, kuinka monen isännän läpi paketti voi mennä ennen paketin hylkäämistä. Jokainen isäntä vähentää kentän arvoa yhdellä. Kun kentän arvo saavuttaa nollan, paketti hylätään ja ICMP viesti paketin elinajan loppumisesta lähetetään takaisin paketin lähettäjälle.

Protokolla (8 bittiä): Määrittää paketin korkeamman tason protokollan. Kaikki verkkokerroksen protokollat, (TCP, UDP jne.), tunnistetaan yksilöllisellä numerolla.

Otsikon tarkistussumma (16 bittiä): Laskennallinen arvo, jolla vastaanottaja tarkistaa tarkistaa paketin eheyden. IP laskee tarkistussumman vain omasta otsikkotiedoistaan. Korkeamman tason protokolla tarkistaa datan eheyden.

Lähdeosoite (32 bittiä): Paketin lähettäjän IP-osoite.

Kohdeosoite (32 bittiä): Paketin vastaanottajan IP-osoite.

Optiot (muuttuva pituus): Harvoin käytössä oleva tieto. Täytteeksi lisätään 0-arvoisia bittejä, kun otsikon koko ei ole kokonaisluku.

Data: Sisältää korkeamman tason protokollan datan.

TCP

Kuviossa 2 on esitelty TCP-paketin rakenne.

0-3	4-6	7-9	10-15	16-31
Lähdeportti				Kohdeportti
Järjestysnumero				
Kuittausnumero				
Otsikon pituus	Varattu	ECN	Liput	Ikkunan koko
Tarkistussumma				Kiireellisyysosoitin
Optiot ja täyte				
Data				

KUVIO 2. TCP-paketti. (RFC Sourcebook, 2012b)

Lähdeportti (16 bittiä): Määrittää paketin lähettäjän portin numeron.

Kohdeportti (16 bittiä): Määrittää paketin vastaanottajan portin numeron.

Järjestysnumero (32 bittiä): Määrittää paketin datan ensimmäisen bitin järjestysnumeron avoimessa TCP-yhteydessä. Jokainen paketti samassa TCP-yhteydessä lisää järjestysnumeroa datan bittien määrän mukaan. Vastaanottaja voi käyttää järjestysnumeroa järjestellessään paketit oikeaan järjestykseen.

Kuittausnumero (32 bittiä): Vastaanottajan lähettämä numero seuraavasta odottamastaan paketista.

Otsikon pituus (4 bittiä): Paketin otsikkotietojen koko.

Varattu (3 bittiä): Varattu tulevaisuuden käyttöä varten. Tämän on sisällettävä vain nollia.

ECN (3 bittiä): Explicit Congestion Notification, valinnainen tieto verkon ruuhkatilanteen määrittämiseen.

Liput (6 bittiä): Koostuu kuudesta bitistä sisältäen kuusi lippua, joita käytetään TCP-yhteyden hallintaan.

Ikkunan koko (16 bittiä): Tässä kerrotaan lähettäjälle, montako bittiä dataa vastaanottaja on valmis vastaanottamaan. Tätä käytetään datavirran hallinnointiin, jolloin lähettäjä ei lähetä dataa nopeammin, kuin vastaanottaja on valmis prosessoimaan.

Tarkistussumma (16 bittiä): TCP-otsikon ja datan tarkistussumma, jolla tarkistetaan paketin virheettömyys.

Kiireellisyysosoitin: (16 bittiä): Määrittää paketin tärkeellisyyden. Tärkeä tieto, kun Liput-kentässä on URG-lippu asetettuna.

Optiot ja täyte (32 bittiä): Valinnaiset tiedot, joilla voidaan jatkaa toisien kenttien tietoja.

Data: (32 bittiä): Paketissa välitettävä todellinen tieto.

UDP

Kuviossa 3 on esitelty UDP-paketin rakenne.

0-15	16-31
Lähdeportti	Kohdeportti
Otsikon pituus	Tarkistussumma
Data	

KUVIO 3. UDP-paketti. (RFC Sourcebook, 2012c)

Lähde- ja kohdeporttit (16 bittiä): Samoin kuin TCP-paketissa. Jos pakettiin vastaaminen ei ole tarpeellinen, voidaan lähdeportti-kenttä jättää tyhjäksi.

Otsikon pituus (16 bittiä): Otsikkotietojen ja datan yhteenlaskettu pituus. Kentän minimipituus on 8.

Tarkistussumma (16 bittiä): Paketin otsikkotietojen ja datan tarkistussumma, jolla varmistetaan paketin virheettömyys. Valinnainen tieto Ipv4:ssä.

Data (32 bittiä): Paketissa välitettävä todellinen tieto.

2.2 Tunkeilijan havaitsemisjärjestelmä (IDS)

Tunkeilijan havaitseminen voidaan määritellä tekona, jolla havaitaan toimet, jotka yrittävät vaarantaa luottamuksellisuuden, eheyden tai resurssin saatavuuden. Toisin sanoen tunkeilijan havaitsemisjärjestelmällä pyritään tunnistamaan yksilöt, jotka yrittävät kiertää tai horjuttaa paikallaan olevia turvajärjestelmiä. (Intrusion Detection FAQ)

IDS-järjestelmän toteutusvaihtoehtoja

Tunkeilijan havaitsemisjärjestelmät voidaan jakaa verkkopohjaisiin (Network IDS, NIDS), isäntäpohjaisiin (HIDS) ja fyysisiin (Physical IDS).

NIDS, verkkopohjainen tunkeilijan havaitsemisjärjestelmä, yrittää tunnistaa luvattoman, laittoman ja poikkeavan käyttäytymisen pelkästään verkkoliikenteen perusteella.

NIDS tarkkailee verkkoliikennettä ja antaa hälytyksen havaitessaan verkkohyökkäyksen. NIDS havaitsee hyökkäykset verkkohyökkäysten sormenjälkien tai poikkeavan verkkoliikenteen perusteella. Verkkohyökkäysten sormenjälkiin perustuvaa NIDS-järjestelmää kutsutaan Sormenjälkipohjaiseksi IDS-järjestelmäksi ja verkkoliikenteen poikeamiin perustuvaa Anomaliapohjaiseksi IDS-järjestelmäksi. Nämä IDS-järjestelmän toteutustavat esitellään luvuissa 2.2.2 ja 2.2.3.

Verkkoliikenteen tunkeutumisen havaitsemisjärjestelmä on asennettuna verkon osaan ja tarkkailee verkon liikennettä. Havaitessaan verkkohyökkäyksen järjestelmä joko estää hyökkäyksen tai vain antaa hälytyksen siitä. Tarkkailtavan verkon liikenne voidaan saada reaaliaikaisena verkosta tai tallennetusta verkkoliikenteen vedostiedostosta. Tässä tutkimuksessa perehdytään tarkemmin anomaliapohjaiseen NIDS-järjestelmään.

HIDS, Host IDS, tunnistaa luvattoman, laittoman ja poikkeavan käytöksen jossain tietyssä laitteessa. HIDS tavallisesti käsittää järjestelmään asennettavat ohjelman, joka tarkkailee ja varoittaa järjestelmän ja ohjelmien mahdollisesti haitallisesta käyttäytymisestä. Nykyään useat Antivirus-ohjelmat sisältävät jonkinlaisen HIDS toteutuksen. Esimerkiksi Sophos-antivirus sisältää HIPS:in (Host Intrusion Prevention System), joka yrittää estää uuden, aikaisemmin tuntemattoman, haittaohjelman toiminnan tietokoneessa (Sophos HIPS).

Fyysinen IDS tunnistaa uhat fyysisille järjestelmille. Fyysinen tunkeutujan havaitsemisjärjestelmä kattaa keinot, joilla varmistetaan fyysisten uhkien havaitseminen käyttäen esimerkiksi vartijoita, valvontakameroita, liiketunnistimia ja kulunvalvontaa. (Intrusion Detection FAQ)

Sormenjälkipohjainen NIDS

Sormenjälkipohjainen tunkeutujan havaitsemisjärjestelmä vertaa yhteyksiä ja kaapattuja paketteja tunnettujen hyökkäystapojen sormenjälkiin eli tunnisteisiin. Yhtäläisyyden löytyessä järjestelmä antaa siitä ilmoituksen. Tämä menetelmä tuottaa yleensä hyviä tuloksia väärien hälytyksien suhteen, mutta sillä on myös haittapuolensa. Koska sormenjälkipohjaisen IDS:n tunnistetietokanta ei ole aina välttämättä ajan tasalla, useimmissa järjestelmissä nollapäiväivähyökkäykset ja muunnellut hyökkäykset, joissa hyökkäystapaa on muunnettu poikkeavammaksi, jäävät havaitsematta. Kun jopa tunnetuista hyökkäystavasta vain hieman muunnettu hyökkäys jää todennäköisesti havaitsematta, hyökkääjällä on mahdollisuus saada hyökkäyksen alla oleva

järjestelmä tai sovellus hallintaansa. Sormenjälkipohjainen IDS tarvitsee myös henkilöstöresursseja pitämään tunnistetietokannan ajantasalla. (Bolzoni, 2009)

Anomaliapohjainen NIDS

Anomaliapohjainen IDS koulutetaan havaitsemaan poikkeamia opettamalla järjestelmä tietoverkon normaaliin liikennemalliin. Kouluttaminen voidaan tehdä reaaliaikaisena kuunnellen turvattavan verkon liikennettä tai käyttäen aiemmin kaapattua verkkoliikenteen vedostiedostoa. Kun anomaliapohjainen IDS tietää verkon normaalin liikenteen, kaikkea tästä poikkeavaa liikennettä voidaan käsitellä anomalisena.

Järjestelmän suurin hyöty on se, ettei järjestelmä tarvitse tietokantaa tunnetuista hyökkäystavoista ennen, kuin hyökkäyksiä voidaan havaita. Koska kaikki havaitut poikkeamat verkkoliikenteessä käsitellään anomalioina, voidaan tunnistaa nollapäivähyökkäykset ja tunnetuista hyökkäystavoista muunnellut hyökkäykset. Anomaliapohjaisen IDS:n tilastollisen havaitsemisperiaatteen haittapuolena järjestelmä tuottaa paljon vääriä hälytyksiä. Näin ollen datalla, jolla anomaliapohjainen IDS koulutetaan, on suuri merkitys potentiaalisen hyökkäyksen ja väärän hälytyksen erottamisessa toisistaan.

Anomaliapohjainen IDS saadaan toimimaan myös ilman, että ensin koulutetaan verkon normaalin liikenteeseen. Tätä kutsutaan valvomattomaksi oppimiseksi, jossa IDS voi jättää opetteluvaiheen kokonaan välistä ja aloittaa suoraan verkkoliikenteen tarkkailemisen. (Paananen, 2010)

Tässä työssä tarkastellaan anomaliapohjaisen IDS-järjestelmän tehokkuutta oikeiden hyökkäysten ja väärien hälytysten suhteen. Myöhemmin tässä työssä puhuttaessa IDS-järjestelmästä viitataan aina anomaliapohjaiseen verkkohyökkäysten havaitsemisjärjestelmään (Anomaly-based Network Intrusion Detection System, A-NIDS).

2.3 Verkkohyökkäykset

Hyökkäyksellä tarkoitetaan yritystä tuhota, paljastaa, muokata, vaurioittaa tai varastaa tietoa luvattomasti tietokoneesta tai tietoverkosta. Myös luvattomat käyttöoikeuksien laajentamiset ja tiedon luvaton käyttö voidaan lukea hyökkäykseksi.

Hyökkäykset voidaan jakaa aktiivisiin ja passiivisiin, sisäisiin ja ulkoisiin. Aktiivinen verkkohyökkäys tarkoittaa hyökkäystä, jolla pyritään tekemään vahinkoa tai haittaa tietokoneelle tai tietoverkolle. Passiivisella hyökkäyksellä yritetään oppia tai kerätä tietoa tietojärjestelmästä tai tietoverkosta aiheuttamatta suoraa vahinkoa tietojärjestelmälle tai tietoverkolle. Passiiviseen hyökkäykseen lukeutuu esimerkiksi luvaton verkkoliikenteen kuuntelu.

Sisäinen hyökkäys tulee organisaation sisältä, henkilön tai ohjelman suorittamana. Esimerkiksi työntekijä, jolla on luvanvarainen pääsy organisaation resursseihin, käyttää resursseja ei sallitulla tavalla tai aiheuttaa niille tahallista vahinkoa. Ulkopuolinen hyökkäys tulee organisaatioon kuulumattomalta taholta.

SANS (SysAdmin, Audit, Network, Security) instituutio raportoi nollapäivähaavoittuvuuksia käyttävien ja tarkasti kohdennettujen hyökkäysten olevan lisääntymässä (SANS 2011).

Organisaatioden kohdentaessaan turvajärjestemiä Internetiin kytkettyihin järjestemiin, saattaa sisäiseen verkkoon tai järjestelmään päässyt hyökkääjä jäädä havaitsematta. Hyökkäykset käyttävät usein kiertoreittejä hyppien järjestelmästä toiseen, kunnes saavuttavat kohteensa aiheuttaakseen vahinkoa tai varastaakseen tietoja.

Yksi tällaisista hyökkäyksistä, nimeltään Stuxnet, oli tarkasti kohdennettu tietokonemato, joka hyökkäsi teollisuudessa käytettyjä järjestelmiä ja laitteita vastaan. Nykyaikaiset tietokonemadot voivatkin olla erittäin kohdennettuja. Seuraavassa kappaleessa esitellään lyhyesti esimerkkitapaus nykyaikaisesta matotyyppisestä verkkohyökkäyksestä.

Stuxnet

Stuxnet havaittiin heinäkuussa 2010, mutta sen on todettu olleen liikkeellä jo vuotta aikaisemmin tai todennäköisesti vieläkin aiemmin. Suurin osa Stuxnet-tartunnoista löydettiin Iranista. IDS:n näkökulmasta mielenkiitoista on, olisiko anomaliapohjainen IDS voinut havaita madon sen muodostamasta verkkoliikenteestä itseään levittäessään tai verkon yli päivittäessään.

Stuxnet on monimutkainen ja kehittynyt tietokonemato. Stuxnetin kohde oli voimalaitoksissa käytettävät teollisuuden ohjausjärjestelmät (ICS, Industrial contol system). Se ohjelmoi uudelleenohjelmoitavia logiikkaohjaimia (PLC, programmable logic controller) toimimaan hyökkääjän haluamalla tavalla piilottaen samalla jälkensä. Tämän saavuttaakseen Stuxnet sisältää monia eri osia, kuten nollapäivähaavoittuvuuksien hyväksikäyttö, Windows rootkit, ensimmäinen laatuaan oleva PLC rootkit, tekniikka virustorjuntaohjemistojen harhauttamiseen, monimutkainen prosessien pistos- ja koukutuskoodi, leviäminen verkon yli, itsensä päivittäminen käyttäen p2p:tä (peer-to-peer) sekä komento- ja ohjausrajapinnat. (Falliere, Murchu & Chien. 2011)

Stuxnet levisi kopioimalla itsensä siirrettäville medioille, kuten USB-muistitikuille, käyttäen hyväkseen Microsoft Windowsin LNK/PIF-tiedostojen automaattisen suorituksen haavoittuvuutta (Microsoft Windows Shortcut ’LNK/PIF’ Files Automatic File Execution Vulnerability, BID 41732) ja sitä kautta verkkoihin suljettuihin tietoverkkoihin. Tietoverkoissa Stuxnet levitti itseään käyttäen Microsoft Windows Print Spooler-haavoittuvuutta (Microsoft Windows Print Spooler Service Remote Code Execution Vulnerability, BID 43073) ja Microsoft Windows Server Service RPC-haavoittuvuutta (Microsoft Windows Server Service RPC Handling Remote Code Execution Vulnerability, BID 31874). Nämä leviämisen mahdollistamat haavoittuvuudet olivat nollapäivähaavoittuvuuksia. Näiden haavoittuvuuksia käyttävien tapojen lisäksi Stuxnet levisi myös kopioimalla itsensä verkkojakoihin ja WinCC–tietokantapalvelimille. (Falliere, Murchu & Chien. 2011)

3. ANOMALIAPOHJAINEN IDS-JÄRJESTELMÄ

3.1 Mikä on anomalia verkkoliikenteessä

Anomalia, eli poikkeama, verkkoliikenteessä on normaalista poikkeava liikenne. Verkkoliikenteen anomaliat voivat johtua monesta eri syystä, kuten viallisesta verkkolaitteesta, verkon ylikuormituksesta ja verkkohyökkäyksestä. Nämä anomaliset tapahtumat häiritsevät verkon normaalia toimintaa.

Verkkoliikenteen anomaliat voidaan suurpiiteisesti jakaa kahteen kategoriaan. Ensimmäiseen kategoriaan voidaan sisällyttää verkon vikaantumisesta ja suorituskykvystä johtuvat anomaliat ja toiseen kategoriaan turvallisuuteen liittyvät anomaliat. Esimerkiksi palvelunestohyökkäykset ja tunkeutumisyritykset tuottavat turvallisuuteen liittyviä anomalioita. (Thottan & Ji 2003)

Tässä tutkimuksessa anomalialla tarkoitetaan yksiselitteisesti normaalista verkkoliikenteestä poikkeavaa liikennettä.

3.2 Verkkoliikenteen analysointi klusteroinnilla

Klusterointi on koneoppimistekniikka, joka perustuu tiedon ryhmittämiseen, samankaltaisuuden tai etäisyysmitan mukaan datapisteinä. Datapisteet muodostetaan valitsemalla tiedosta ominaisuuksia, joista lasketaan datapisteen koordinaatit nuloitteiseen avaruuteen. Klusterointi tapahtuu sijoittamalla lasketut datapisteet avaruuteen valitsemalla yksi datapiste klusterin keskipisteeksi ja asettelemalla muut datapisteet tämän keskipisteen läheisyyteen. Klustereiden ulkopuolelle jäävät datapisteet merkitään poikkeamiksi. Tämä on tyypillinen tapa havaita anomalioita klusteroinnilla. (Paananen, 2010)

Portnoy, Eskin & Stolfo (2001) esittelevät yksinkertaisen klusterointimenetelmän, jossa jokaisesta opetusdatan uudesta datapisteestä tulee uuden klusterin keskipiste, jos datapiste ei kuulu jo johonkin olemassa olevaan klusteriin.

Klusteroinnissa käytettävä algoritmi toimii seuraavasti. Oletetaan, että meillä on laskettavan ominaisuuden arvo *M* ja klustereille säde *W*. Lasketaan etäisyys datapisteen *d* ominaisuuden *M* ja klusterin *C* keskipisteessä olevan datapisteen välillä.

Klustereiden muodostaminen opetusvaiheessa:

1. Alusta sarja klustereita, *S*.
2. Valitse opetusdatasta datapiste *d*. Jos S on tyhjä, niin luo uusi klusteri, jossa d on klusterin keskipiste, ja lisää se S:ään. Jos S ei ole tyhjä, niin etsi klusteri C, joka on lähimpänä datapistettä *d*.
3. Jos etäisyys *d*:n ja *C*:n välillä on pienenmpi tai yhtäsuuri kuin klustereille määritetty säde *W*, liitä *d* klusteriin *C*. Jos etäisyys on enemmän kuin *W*, uusi klusteri on luotava *d*:lle.
4. Toista kohtia 2 ja 3, kunnes opetusdata on käyty kokonaan läpi.

Tässä työssä tutkittavan IDS-järjestelmä tutkii verkkoliikennettä, tarkemmin TCP/IP- ja UDP/IP-paketteja ja niiden ominaisuuksia käyttäen klusterointia. TCP/IP- ja UDP/IP-pakettien ominaisuuksien arvot eivät ole suoraan vertailukelpoisia. Jos ominaisuuksien arvot eivät ole suoraan vertailukelpoisia, ei niitä voida suoraan käyttää datapisteiden laskemisessa. Arvot ensin saatava muunnettua samalle asteikolle, että tämä vertailukelpoisuus saavutetaan. Tämä muuntaminen tapahtuu normalisoinnilla, joka on selitetty luvussa 3.3.

3.3 Pakettien ominaisuuksien normalisointi

Anomaliapohjainen IDS käyttää verkkoliikenteen ominaisuuksia hyökkäyksien tunnistamiseen. Esimerkiksi IP-paketilla on numeerisia ja nimellisiä arvoja. Klusterointimenetelmä, jota tässä työssä käytetään, perustuu datapisteiden etäisyyksiin toisistaan. Datapisteiden etäisyydet lasketaan koordinaateista, ja koordinaatit muodostuvat ominaisuuksien arvoista. Koska ominaisuuksien arvoilla on eri suuruusluokkia, on arvot ensin normalisoitava. Normalisoinnilla saadaan paketin ominaisuuksista keskenään verrattavia arvoja ja saadaan paketin hallitsevat ominaisuudet suljettua pois.

Numeerisia ominaisuuksia esimerkiksi IP-paketissa on TTL (Time To Live) ja kokonaispituus. TTL:n arvo on väliltä 0 ja 255 ja kokonaispituuden arvo välillä 0 ja 65535. Jos paketeista luodaan datapisteitä käyttäen näitä ominaisuuksia ja lasketaan luotujen datapisteiden etäisyys, niin silloin paketin kokonaispituus määräisi suuresti datapisteiden välisen etäisyyden TTL-arvon vaikutuksen olessa vähäinen. Normalisoinilla saadaan tasoitettua tämä vaikutus. (Paananen, 2010)

Ensin on päätettävä asteikko ominaisuuksien arvoille ja muutettava jokainen arvo tälle asteikolle. Tämä saadaan tehtyä käyttämällä seuraavaa kaavaa:

$$\delta = \frac{d}{dmax} \tag{1}$$

Siinä *d* on normalisoitava arvo ja *dmax* on omaisuuden maksimiarvo. Tällä normalisoinnilla saadaan kaikki arvot välille [0,1].

Standardipistemäärä saadaan laskettua kaavalla:

$$z = \frac{x - \mu}{\sigma} \tag{2}$$

Siinä x on normalisoita arvo, μ on arvojoukon keskiarvo ja σ on arvojoukon keskihajonta. Standardipistemäärä kertoo, kuinka monta pykälää arvo poikkeaa keskiarvosta. Tämä tapa tuottaa hieman enemmän normalisointiasteikkoja riippuen normalisoitavien arvojen hajonnasta. (Paananen, 2010)

Nimellisien arvojen normalisointi halutulle asteikolle ei ole yhtä suoraviivaista kuin numeerisien arvojen normalisointi. Koska nimellisen ominaisuuden arvo on ei-numeerinen, sitä ei voida suoraan verrata toiseen nimelliseen ominaisuuteen. Esimerkiksi IP-paketissa oleva tieto käytössä olevasta protokollasta numero 6 tarkoittaa TCP-protokollaa ja numero 17 UDP-protokollaa. Näiden lukujen vertaaminen ei anna meille mitään oleellista tietoa. (Paananen, 2010)

Nimellisten arvojen normalisoinnissa on käytettävä niiden esiintymistiheyttä eikä todellisia arvoja. Normalisoimalla tiheimmin esiintyvät arvot lukuun 1 ja harvemmin esiintyvät arvot lukuun 0. Nimelliset arvot normalisoidaan esiintymistiheyden mukaan kaavalla:

$$\delta = \frac{f}{fmax} \tag{3}$$

Siinä *f* on arvon esiintymistiheys ja *fmax* on suurin esiintymistiheys kaikista löydetyistä arvoista.

Normalisoitu ominaisuuden arvo merkkaa yhtä datapisteen koordinaattia ja kaikkia, tai yhtä, TCP/IP- tai UDP/IP-paketin arvoja voidaan käyttää datapisteessä. Näin datapisteelle saadaan koordinaatit *n*-uloitteiseen avaruuteen, jossa *n* on tarkastelussa mukana olevat paketin ominaisuuksien lukumäärä. (Paananen, 2010)

3.4 Anomaliapohjaisen IDS–järjestelmän ongelma

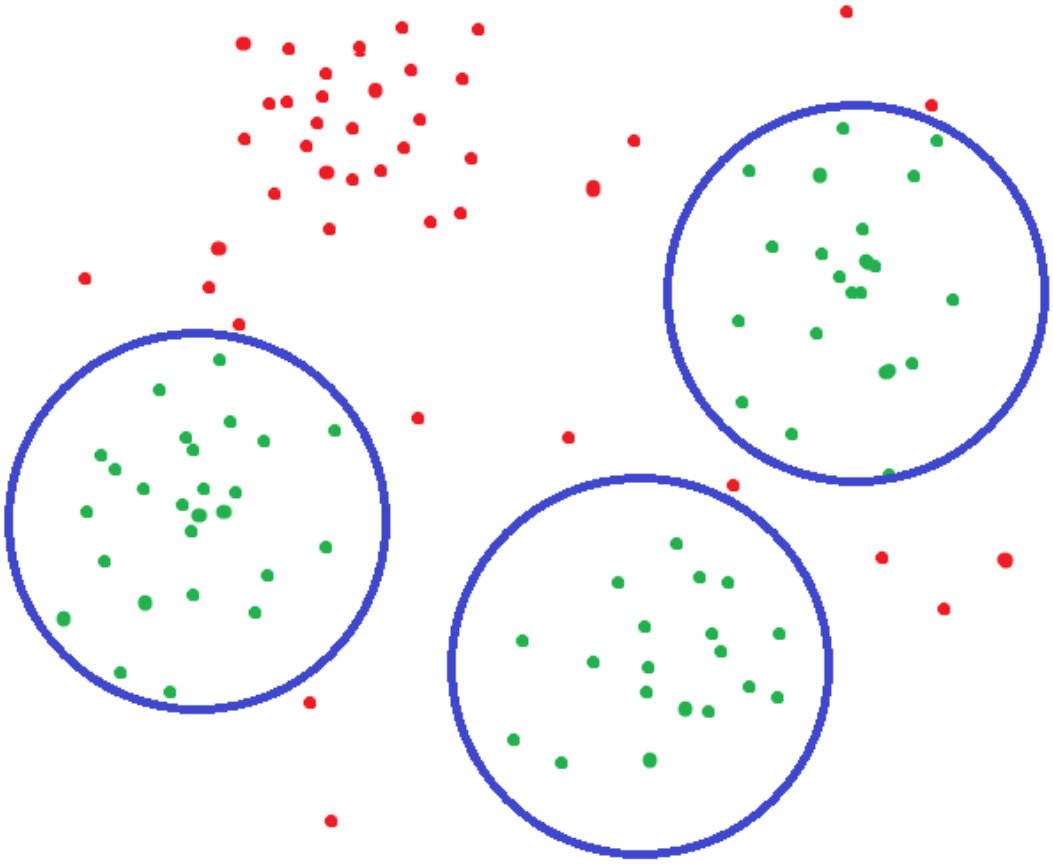
Anomaliapohjainen IDS-järjestelmän tehokkuus määräytyy sen kykyyn havaita verkkohyökkäyksiä. Anomaliapohjainen IDS-järjestelmä tuottaa myös tuottaa paljon myös vääriä hälytyksiä. Luotettavan, hyökkäysvapaan ja tosielämän verkkoliikennettä vastaavan koulutusmateriaalin tuottaminen on erittäin vaikeaa. Väärien hälytyksien ja oikeiden havaintojen suhdetta voidaankin pitää anomaliapohjaisen IDS-järjestelmän tehokkuuden mittana. Hälytysten oikeanmukaisuuden tarkastamiseen tarvitaan ih-

minen, koska järjestelmä ei osaa itse tarkastaa onko havaittu anomalia oikea verkkohyökkäys vai vain normaali poikkeama verkkoliikenteessä. Vääristä hälytyksistä käytetään myös nimeä ’väärä positiivinen’. Ongelmaan yritetään ratkaista verkkoliikenteen uudelleenklusteroinnilla. Menetelmä on yksikertaisesti esitelty luvussa 3.5.

3.5 Uudelleenklusterointi

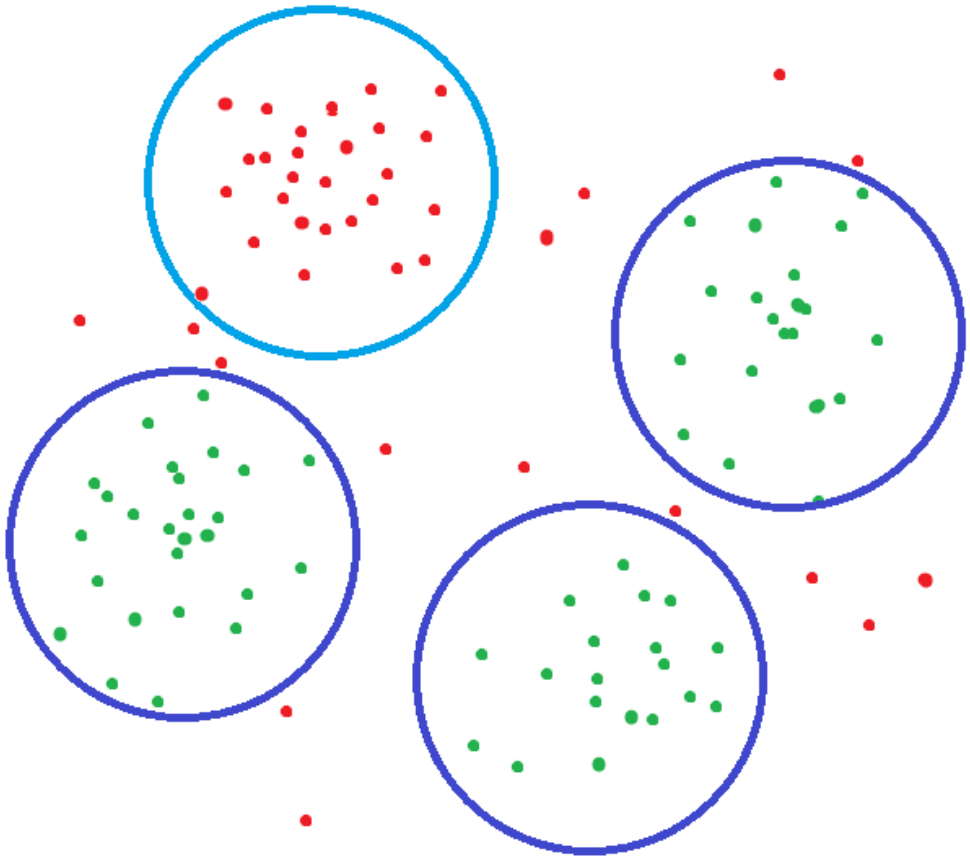
Uudelleenklusteroinnilla yritetään ratkaista edellisessä luvussa esitettyä anomaliapohjaisen IDS–järjestelmän ongelmaa väärien hälytysten ja havaittujen, oikeiden verkkohyökkäysten suhteen. Tässä esitellään yksinkertaiseti uudelleenklusteroinnin periaatetta. Tässä työssä IDS-järjestelmään toteutetusta uudelleenklusterointimetelmästä kerrotaan luvussa 3.6.

KUVIO 4 on yksinkertainen esimerkkikuvio tietoliikenteestä laskettuista datapisteistä 2-uloitteisessa avaruudessa. Siniset ympyrät ovat IDS-järjestelmän koulutusvaiheessa muodostuneet klusterit. Vihreät ja punaiset pisteet ovat verkkoliikenteestä kaapatuista paketeista laskettuja datapisteitä. Vihreät datapisteet sijoittuvat olemassa olevien klustereiden sisälle, jolloin niitä käsitellään normaaleina. Punaiset datapisteet jäävät koulutusvaiheessa luotujen klustereiden ulkopuolle, jolloin niitä käsitellään anomalioina. Klustereiden ulkopuolelle jäävät datapisteet muodostuvat tuntemattomasta verkkoliikenteestä, jota ei esiintynyt järjestelmän koulutusvaiheessa.



KUVIO 4. Yksinkertainen kuvio verkkoliikenteestä muodostetuista datapisteistä 2-uloitteisessa avaruudessa.

Kuviossa 4 punaisia, anomalioita esittäviä datapisteitä on kohtalaisen paljon verrattuna vihreisiin datapisteisiin, jotka esittävät normaalia verkkoliikennettä. Kuviossa 5 esitetään tilannetta uudelleenklusteroinnin jälkeen. Vaaleansininen ympyrä kuvastaa uutta uudelleenklusteroinnilla luotua klusteria, joka on muodostunut kuviossa 4 esiintyvän vasemman yläreunan kohtalaisen tiiviin punaisten datapisteiden ryhmän kohdalle. Vaaleansinisen klusterin sisältämää verkkoliikennettä käsitellään nyt normaalina verkkoliikenteenä. Nyt klustereiden ulkopuolelle jäävien datapisteiden määrä on huomattavasti pienenmpi.



KUVIO 5. Yksinkertainen kuvio verkkoliikenteestä muodostetuista datapisteistä 2-uloitteisessa avaruudessa uudelleenklusteroinnin jälkeen

Uudelleenklusteroinnissa esiintyy perustuvanlaatuinen ongelma. Tekniikan on oletettava, että verkkohyökkäyksiä esiintyy normaalia verkkoliikennettä paljon pienempi määrä. Portnoy, Eskin & Stolfo (2001) esittelevät ratkaisun, jossa oletetaan, että jokin tietty määrä vähiten datapisteitä sisältävistä klustereista muodostuu anomalioista. Tämän normaalin ja hyökkäyksen sisältämän verkkoliikenteen välisen suhteen määrittäminen voi olla vaikeaa, johtuen verkkoliikenteen luonteesta. Verkossa, jossa anomalisen liikenteen määrä on lähes olematon, normaali liikenne luokiteltaisiin suurelta osin anomaliseksi tuottaen suureen määrän vääriä hälytyksiä. Toisaalta paljon verkkoliikennettä aiheuttava verkkohyökkäys muodostaisi paljon datapisteitä sisältävän klusterin. Esimerkiksi palvelunestohyökkäys luokiteltaisiin tätä klusterointitapaa käyttäen normaaliksi verkkoliikenteeksi.

3.6 Uudelleenklusteroinnin toteutus

Uudelleenklusterointi toteutettiin jo olemassa olevaan IDS-järjestelmään uutena ominaisuutena. IDS-järjestelmä on toteutettu C++ kielellä ja käyttöliittymä Qt:lla. Järjestelmä pyörii Linux:issa.

Uudelleenklusterointimenetelmä tehtiin osaksi luvussa 3.2 esiteltyä yksinkertaista klusterointimenetelmää ja se sisältää seuraavat asetukset toiminnan säätämiseen.

Uudelleenklusteroinnin peruste: Tarkasteluväli, milloin uudelleenklusteroinnin tarve tarkastetaan. Vaihtoehtoina on aika, pakettimäärä tai testaustiedoston loppuminen. Aika-vaihtoehdolla uudelleenklusteroinnin tarve tarkastetaan tietyin aikavälein, pakettimäärä -vaihtoehdolla tarve tarkastetaan määritellyn pakettimäärään ylittyessä ja testaustiedoston loppuminen –vaihtoehdolla uudelleenklusteroinnin tarve tarkastetaan testaustiedoston tutkimisen jälkeen.

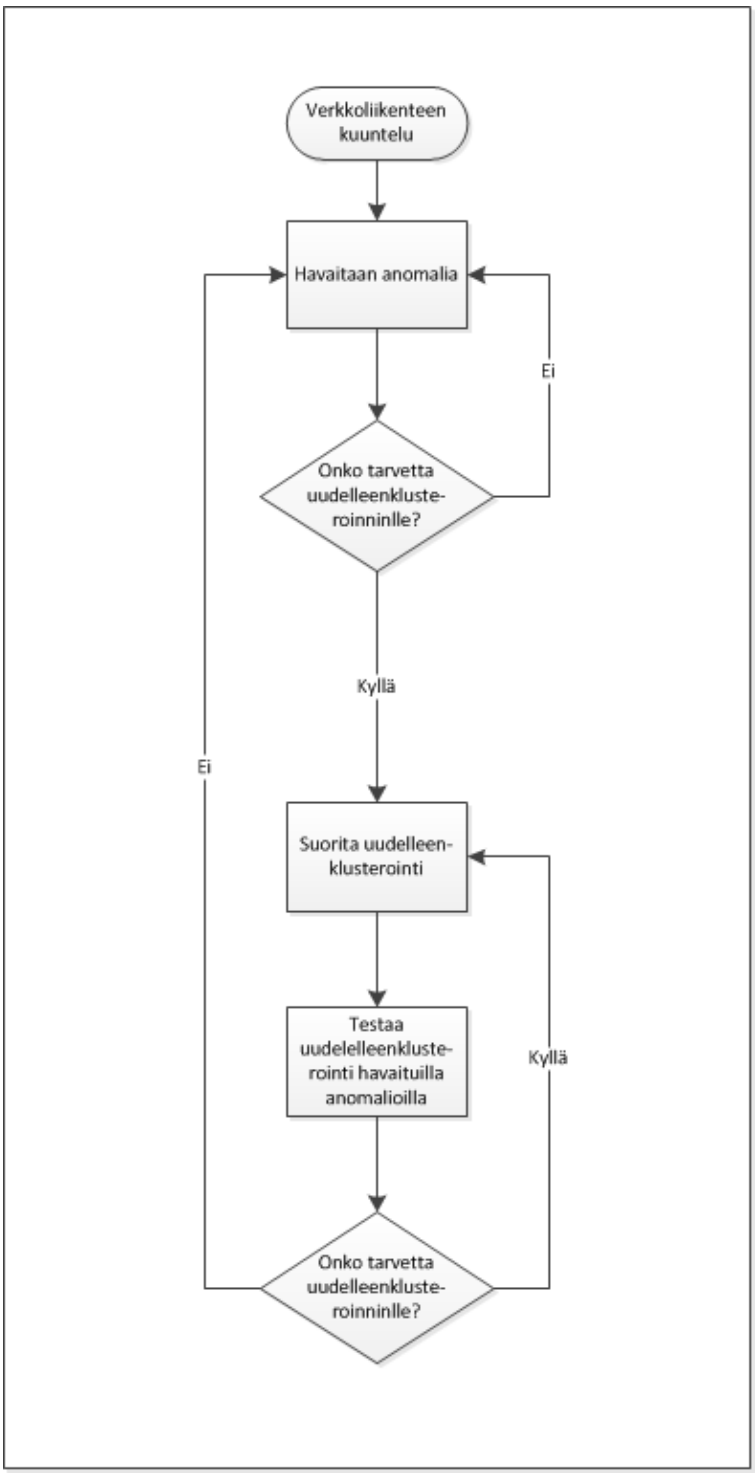
Uudelleenklusteroinnin perusteen arvo: Uudelleenklusteroinnin tarpeen laukaisuarvo. Aikamääre annetaan minuutteina tai pakettien määränä.

Anomalioiden määrän alaraja tarkasteluvälillä havaittuina anomalioina: Haluttujen anomalioiden määrän alaraja. Uudelleenklusterointi-toiminnallisuuden ollessa käytössä, anomalioiden määrä säädetään tämän luvun yläpuolelle.

Anomalioiden määrän yläraja tarkasteluvälillä havaittuina anomalioina: Haluttujen anomalioiden määrän yläraja. Uudelleenklusterointi-toiminnallisuuden ollessa käytössä, anomalioiden määrä säädetään tämän luvun alapuolelle.

Esimerkiksi käyttäjä määrittelee alarajaksi 50, ylärajaksi 100 ja tarkasteluväliksi 10000 pakettia. Tällöin joka 10000 käsitellyn paketin jälkeen tutkitaan, onko havaittujen anomaloiden määrä välillä 50-100. Jos määrä on ali annetun välin, niin uudelleenklusterointi poistaa klustereita niin kauan, kunnes havaittujen anomaloiden määrä on annetulla välillä. Jos määrä on on yli annetun välin, niin uudelleenklusterointi muodostaa klusterointialgoritmin mukaisesti uusia klustereita, kunnes jäljelle jäävien havaittujen anomalioiden määrä on annetulla välillä. Toiminta on samantapainen myös tarkasteluvälin ollessa aikamääre tai testaustiedoston loppuminen, vain uudelleen-

klusteroinnin laukaisuperuste on eri. Kuviossa 6 on esitelty uudelleenklusteroinnin toiminta vuokaaviona käytettäessä uudelleenklusteroinnin perusteena pakettimäärää.



KUVIO 6. Toteutetun uudelleenklusterointimenetelmän toiminta.

Kuviossa 6 uudelleenklusteroinnin toiminnan lähtopisteenä anomalian havaitseminen, jonka jälkeen tarkastetaan uudelleenklusteroinnin tarve. Tarpeen täyttyessä suoritetaan uudelleenklusteroinnin ensimmäinen vaihe, jolloin luodaan uusia klustereita tai vanhoja poistetaan. Tämän jälkeen testataan uudelleenklusteroinnin vaikutus syöttämällä havaitut anomaliat järjestelmälle uudestaan. Testauksen jälkeen uudelleeklusteroinnin tarve tarkastetaan uudelleen ja suoritetaan tarvitta toimenpide, joko ajetaan uudelleenklusterointi uudestaan tai palataan normaaliin toimintaan.

Uudelleenklusteroinnin jälkeinen testaus suoritetaan käyttäen joko havaittuja anomalioita tai liikennettä, josta anomaliat löydettiin. Havaittuilla anomaloilla testaaminen on nopeampaa, koska pakettien määrä on yleensä pienempi. Uudelleenklusteroinnin vaikutuksen testaaminen voitaisiin jättää myös kokonaan pois, mutta silloin jäisi pois myös iterointi. Iteroinnin myötä päästään kuitenkin tarkempaan klusterointiin.

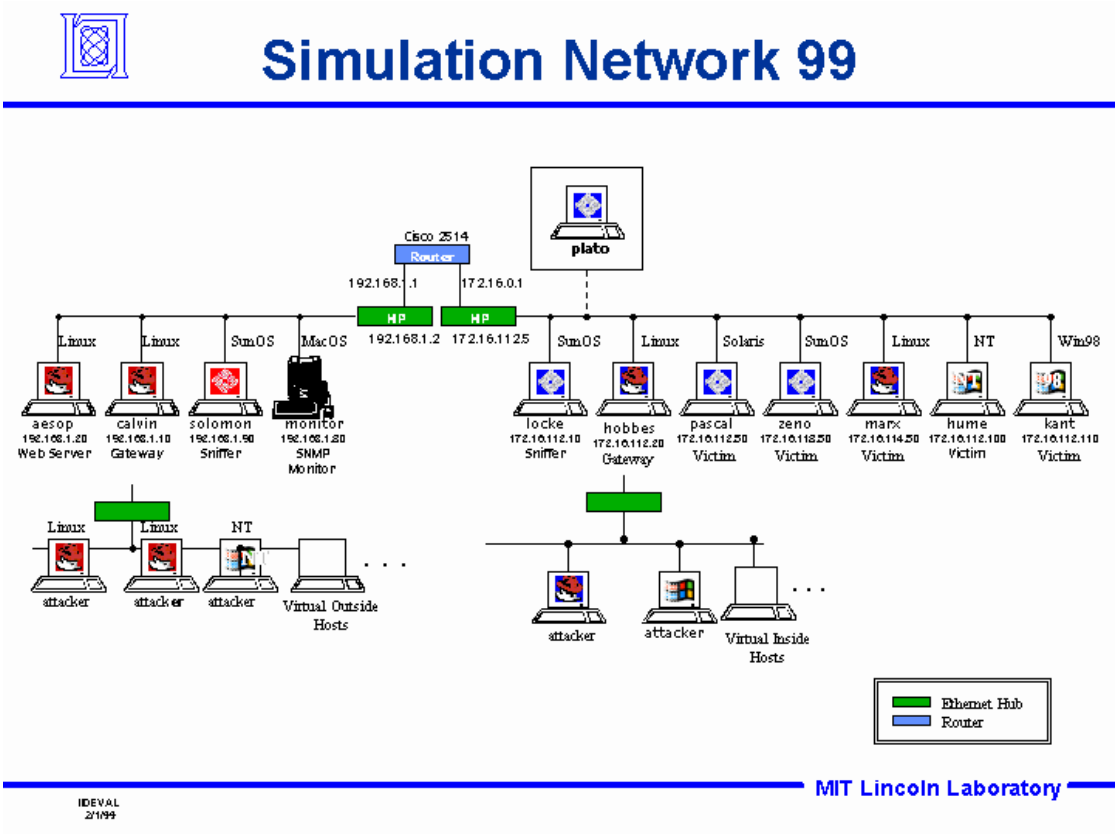
4. TESTAUS

4.1 Testausmateriaali

Tässä työssä käytettiin IDS-järjestelmän koulutus- ja testausmateriaalina DARPA 1999 -materiaalia (DARPA 1999a), JYVSECTEC-projektin (JYVSECTEC) tarjoamaa materiaalia ja Ostinato-työkalulla luotua dataa (Ostinato). DARPA 1999 testausmateriaalilla testattiin IDS-järjestelmälle tehdyn uudelleenklusterointimenetelmän vaikutuksen havaittaviin oikeisiin verkkohyökkäyksiin, JYVSECTEC -projektista saadulla materiaalilla simuloitiin järjestelmän toimivuutta reaalimaailman tilanteessa ja Ostinato:lla luodulla materiaalilla täydennettiin JYVSECTEC-materiaalilla tehtyä testausta.

Tässä luvussa esitellään kaikki käytetyt testimateriaalit. Materiaalien kanssa käytetyt testausmenetelmät löytyvät luvusta 4.3.

DARPA testimateriaali on varta vasten generoitua verkkoliikennettä IDS-järjestelmien testaamiseen ja tehokkuuden mittaamiseen. DARPA 1999 datasetti sisältää verkkoliikennettä viiden viikon ajalta maanantaista perjantaihin. Yksi päivä sisältää noin 2 miljoonaa kaapattua pakettia. Kuviossa 6 on esitetty DARPA 1999:n verkkorakenne.



KUVIO 7. DARPA 1999 verkkorakenne (DARPA 1999b)

Testimateriaali muodostuu 5 viikosta kaapattua verkkoliikennettä, joista viikot 1 ja 3 eivät sisällä verkkohyökkäyksiä. Näitä viikojen 1 ja 3 verkkoliikennettä voidaan käyttää anomaliapohjaisen IDS järjestelmän kouluttamiseen. Viikkon 2 verkkoliikenne sisältää useita verkkohyökkäyksiä, jotka olivat julkaistu IDS järjestelmien kehitystä varten. Viikkojen 4 ja 5 verkkoliikenne sisältää 201 verkkohyökkäystä, joita käytetään tyypillisesti IDS järjestelmän testaamiseen. Viikon 4 verkkoliikenteestä puuttuu yksi päivä, joka vähentää mahdollisten havaittavien hyökkäyksien maksimimäärää.

JYVSECTEC-projektin tuottama testimateriaali on satunnaista TCP/UDP-verkkoliikennettä erilaisilla volyymeillä. Liikennemäärät vaihtelevat 25 kbps ja 1

Mbps välillä. Verkkoliikenne on tyypiltään sisäverkon ja Internetin välistä verkkoliikennettä. Materiaali on jaettu liikennemäärien mukaan omiin osioihinsa, joita yhdistelemällä saadaan haluttuja muutoksia verkkoliikennemääriin. Muuttuvilla verkkoliikemäärillä testattiin IDS-järjestelmään toteutetun uudelleenklusterointimenetelmän toimivuutta. Materiaalia voi myös käyttää anomaliapohjaisen IDS-järjestelmän kouluttamiseen.

Ostinato:lla luotu materiaali on generoitua verkkoliikennettä. Ostinato:lla verkoliikennettä voi generoida halutuntyyppiseksi. Tähän tutkimusta varten luotiin 4 erilaista verkkoliikenteen vedostiedostoa, joista yksi oli tarkoitettu IDS-järjestelmän kouluttamiseen ja 3 testaamiseen. Kaikkien tiedostojen verkkoliikenne muodostui TCP/IP-paketteista nopeudella 1000 pakettia sekunissa ja pakettien määrä jokaisessa tiedostossa oli 10000. Opetusdassa pakettien lähdeosoitteet olivat 10.0.0.0/24-aliverkon sattumanvaraisia osoitteita ja kohdeosoitteet sattumanvaraisia osoitteita 10.0.10.0/24-aliverkossa. Opetusdatan kaikkien pakettien lähdeportti oli 28001 ja kohdeportti 80. Kaikki testaamiseen tarkoitettut tiedostot sisälsivät lähes vastaavaa verkkoliikennettä kuin opetusdata. Erona oli, että testaustiedostoihin oli generoitu tietty osa hieman opetusdatasta poikkeavaa liikenettä.

Ensimmäinen testaustiedoston viimeiset 100 pakettia poikkesivat muista kohdeosoittelitaan ja kohdeporteiltaan. Kohdeosoitteet olivat aliverkossa 10.10.10.0/24 ja kohdeporttina oli 8080.

Toisessa testaustiedostossa opetusdatasta poikkeavia paketteja oli 500, joissa kohdeosoitteet olivat aliverkossa 10.254.254.0/24, lähdeporttina oli 18001 ja kohdeporttina oli 800.

Kolmannessa testaustiedostossa opetusdatasta poikkeavaa liikennettä oli 50, joissa kohdeosoitteet olivat aliverkossa 10.128.128.0/24, lähdeporttina oli 46000 ja kohdeporttina oli 8080.

Ostinato-materiaalia käytettiin testaamaan uudelleenklusterointimenetelmän kykyä sopeutua muuttuviin liikennemääriin. Koska tiedostoihin oli generoitu tietty määrä anomalioita per pakettien yhteismäärä, voitiin materiaalilla myös testata IDS-järjestelmän kykyä löytää kyseiset anomaliat.

4.2 Testausympäristö

Toteutettujen ominaisuuksien testausympäristönä toimi jo olemassa oleva IDS-järjestelmä. Järjestelmä pitää sisällään oman testaustoiminnon, jossa opetusdata sekä arvioitava liikenne voidaan antaa sille pcap:eina. Tällöin testaus voidaan suorittaa yhdellä koneella, eikä tarvita oikeaa verkkoliikennettä tai verkkoinfrastruktuuria.

IDS-prototyypissä on myös sisäänrakennettu hälytysten evaluointi DARPA 1999 – materiaalia varten. Evaluointi tapahtuu käyttäen EVAL-ohjelmaa (Mahoney, 2001). EVAL raportoi oikeiden ja väärien havaintojen määrän ja mitkä hyökkäykset havaittiin.

IDS-järjestelmä pyörii virtuaalikoneessa, jossa käyttöjärjestelmänä on Ubuntu 12.04 LTS.

4.3 Testaaminen

Tässä luvussa esitellään eri testimateriaalien kanssa käytetyt testausmenetelmät. Testimateriaalit esiteltiin luvussa 4.1.

Uudelleenklusteroinnin testaamisessa käytettiin hyväksi JYVSECTEC–projektin tarjoamaa materiaalia. JYVSECTEC-materiaali oli jaoteltu eri verkkoliikennemäärien mukaan pcap-tiedostoihin. Verkkoliikennemäärät muodostuivat nopeuksista 100kbps – 400kbps kutakin yhden ja kahden tunnin ajoilta. Materiaaleja yhdistelemällä saatiin vielä aikaiseksi erilaisia verkkoliikennemääriä.

Verkkoliikennemäärän vaihtelu on olennainen osa uudelleenklusteroinnin testauksessa. Verkkoliikenteen olessa määrältää tasaista, anomalioita voidaan havaita koko-ajan suhteellisen tasainen määrä, jolloin uudelleenklusteroinnille ei ole tarvetta. Kun verkkoliikenteen määrä muuttuu, myös havaittujen anomalioiden määrä muuttuu. Uudelleenklusteroinnilla saadaan taas anomalioiden määrä halutulle tasolle. Verkko-

liikenteen määrää vaihtelemalla saadaan siis testattua kuinka tehokkaasti uudelleenklusterointi kasvattaa tai pienentää anomalioiden määrää suhteessa verkkoliikenteen määrään.

Uudelleenklusteroinnin toiminnan testaaminen muuttuvalla verkkoliikennemäärällä:

1. Koulutetaan IDS-järjestelmä valitulla verkkoliikenteellä.
2. Valitse anomalioiden määrälle ylä- ja alarajat.
3. Syötetään järjestelmään verkkoliikenttä, joka poikkeaa koulutusvaiheessa käytetystä verkkoliikenteestä, kunnes uudelleenklusteroinnin tarve täyttyy.
4. Tarkastetaan onko havaittujen anomalioiden määrä valittujen rajojen sisällä.
5. Toista vaiheita 3-4 kunnes uudelleenklusterointia ei enää tarvita.
6. Kasvatetaan liikennemäärä ja toistetaan taas vaiheita 3-4.
7. Palautetaan liikennemäärä alkuperäiseksi ja toistetaan taas vaiheita 3-4.

Uudelleenklusterointi toimi oikein kun vaiheen 3 uudelleenklusteroinnin tarpeen täyttyessä, eli kun havaittujen anomalioiden määrä ei ole vaiheessa 2 valitulla välillä, järjestelmä automaattisesti suorittaa uudelleenklusteroinnin. Uudelleenklusterointia suoritetaan niin kauan, että anomalioiden havaittujen anomalioiden määrä saadaan halutulle välille. Liikennemäärää kasvattamalla saatiin testattua uudelleenklusteroinnin toiminta liikennemäärän kasvaessa ja liikennemäärän palautettaessa alkuperäiseksi saatiin testattua uudelleenklusteroinnin toiminta liikennemäärän pienentyessä. Uudelleenklusteroinnin kyky sopeutua muuttuviin liikennemääriin mitattiin laskemalla, montako uudelleenklusterointia tarvittiin halutun tuloksen saavuttamiseksi.

DARPA 1999 -materiaalia käytettiin testaamaan uudelleenklusterointimenetelmän vaikutusta havaittavien oikeiden verkkohyökkäysten ja väärien hälytysten suhteeseen. Käytetty IDS-järjestelmä koulutettiin DARPA 1999 –materiaalin verkkohyökkäyksistä vapaalla liikenteellä, eli viikkojen 1 ja 3 verkkoliikenteellä, ja testattiin verkkohyökkäyksiä sisältävällä liikenteellä, eli viikkojen 2, 4 ja 5 verkkoliikenteellä.

Testausmenetelmä yksinkertaistettuna oli seuraavanlainen:

1. Kouluta IDS-järjestelmä DARPA 1999:sen koulutusmateriaalilla.
2. Testaa IDS-järjestelmän tehokkuus DARPA 1999:n verkkohyökkäyksiä sisältävällä materiaalilla.
3. Tarkastele, onko havaittujen verkkohyökkäysten määrä halutulla välillä.
4. Suorita uudelleenklusterointia kunnes kohdan 3 ehto täyttyy.

Tämän testin perusteena ja tulosten vertailukohtana oli Harri Paanasen opinnäytetyössä (Paananen, 2010) parhaan tunnistustuloksen antaneen testin tuloksiin käyttäen samaa IDS-järjestelmää ja samoja asetuksia. Testissä otettiin tulokset ylös ennen ja jälkeen uudelleenklusteroinnin. Vaiheen 2 tuloksia verrattiin vertailukohteen samanlaisen testin tuloksiin. Uudelleenklusteroinnin jälkeen saaduista tuloksista tarkasteltiin oikeiden ja väärien hälytysten suhdetta. Jos oikeiden ja väärien hälytysten suhdeluku parantui havaittujen oikeiden verkkohyökkäysten hyväksi, toteutettu uudelleenklusterointimenetelmä paransi IDS-järjestelmän kykyä havaita verkkohyökkäyksiä.

Uudelleenklusterointimenetelmän testausmenetelmä käyttäen Ostinato:lla luotua materiaalilla on samansuuntainen kuin JYVSECTEC-materiaalin kanssa käytetty testausmenetelmä, mutta testit ajettiin uudelleenklusterointimenetelmän kehitysvaiheessa.

5. TULOKSET

DARPA 1999 –materiaalilla tehdyt testit osoittavat toteutetun uudelleenklusteroin-timenetelmän vaikutuksen IDS-järjestelmän kykyyn havaita ne oikeat verkkohyök-käykset. Anomaliapohjaiset IDS-järjestelmät tuottavat suuren määrän hälytyksiä, joista vain osa on oikeita havaittuja verkkohyökkäyksiä.

TAULUKKO 4. DARPA 1999 –materiaalin testitulokset

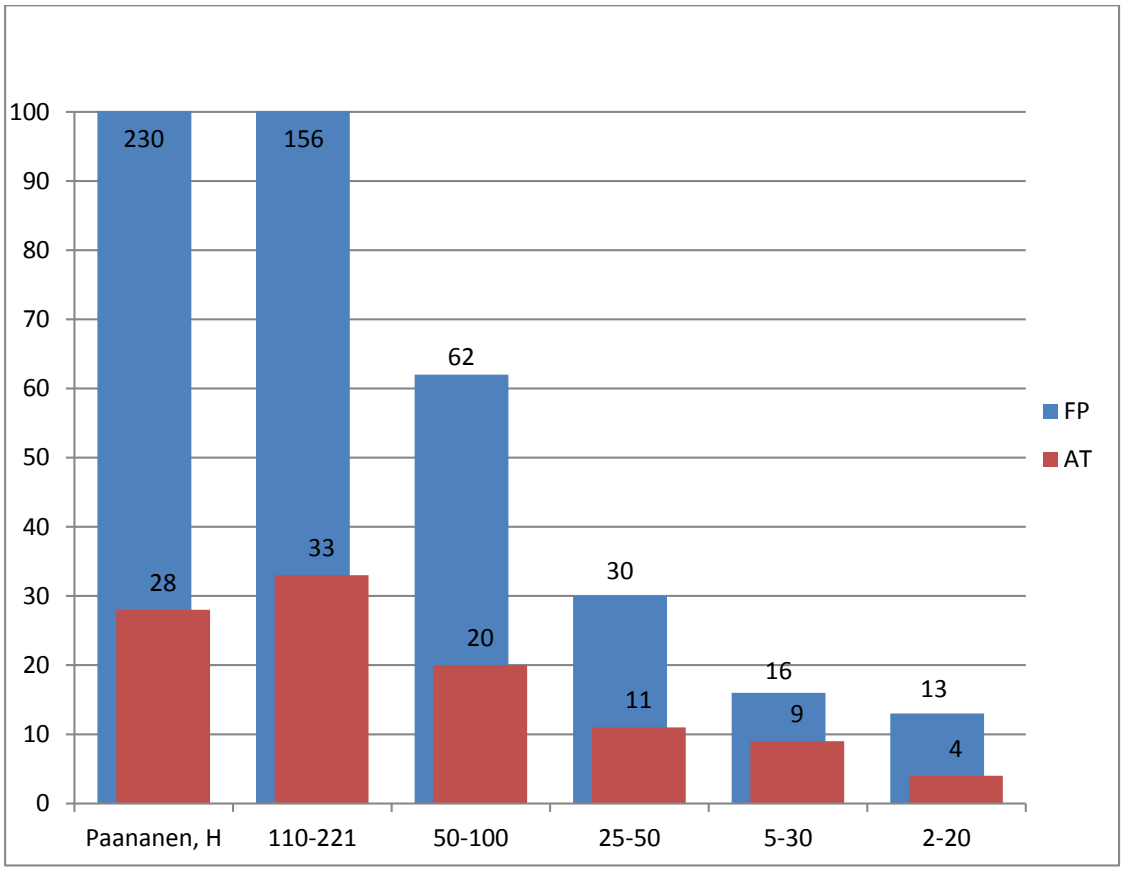
Halutut anomaliat	Havaitut anomaliat	TP	FP	TP%	FP%
-	641	141	500	22,00	78,00
110 - 220	215	59	156	27,44	72,56
50 - 100	100	38	62	38,00	62,00
25 – 50	47	17	30	36,17	63,83
5 - 30	30	14	16	46,67	53,33
2 - 20	17	4	13	23,53	76,47

Taulukossa 4 on DARPA-materiaalilla tehtyjen testien tulokset. Koulutusdatana toimi DARPA-materiaalin viikot 1 ja 3 ja testimateriaalina viikot 4 ja 5. Tarkasteltavina pa-kettien ominaisuuksina oli vastaanottajan IP, vastaanottajan kohdeportti ja TTL-arvon esiintymistiheys. Ensimmäisellä rivillä on tulokset ilman uudelleenklusterointia. TP on ’True Positive’, eli havaitut oikeat positiiviset, ja FP on ’False Positive’, eli väärä posi-tiivinen. FP% ja TP% ovat oikeiden ja väärien prosentuaaliset osuudet havaituista anomalioista.

Ilman uudelleenklusterointia anomalia havaintojen määrä oli 641, joista 141 anoma-liaa liittyi verkkohyökkäyksiin. Näin anomalioiden tunnistustarkkuudeksi saatiin 22%. Uudelleenklusterointi paransi tunnistustulosta noin 7%, kun halutujen anomalioiden määrä säädettiin lähelle DARPA 1999 –materiaalin sisältämän verkkohyökkäysten määrää. Edelleen tiukentamalla halutujen anomalioiden haarukkaa päästiin 35-47 prosentin tunnistustarkkuuteen. Haluttujen anomaloiden määrä 5-30 alkaa olla pie-

nin mahdollinen, järkevä taso, jolloin jää jo pois paljon mahdollisia havaittavia verkkohyökkäyksiä. Tämän alapuolella tunnistustarkkuus huonontui nopeasti, sillä useimmat verkkohyökkäykset sisältävät ennemmän liikennettä, kuin vain muutaman paketin. Haluttujen anomaloiden määrä säädettynä alimmilleen, uudelleenklusterointi käsittelee vain yksittäisiä anomalioita.

Kuviossa 7 on väärien hälytysten suhde havaittuihin oikeisiin verkkohyökkäyksiin. Testin tulokset on peräisin samasta testistä, jonka tulokset taulukossa 4 on esitelty. FP on väärä positiivinen anomalia ja AT havaittu oikea, yksittäinen verkkohyökkäys. AT-luku kertoo havaitut oikeat verkkohyökkäykset, ei anomalioden määrää. Yksi verkkohyökkäys voi muodostaa paljonkin anomaliseksi tunnistettavaa liikennettä.



KUVIO 8. Väärien hälytysten suhde havaittuihin verkkohyökkäyksiin.

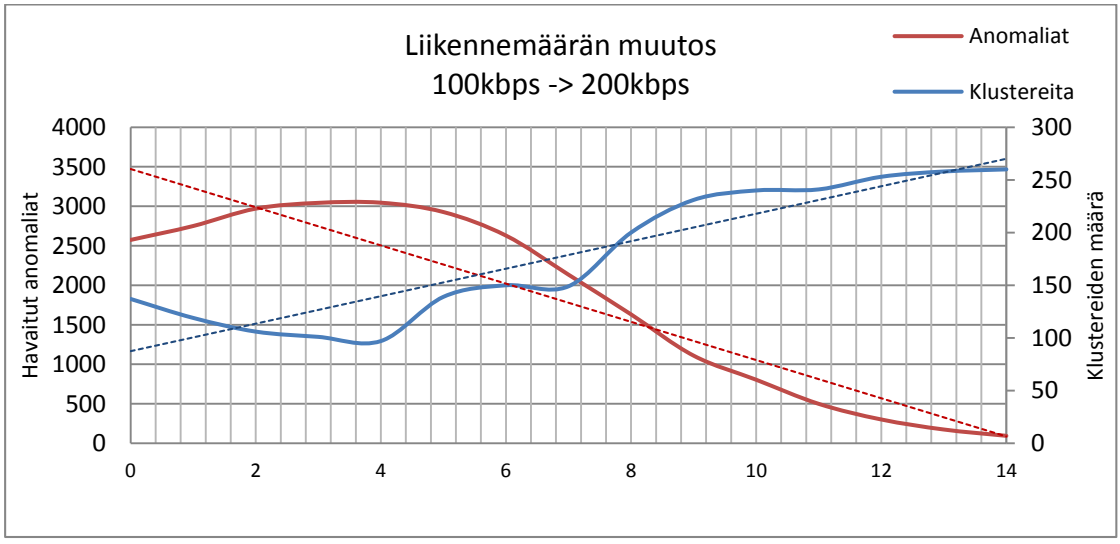
Kuvion 7 ensimmäinen pylväs on verrokkitestin tulos ja toiset uudelleenklusteroinnilla saadut tulokset. X-akselilla on haluttujen anomalioiden määrät. Uudelleenkluste-

rointi onnistui vähentämään vääriä hälytyksiä suhteessa havaittuihin oikeisiin verkkohyökkäyksiin.

Vaikka havaittujen verkkohyökkäysten määrä pieneni haluttujen anomaloiden määrän mukana, enemmän kuitenkin väheni väärien hälytysten määrä. Uudelleenklusterointimenetelmä siis paransi oikeiden verkkohyökkäysten havaittavuutta.

JYVSECTEC-materiaalilla testattiin uudelleenklusterointimenetelmän sopeutumista muuttuviin liikennemääriin. JYVSECTEC-materiaali oli jaettu useisiin eri verkkoliikennemäärät sisältäviin tiedostoihin, joita yhdistelemällä saattiin halutut liikennemäärät aikaiseksi.

Kuviossa 8 näkyy uudelleenklusteroinnin vaikutus klustereiden ja anomalioiden määrään verkkoliikenteen kasvaessa kaksinkertaiseksi. Kuvion 8 kuvaajien arvot löytyvät taulukosta 4.



KUVIO 9. Uudelleenklusteroinnin vaikutus anomalioiden ja klustereiden määrään verkkoliikenteen kasvaessa.

Kuvion 8 vasemmanpuoleisen Y-akselilla on havaittujen anomalioiden määrä ja oikeanpuoleisessa klustereiden määrä. X-akselilla on tarvittujen klusterointikertojen määrä, jotta haluttu anomaliamäärä saavutettiin. Tässä testissä halutun anomaliamäärän

alarajaksi oli asetettu 50 ja ylärajaksi 100. Kohdassa 0 ei ole vielä suoritettu uudelleenklusterointia ja kohdassa 14 uudelleenklusterointi on valmis.

Kuviosta näkee klustereiden määrän suhteessa havaittujen anomalioiden määrään. Anomalioiden määrän kasvu ja klustereiden määrän väheneminen heti uudelleenklusteroinnin alussa toistui kaikissa JYVSECTEC-materiaalilla tehdyissä testeissä. Verkkoliikenteen taas vähentyessä, anomalioiden määrä ensin pieneni ja klustereiden määrä kasvoi. Tämä ilmiö esiintyi voimakkaimmin aina ensimmäisellä uudelleenklusteroinnilla ja verkkoliikenteen määrän suurella äkillisellä muuttumisella. Ilmiöön voi löytyä selitys JYVSECTEC-materiaaliasta tai käytetystä klusterointimenetelmästä. JYVSECTEC-materiaalissa eri liikennemäärät sisältävien tiedostojen verkkoliikenne ei ole yhdenmukaista. Esimerkiksi verkkoliikenne nopeudella 200kbps sisältää myös erilaista liikennettä, mitä nopeudella 100kbps oleva verkkoliikenne on. Verkkoliikenteen muuttuessa erilaiseksi, liikenne on IDS-järjestelmälle uutta, jolloin se tunnistaa sen anomaliseksi verkkoliikenteeksi. Tästä johtuu anomaliamäärän kasvu. Muutaman klusterointi-iteraation jälkeen uudelleenklusterointimenetelmä on kouluttanut IDS-järjestelmän uudelle liikenteelle. Uudelleenklusterointimenetelmä oppii iteraatioden käsittelemään juuri niitä oikeita klustereita, joilla on paras vaikutus havaittavien anomaloiden määrään. Tämä oppiminen näkyy taulukossa 4 rivillä 6, eli kuudennen klusterointi-iteraation kohdalla. Kyseisellä kohdalla klustereiden määrä väheni yhdellä, mutta anomalioiden määrä väheni silti melkein 500:lla.

TAULUKKO 5. Uudelleenklusteroinnin vaikutus anomalioiden ja klustereiden määrään verkkoliikenteen kasvaessa.

Klusterointi-iteraatio	Klustereiden määrä	Havaitut anomaliat
0	137	2574
1	119	2750
2	106	2970
3	101	3043
4	97	3045
5	139	2927
6	150	2629
7	149	2134
8	200	1631
9	231	1109
10	240	804
11	241	499
12	253	301
13	258	174
14	260	94

JYVSECTEC-materiaalilla tehdyissä testeissä ilmeni myös ongelmia uudelleenklusterointimenetelmässä. Haluttujen anomaloiden määrän ollessa säädettynä alhaiselle tasolle, uudelleenklusterointi ei toimi oikein verkkoliikenteen äkillisesti kasvaessa tai vähentyessä. Uudelleenklusteroinnin alussa oleva anomaliamäärän laskeminen tai havaittujen anomalioiden puuttuminen voi tukahduttaa uudelleenklusterointimenetelmän. Jos havaittujen anomaloiden määrä tippuu lähelle 0:llaa, uudelleenklusterointiin ei ole riittävästi liikennettä saatavilla. Uudelleenklusterointi on tällöin suoritettava aikaisemmalla verkkoliikenteellä, josta anomalioita vielä havaittiin. Tämän liikenteen saatavuus voi olla ongelmallista.

Ostinato:lla luodulla materiaalilla tehdyt testit tehtiin uudelleenklusterointimenetelmän kehitysvaiheessa. Testeistä saaduilla tuloksilla ohjattiin uudelleenklusterointimenetelmän kehitystä samanlaisin testei mitä JYVSECTEC-materiaalin kanssa käytettiin. Materiaali sopi oivallisesti kehitysvaiheen testeihin, koska jokaisen eri vedoksen anomaliamäärä tiedettiin etukäteen.

6. PÄÄTELMÄT

Anomaliapohjainen IDS–järjestelmä on mielenkiintoinen tutkimuksen aihe sen kyvystä havaita ennestään tuntemattomia hyökkäyksiä. Anomaliapohjaisiin IDS-järjestelmiin liittyy monia haasteita, jotka ovat haitanneet menetelmän levinneisyyttä. Näihin kuuluu esimerkiksi korkea väärien positiivisten määrä, ja luotettavan koulutusmateriaalin tarve.

Tässä tutkimuksessa onnistuttiin ratkaisemaan anomaliapohjaisen IDS–järjestelmän ongelmaa suuren väärien positiivisten havaintojen osalta. Toteutettu uudelleenklusterointimenetelmä vähensi väärien positiivisten havaintojen määrää suhteessa oikeisiin positiivisiin. Säädettäessä haluttujen anomalioiden määrä halutuksi, osa hyökkäyksistä jää kuitenkin havaitsematta. Tällöin havaittujen hyökkäysten määrä laskee suhteessa kaikkiin verkkoliikenteessä oleviin hyökkäyksiin nähden. Jos IDS-järjestelmä luokittelisi kaiken verkkoliikenteen anomaliseksi, tunnistettaisiin verkkohyökkäyksistä 100 %. Tämä ei kuitenkaan ole reaalimailmassa hyödyllistä, sillä se ei edistä yhtään oikeiden verkkohyökkäysten löytämistä. Tuloksista käy kuitenkin ilmi uudelleenklusteroinnin positiivinen vaikutus väärien ja oikeiden positiivisten suhteeseen.

Uudelleenklusterointimentelmässä esiintyi kuitenkin myös ongelmia. Mentelmän kyky sopeutua muuttuneisiin liikennemääriin ei ole paras mahdollinen. Esimerkiksi liikenteen nopeasti tippuessa ja havaittujen anomalioiden määrän lähestyessä nollaa, uudelleenklusterointimentelmä tukahduttaa itsensä uudelleenklusterointiin tarvitta-

van materiaalin puuttuessa. Jos kyseistä materiaalia ei ole saatavilla, mentelmän olisi nollattava itsensä ja IDS:n olisi aloitettava verkkoliikenteeseen kouluttautuminen alusta. Ongelma voi johtua käytössä olevasta klusterointimentelmästä, jossa klustereiden muodostamiseen vaikuttaa liikaa datan, eli pakettien, saapumisjärjestys. Tästä johtuen klustereista saattaa muodostua hyvinkin erilaisia datan saapumisjärjestyksen muuttuessa. Tämä voi aiheuttaa merkittävää vaihtelua havainnointikyvyssä. Ongelma voisi poistua vaihtamalla klusterointimentelmä kehittyneempään, sellaiseen klusterointi algoritmiin, joka muodostaa klusterit aina täsmälleen samanlaisiksi riippumatta liikenteen järjestyksestä.

Testauksen aikana havaittiin että uudelleenklusteroinnin tarkkuus parantui klusterointi-iteraatioden myötä huomattavasti. Mitä pidemmälle klusterointia iteroitiin, sitä tarkemmin löydettyjen anomaloiden määrä alkoi pysyä tavoitellulla välillä. Ikäänkuin mentelmä oppi tunnistamaan juuri ne oikeat klusterit, joiden käsittelyllä on paras vaikutus anomalioiden tunnistamiseen.

Uudelleenklusteroinnin toteutuksen myötä huomattiin, kuinka IDS-järjestelmä ei enää välttämättä tarvitsekkaan opetusmateriaalia vaan uudelleenklusteroinnin kautta järjestelmä kouluttautuu verkkoliikenteeseen jatkuvasti. Tämä tuli esille erityisesti muuttuvien verkkoliikennemäärien testeissä. Periaatteessa järjestelmän voi laittaa suoraan tarkkailemaan verkkoliikennettä ilman kouluttamista. Tällöin lähtötilanteessa järjestelmän käsittelemä verkkoliikenteen määrä on periaatteessa 0, josta se kasvaa heti verkossa olevan liikennemäärän tasolle. Tässä tapauksessa alussa kaikki liikenne määritellään anomaliseksi jolloin järjestelmä kouluttauttuu uudelleenklusteroinnin kautta siten, että vain haluttu määrä luokitellaan anomaliseksi.

LÄHTEET

DARPA 1999a. Defence Advanced Research Projects Agency Intrusion Detection Data Sets. Viitattu 17.10.2012. http://www.ll.mit.edu/mission/communications/ist/corpora/ideval/data/index.html

DARPA 1999b Defence Advanced Research Project Agency Intrusion Detection Data. Off-line Simulation Network. Viitattu 26.10.2012. http://www.ll.mit.edu/mission/communications/ist/files/Network_Topology.gif

Falliere, N., Murchu, L. & Chien, E. 2011. W32.Stuxnet Dossier. Helmikuu 2011. Symantec. Viitattu 26.7.2012. http://www.symantec.com/content/en/us/enterprise/media/security_response/whitepapers/w32_stuxnet_dossier.pdf

Internet Protocol, IPv4. 2012. InetDaemon. Viitattu 11.7.2012. http://www.inetdaemon.com/tutorials/internet/ip/index.shtml

Intrusion Detection FAQ: What is Intrusion Detection. SANS. Viitattu 16.8.2012. http://www.sans.org/security-resources/idfaq/what_is_id.php

JYVSECTEC. Jyväskylä Security Technology. 2012. Viitattu 26.10.2012. http://jyvsectec.fi/

Mahoney M. 2001. Network Anomaly Intrusion Detection Research at Florida Tech. Viitattu 2.1.2013. http://cs.fit.edu/~mOmahoney/dist/

Ostinato. Packet/Traffic Generator and Analyzer. Viitattu 17.12.2012. http://code.google.com/p/ostinato/

Paananen, H. 2010. Feature Selection in Anomaly-based Network Intrusion Detection Systems. Jyväskylän Yliopisto. Viitattu 19.12.2012.

Portnoy L., Eskin E. & Stolfo S. 2001. Intrusion Detection with Unlabeled Data Using Clustering. Viitattu 4.10.2012. http://cseweb.ucsd.edu/~eeskin/papers/cluster-ccsdmsa01.pdf

RFC Sourcebook 2012a. IP, Internet Protocol. Viitattu 12.7.2012. http://www.networksorcery.com/enp/protocol/ip.htm

RFC Sourcebook 2012b. TCP, Transmission Control Protocol. Viitattu 4.7.2012. http://www.networksorcery.com/enp/protocol/tcp.htm

RFC Sourcebook 2012c. UDP, User Datagram Protocol. Viitattu 4.7.2012. http://www.networksorcery.com/enp/protocol/udp.htm

SANS 2011. 20 Critical Security Controls. SANS. Lokakuu 2011. Viitattu 16.8.2012. http://www.sans.org/critical-security-controls/cag3_1.pdf

Sophos HIPS. Protecting Against Zero-day Threats. Viitattu 2.1.2013. http://www.sophos.com/en-us/why-sophos/innovative-technology/hips.aspx

Thottan, M. & Ji, C. 2003. Anomaly Detection in IP Networks. IIIE Transactions on Signal Processing, vol 51, no. 8. Viitattu 20.12.2012. http://users.ece.gatech.edu/jic/sig03.pdf

Transmission Control Protocol, TCP. 2013. InetDaemon. Viitattu 11.7.2012. http://www.networksorcery.com/enp/protocol/tcp.htm

User Datagram Protocol (UDP). 2012. InetDaemon. Viitattu 11.7.2012. http://www.inetdaemon.com/tutorials/internet/udp/index.shtml