HONEYNET

LAHDEN AMMATTIKORKEAKOULU
Tietotekniikan koulutusohjelma
Tietoliikennetekniikka
Opinnäytetyö
Kevät 2008
Tommi Sällinen

Lahden ammattikorkeakoulu
Tietotekniikan koulutusohjelma

SÄLLINEN TOMMI: Honeynet

Tietoliikennetekniikan opinnäytetyö, 62 sivua, 8 liitesivua

Kevät 2008

TIIVISTELMÄ

Internet on käymässä vaaralliseksi paikaksi. Monenlaiset tietoturvaloukkaukset ja tunkeutumisyritykset uhkaavat sekä tavallisten kotikäyttäjien tietokoneita että yritysmaailman laitteita ja lähiverkkoja, ja tavoitteena on niiden kaappaaminen rikollisten hallintaan. Tietotekniikkaan liittyvä rikollisuus on muuttunut ammattimaiseksi, ja sen tavoitteeksi on noussut puhtaan taloudellisen hyödyn saaminen. Tietokoneiden turvaohjelmien loppukäyttäjien on hankala varmistaa tietoturvan toteutuminen siitäkin huolimatta, että järjestelmän päivitykset olisi suoritettu ja että käytettäisiin tehokkaita suojausohjelmia. Lisäksi hyökkäävä puoli on aina aloitteentekijä tietoturvaloukkauksissa.

Tässä opinnäytetyössä tutkitaan the Honeynet Projectin tekniikkaa, joka mahdollistaa aktiivisen tavan seurata verkon tapahtumia ja vastata tietoturvauhkiin. The Honeynet Project on internetyhteisö, joka on kehittänyt ideaa tuotantoympäristön ulkopuolelle sijoitettavasta suojaamattomasta tietoverkosta, joka tarjoaa ulkopuolisille aitoa ympäristöä vastaavan hyökkäyskohteen. Kyseinen tietoverkko on vahvan tarkkailun alla, ja hyökkäyksissä käytetyt menetelmät saadaan paljastettua yhteisön kehittämien välineiden avulla. Yhteisön verkkosivuilla on myös osio, jossa aikaisemmin tallennettua hyökkäysdataa puretaan ja lisätään tällä tavalla yleistä tietoutta erilaisista uhkista ja tavoista, joilla niitä vastaan voidaan varustautua.

Työssä on vertailtavana kaksi erilaista verkkomallia, jotka on rakennettu edellä mainitun idean pohjalta. Kummankin verkon toiminnan tarkoitus on kaiken liikenteen tallentaminen ja tapahtumien erittelyn mahdollistaminen jälkikäteen. Toinen verkoista edustaa alkuperäistä honeynet-mallia, jossa käytetyt laitteet ja apuohjelmat ovat toteutettavissa miltei minkä tahansa Linux-jakelun mukana tulevilla ohjelmilla. Toinen verkko käyttää tätä tarkoitusta varten suunniteltua ohjelmistoa, joka muuttaa sopivilla laitteilla varustetun PC-tietokoneen verkon kontrolli- ja analysointipisteeksi. Työn kuluessa kummatkin verkot ja niihin kuuluvat laitteet rakennetaan alusta asti toimiviksi kokonaisuuksiksi sekä suoritetaan testijaksot, joiden aikana liikkunut data tallennetaan kokonaisuudessaan. Datan analysoinnin tulokset esitetään melko pintapuolisesti, koska muuten aihe laajenisi paljon tämän opinnäytetyön ulkopuolelle. Työn aikana käy ilmi, että kumpaakin honeynet-mallia voidaan hyvin käyttää ulkopuolelta tulevan liikenteen kartoittamiseen, mutta uudempi malleista tarjoaa tähän helpomman ja tietoturvallisemman tavan.

Avainsanoja: honeynet, honeypot, tietoturva

Lahti University of Applied Sciences
Faculty of Technology

SÄLLINEN TOMMI: Honeynet

Bachelor's Thesis in Telecommunications Technology, 62 pages, 8 pages of appendices

Spring 2008

ABSTRACT

The Internet has become a dangerous place. There are many kinds of security threats which make connecting to the Internet unsafe. Criminals in the field of information technology act like any other professionals whose goal is to maximize economical benefit. Their product is access to compromised computers and they hire the possibility to use that computing power for criminal purposes. Despite the fact that there are numerous security programs which can be used in order to protect computers against attackers' attempts to exploit them, it might be difficult to determine the level of information security. Moreover, the defending mechanisms are always quite passive, leaving the initiative to the attacker.

The aim of this thesis is to study Honeynet, a technique which makes it possible to actively monitor the acts of the attacker and learn his methods and motives. This technique was invented by an internet community called the Honeynet Project. The main idea is to build a network which is left unsecured, except for some traffic restrictions, and to monitor the network traffic thoroughly. The network operates like a real one and it is impossible for the attacker to notice being under monitoring. The community has developed many powerful tools which make the acts of the attacker visible to the monitor, even when communication is encrypted. The Honeynet Project has also gathered detailed information about attacks and they share that information in the internet site of the project.

In the case section of the thesis, two different honeynets were built and configured. The first honeynet has the original honeynet structure and it includes only parts available in almost every freely downloadable Linux distribution. The second honeynet structure includes an extra device, which is programmed especially for that purpose. All parts are still open source and freely available. In addition, the honeynets include victim computers called honeypots running common operating systems. Both honeynets went through testing sequences and during that period all communication data was captured and analyzed. Experience from the testing period and obtained results showed that both honeynet models are capable of capturing data accurately, but generation II honeynet is more secure and offers a more flexible way to monitor and analyze data.

Keywords: honeynet, honeypot, information security

SISÄLLYSLUETTELO

LYHENTEET

DMZ	Demilitarized Zone, neutraali alue internetin ja suojatun verkon välillä
DNS	Domain Name System, järjestelmä, joka muuttaa sivunimet IP-osoitteiksi
DSL	Damn Small Linux, pienikokoinen Linux-jakelu
GUI	Graphical User Interface, graafinen käyttöliittymä
ICMP	Internet Control Message Protocol, protokolla virheilmoitusten ja kontrolliviestien lähettämiseen
IDS	Intrusion Detection System, tunkeutumisen havaitsemisjärjestelmä
IETF	Internet Engineering Task Force, työryhmä, joka ohjaa internetin kehitystä
IPSec	Internet Protocol Security, protokollaperhe tietoliikenteen suojaamiseen
IRC	Internet Relay Chat, tekstipohjainen monen käyttäjän keskustelujärjestelmä
MAC	Media Access Control, verkkosovittimen ethernet-verkossa yksilöivä osoite
NIC	Network Interface Card, verkkoliityntäkortti
NSF	National Science Foundation, Yhdysvaltain kansallinen tiedesäätiö
OSI	Open Systems Interconnect, avoin standardi laitteiden kytkemiseksi toisiinsa
SA	Security Association, kahden laitteen välinen sopimus käytetystä tietoturvamenettelytavasta
SSH	Secure Socket Layer, turvallinen etäkäyttöohjelma (asiakas)
SSHD	Secure Socket Layer Daemon, turvallinen etäkäyttöohjelma (palvelin)
TCP	Transmission Control Protocol, yhteydellinen kuljetusprotokolla
TTL	Time To Live, määrittelee IP-paketin elinajan hyppyinä verkossa
UDP	User Datagram Protocol, yhteydetön kuljetusprotokolla
VPN	Virtual Private Networking, virtuaalinen lähiverkko

1 JOHDANTO

Opintojen alkuvaiheessa tietoturvaan liittyvät asiat kiinnostivat minua selvästi vähemmän kuin monet muut tietotekniikan alalla vastaan tullet aiheet. Kiinnostavampaa oli opiskella teoriaa, joka vaikuttaa kaikkien asioiden taustalla ja rakentaa niiden tietojen perusteella käytännössä erilaisia toimivia kokonaisuuksia. Tilanne muuttui, kun aloin kiinnittää huomiota omassa verkossa tapaamaani satunnaiseen liikenteeseen, joka ei ollut lähtöisin omasta toiminnastani. Myöhemmin heräsi kiinnostus tietää, mitä kaikkea omassa verkossa tapahtuu ja ottaa tilanteen hallinta kokonaan omiin käsiin. Oma verkko on yksityisaluetta, jonne ulkopuolisilla ei pitäisi olla asiaa ilman lupaa.

Kun löysin the Honeynet Projectin kehittämään tarkkailuverkon idean ja tutustuin asiaan tarkemmin, huomasin, että se liittyy tarkalleen edellä mainittuun ilmiöön ja tarjosi myös kiinnostavan aiheen opinnäytetyön tekemiseksi. Opinnäytetyössä yhdistyy tietoverkon turvallisuuden kartoittamisen erittäin käytännönläheinen ongelma melko suureen määrään tutkimustyötä ongelman selvittämiseksi ja ratkaisemiseksi.

Opinnäytetyön alussa esitetään katsaus internetin historiaan ja kuvataan erilaisia yleisimpiä tietoturvauhkia, jotka ovat tällä hetkellä ajankohtaisia. Opinnäytetyön varsinainen aihe on the Honeynet Projectin kehittämä tietoverkkoratkaisu, jonka avulla voidaan aidossa mutta samalla mahdollisimman turvallisessa ympäristössä tutkia kaikenlaisia tietoturvaloukkauksia ja hyökkääjien käyttämiä menetelmiä. Aihetta käsitellään ensin teoriassa, minkä jälkeen käytännön osassa rakennetaan testejä varten kaksi erilaista verkkomallia. Verkkojen rakentamisen vaiheet kuvataan käytännön osassa ja liitteissä. Testien tuloksena tallennettua dataa analysoidaan eri honeynet-malleissa käytettävissä olevien analysointikeinojen selvittämiseksi ja niiden välisten eroavaisuuksien löytämiseksi. Tapahtumien tarkka analysointi jää pois, koska se on hyvin laaja aihe ja ylittää tämän opinnäytetyön rajauksen.

Opinnäytetyön tavoite on tutkia, kuinka kattava kuva testiverkossa kulkevasta liikenteestä voidaan saada ja eroavatko erilaisten verkkototeutusten antamat tulokset toisistaan. Työhön kuuluu kahden erilaisen verkkototeutuksen rakentaminen ja kumpaankin sisältyy paljon erilaisia asennuksia ja konfigurointeja. Toimivan lopputuloksen aikaansaaminen on yksi opinnäytetyön tavoitteista. Liikenteen analysointi ulotetaan vain tasolle, jolla voidaan erotella merkittävät yhteydet muun liikenteen joukosta. Honeynet-toteutuksien käyttämät keinot tämän aikaansaamiseksi ovat erilaisia, ja niitä vertaamalla yritetään saada selville eri versioiden ominaisuudet ja kelpoisuus analysoinnin suorittamiseen. Erilaisuuksia toteutusten välillä on tarkoitus saada esiin myös vertaamalla käytön helppoutta, saatujen tulosten selkeyttä sekä tietoturvan tasoa. Työssä käytetään kolmea erilaista käyttöjärjestelmää kohteina hyökkääjien operaatioille, ja näiden vaikutusta liikenteen muodostumiseen vertaillaan.

2 TIETOTURVA INTERNETISSÄ

2.1 Internetin historia

Internetin historia voidaan ajoittaa alkavaksi 1960-luvun alkupuolelta, jolloin Yhdysvaltojen ilmavoimat halusi kehittää massiivisesta ydiniskusta selviävän tietoliikenneverkon silloisen keskuskoneisiin pohjautuvan järjestelmän korvaajaksi. Vanha järjestelmä oli erittäin haavoittuvainen, koska käyttäjät olivat omilla päätteillään yhteydessä vain keskuskoneeseen ilman suoraa yhteyttä toisiinsa. Keskuskoneen vikaantuminen tai tuhoutuminen olisi katkaissut tietoliikenteen kokonaan. Huolenaiheena oli koko verkon fyysinen ja toiminnallinen säilyvyys, kyky toipua sekä keskeytymätön toiminta. Verkon tietoturva, siinä mielessä kuin se nykyään ymmärretään, ei vielä tuolloin ollut huolenaihe, vaikka perinteisessä mielessä eri tarkoitusta varten kerättyjen tietojen säilyttäminen ja turvaaminen olikin ilman muuta tärkeää. (Hakkerin käsikirja 2002, 778.)

Tästä ideasta alkunsa saanut tietoliikenneverkko toteutui vuonna 1969. Järjestelmää kutsuttiin ARPANETiksi ja siihen kuului aluksi vain neljä tietokonetta neljästä yliopistollisesta laitoksesta USA:ssa. Tietokoneiden käyttöjärjestelmät eivät kuitenkaan vielä tuohon aikaan osanneet käyttää hyväkseen uudentyyppistä verkkoa. Vasta UNIX-käyttöjärjestelmän kehittyminen 1970-luvun puoleen väliin mennessä ratkaisi tämän ongelman ja antoi käyttöön uusia ominaisuuksia, joiden avulla hajautettua verkkoa voitiin vasta toden teolla alkaa hyödyntämään. Näitä uusia ominaisuuksia olivat mm. sähköposti, joka otettiin käyttöön 1972 sekä vuonna 1974 toimivaksi saatu TCP (Transmission Control Protocol), joka on yksi tärkeimmistä internetin tietoliikenteessä nykyään käytettävistä protokollista. (Hakkerin käsikirja 2002, 780.)

Vuonna 1975 ARPANETin katsottiin tulleen valmiiksi alkuperäistä käyttötarkoitustaan varten ja sen hallinta siirrettiin valtion organisaatiolle, jonka nimi oli tuolloin United States Defence Communications Agency. Kun vuonna 1972 tähän ensimmäiseen internetiin oli kuulunut vain n. 40 työasemaa, kasvoi koneiden määrä vuosien 1974 ja 1980 välillä monikymmenkertaiseksi. Syynä tähän oli

UNIX-käyttöjärjestelmän käyttöönottaminen kaikissa USA:n yliopistoissa. Hyvin nopeasti tämän vaikutukset alkoivat tuntua myös kaupallisessa maailmassa, mikä lisäsi UNIXin leviämistä. (Hakkerin käsikirja 2002, 781 – 782.)

UNIXin käytön laajentuminen johtui myös sen koodin avoimuudesta. Lähdekoodi oli vapaasti saatavilla ja sen muokkaaminen erilaisille alustoille ja ympäristöille oli täysin sallittua. Tämä oli kaksiteräinen miekka, koska sen ansiosta myös järjestelmän turvarakenteiden vikoihin pystyttiin perehtymään yksityiskohtaisesti. Toisin kuin kaupallisten ja suljettujen ohjelmistojen tapauksessa, joissa järjestelmän haavoittuvuudet joudutaan etsimään kokeilemalla, saattoi UNIXin lähdekoodia käyttää suoraan apuna haavoittuvuuksien etsimisessä. Tosin ennen 1980-luvun alkua tällaisesta toiminnasta ei vielä juurikaan tiedetty. (Hakkerin käsikirja 2002, 782 – 783.)

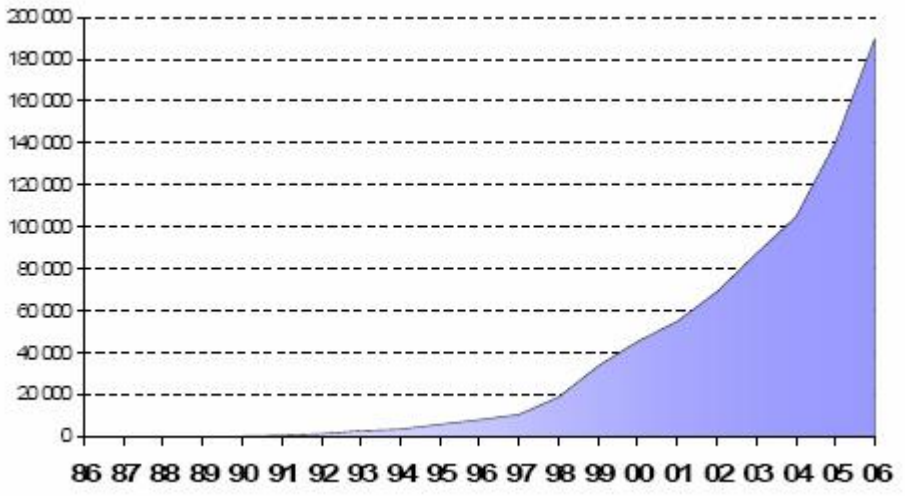
Alkuperäisen verkon kehittämisen tarkoituksena oli nimenomaan varmistaa tiedon liikkuminen ja perille meneminen. Alkuaikoina ei verkossa edes ollut sellaisia tietoja, joiden salaaminen verkon muilta käyttäjiltä olisi ollut tarpeellista. Vähitellen verkon laajentuessa ja siihen liittyessä uusia käyttäjäryhmiä tuli tietojen salaaminen ja käyttäjien autentikointi yms. ajankohtaiseksi. UNIXin turvaominaisuuksia on sittemmin paranneltu tai siihen on liitetty kokonaan uusia ominaisuuksia, jotka vastaavat varsinkin laajan verkottumisen haasteisiin. Näitä ovat mm. salakirjoitetut salasanat, vahva tiedostojen ja käyttöoikeuksien hallinta, järjestelmätason autentikointimenetelmät ja kehittyneet lokijärjestelmät. UNIXiin tuli myös saataville paljon ulkopuolisia ohjelmia, joilla tietoturvaa pystyi parantamaan edelleen ja muokkaamaan sitä erilaisiin tilanteisiin tarkalleen sopivaksi. Näistä ohjelmistoista monet ovat sellaisia, jotka soveltuvat hyökkäävän osapuolen käyttöön aivan yhtä hyvin kuin järjestelmän tietoturvaa parantavalle ylläpitäjälle. Esimerkiksi turvatarkastustyökalut, joilla voidaan automaattisesti etsiä minkä tahansa verkon toiminnassa olevat ja verkkoon liitetyt tietokoneet ja niiden saapuvia yhteyksiä kuuntelevat sovellukset, ovat erittäin hyödyllisiä sekä laillisille että laittomille käyttäjille. Uusista ominaisuuksista johtuvalla järjestelmän monimutkaistumisella saattaa myös olla negatiivinen vaikutus tietoturvaan. (Hakkerin käsikirja 2002, 787 – 788.)

1990-luvulle tultaessa internet oli lähes kokonaan sotilas- ja yliopistohenkilökunnan käytössä ja käyttö kaupallisiin tarkoituksiin oli ehdottomasti kielletty. Käyttäjiä oli arviolta muutamia satojatuhansia. Verkon hallinnasta vastasi Yhdysvaltain kansallinen tiedesäätiö (NSF). Vuonna 1991 NSF kuitenkin luopui määräysvallastaan runkoverkon ylläpidosta aiheutuneiden kustannusten takia ja vetäytyi kokonaan verkonvalvonnasta vuonna 1995. Tämän jälkeen internet kaupallistui nopeasti. Kun samoihin aikoihin oli julkaistu verkon käyttöä helpottavia graafisia sovelluksia, oli internetin käyttö avoinna myös tavallisille ihmisille. Sovelluksista voidaan mainita Gopher, joka tekstipohjaisesta käyttöliittymästään huolimatta helpotti verkossa navigointia sekä WWW-selaimet monelta eri valmistajalta. Vaikka tietokoneet olivat tuolloin vielä todella kalliita, oli halukkailla ainakin periaatteessa mahdollisuus koneen hankkimiseen ja sen kytkemiseen verkkoon. (Hakkerin käsikirja 2002, 788 – 789.)

Noin vuodesta 1995 lähtien internetin kanssa tekemisissä olevien ihmisten määrä alkoi kasvaa räjähdysmäisesti verkkoon liitettyjen tietokoneiden määrän mukana. Nykyään on vaikea kuvitella sellaista alaa, jolla tietokoneita ei käytettäisi, ja useimmilla ihmisillä ainakin teollistuneissa maissa on käytössään henkilökohtaisia tietokoneita lähes jokaisella elämän alueella. Verkon käyttäjiä tulee koko ajan lisää ja uusia sovelluksia kehitetään, mutta edelleen on käytössä alkuperäisiä liikennöintiprotokollia ja tietoturvakäytäntöjä, jotka eivät ole tämän päivän vaatimusten mukaisia. Tämä luo erinomaisen tilaisuuden henkilöille, jotka osaavat ja haluavat käyttää muita ihmisiä hyväkseen verkon kautta.

2.2 Uhkakuvia

Ensimmäiset haittaohjelmat ilmestyivät vasta 1980-luvun puolen välin jälkeen. Leviäminen oli melko hidasta ohjelmien hitaan tarttumismekanismin takia. Morris-madon nimellä historiaan päässyt haittaohjelma onnistui kuitenkin tartuttamaan suurimman osan silloiseen internetiin liitetyistä koneista. Tämän tapauksen jälkeen tietokoneiden suojaamiseen alettiin kiinnittää enemmän huomiota. Kuviosta 1 ilmenee haittaohjelmien määrän lisääntyminen 1980-luvun lopusta lähtien vuoden 2006 loppuun asti.



KUVIO 1. Haittaohjelmien määrän kehitys (Mustonen 2006.)

Tietoturvaan kohdistuvat uhkat voidaan jaotella monella tavalla. Yksi tapa on jakaa ne uhkiin, jotka:

1. Kohdistuvat tiedostojen sisältöön tai palveluihin. Tämän tyypin hyökkäykset ovat kohdennettuja tiettyyn tietokoneeseen tai verkkoon.

2. Etsivät haavoittuvuuksia sisältäviä laitteita. Hyökkäys on kohdistettu satunnaiseen kohteeseen ja tavoitteena on saada hyökkääjän hallintaan resursseja, joilla voidaan myöhemmin suorittaa laajamittaisia hyökkäyksiä tarkasti valittuihin kohteisiin.

3. Kohdistuvat verkkopalveluita käyttäviin asiakkaisiin. (Kajantie 2006.)

Ensimmäisen tyypin uhka kohdistuu tunnettuja yrityksiä ja erilaisia yhteiskunnan tarjoamia tietotekniikkaan ja verkottumiseen perustuvia palveluita kohtaan. Yhtä hyvin hyökkäyksien kohteena voisi olla yksityishenkilöt, mutta silloin hyödyn saaminen olisi epätodennäköisempää, eikä mahdollinen palvelun estäminen vaikuttaisi tarpeeksi laajaan käyttäjäkuntaan. Tällaisia hyökkäyksiä on tapahtunut

lähiaikoina paljon, ja monien hämmästyttävänä piirteenä on ollut nopeus, jolla hyökkäävä osapuoli on reagoinut hyökkäyksen syyltä vaikuttaneeseen tapahtumaan. Melko äärimmäisenä esimerkkinä toimivat patsaskiistan tapahtumat Virossa huhtikuun lopussa 2007, jolloin mm. virolaisviranomaisten verkkosivut joutuivat palvelunestohyökkäysten kohteiksi ja olivat tavoittamattomissa useiden päivien ajan. (Tietoviikko 2007.)

Toisen tyypin uhka kohdistuu käyttöjärjestelmien ja ohjelmistojen tunnettuihin tai toistaiseksi julkaisemattomiin haavoittuvuuksiin, joiden hyödyntäminen on mahdollista lähettämällä tietyllä tavalla muotoiltua liikennettä palvelua tarjoavaan tietokoneeseen. Hyökkääjä pääsee ajamaan luvattomasti omia komentojaan vieraassa järjestelmässä ja voi saada siihen täyden hallinnan, riippuen palvelun tai käyttäjän senhetkisistä käyttöoikeuksista. Täysi hallinta tarkoittaa käytännössä sitä, että hyökkääjä on saanut käyttöönsä järjestelmän sisäänpääsyyn oikeuttavan käyttäjätunnuksen ja salasanan ja voi näin ollen tehdä järjestelmässä mitä haluaa. (Kajantie 2006.)

Käyttöjärjestelmien ja ohjelmistojen valmistajat julkaisevat melko nopeasti päivityksiä tuotteisiinsa uusien julkitulleiden haavoittuvuuksien korjaamiseksi ja asiakkaidensa tietojärjestelmiensä suojaamiseksi luvattomalta käytöltä. Haavoittuvuuden korjaaminen vie aina oman aikansa ja tämän ajan järjestelmät ovat alttiina ns. nollapäivähyökkäyksille, mikä tarkoittaa sellaisen tunnetun haavoittuvuuden hyväksikäyttöä, johon ei ole vielä saatavilla virallista korjaavaa päivitystä. (Kajantie 2006.)

Kolmannen tyypin uhkat vaanivat internetsivuilla, joille on piilotettu hyökkäävää koodia. Pelkkä vierailu väärällä sivustolla riittää tietokoneen saastuttamiseksi. Tämä on erityisen vaarallista, koska kaikki käyttöjärjestelmät ja internetselaimet ovat jossain määrin haavoittuvaisia tälle uhalle. Käyttäjän tietokonetta suojaava palomuuri ei voi estää tällaista uhkaa, koska käyttäjä avaa itse yhteyden haittaohjelman sisältävälle sivulle ja mahdollistaa sen seurauksena haittaohjelmien pääsyn tietokoneelle. Tietokoneen saastumisen estämiseksi tarvitaan ajan tasalla oleva virusohjelmisto tai muita suojaohjelmia, jotka pystyvät havaitsemaan haitallisen koodin reaaliaikaisesti. (Kajantie 2006.)

2.3 Verkon käyttäjään kohdistuvia haittoja

2.3.1 Virukset

Virus on haittaohjelma, joka leviää kopioitumalla muihin ohjelmiin. Se ei pysty leviämään itsenäisesti, mutta kuljettajinaan se voi käyttää mm. erilaisia tiedostoja, sähköpostia tai WWW-sivuja. Lisäksi viruksen tarttuminen koneelle vaatii yleensä käyttäjän toimia, sähköpostin liitetiedoston avaamisen tai linkin klikkaamisen saastuneella WWW-sivulla. Virukset pystyvät muuttamaan ja tuhoamaan tietojärjestelmien sisältämiä tietoja sekä vaikuttamaan esim. hidastavasti koko järjestelmään. (Kerttula 1998.)

Virustorjuntaohjelmat tunnistavat virukset sen binäärikoodista otetun näytteen ns. sormenjäljen mukaan. Tietoturvayhtiöt pitävät yllä ja täydentävät tietokantaa, joka pitää sisällään kaikkien tunnistettujen virusten sormenjäljet. Virusten uudet muodot osaavat muuntaa omaa koodiaan siten, että virustorjuntaohjelmat eivät niitä enää tunnista ja käytännössä täysin sama virus vaatii tietoturvayhtiöiden tunnistusrutiinin läpikäymisen uudelleen. Tämä pidentää entisestään sitä aikaa, jolloin viruksilla on mahdollisuus aiheuttaa suurinta vahinkoa. (Helenius 2006.)

2.3.2 Verkkomadot

Madot eroavat viruksista siinä, että ne pystyvät leviämään itsenäisesti käyttämällä hyväkseen verkko-ohjelmistojen ja käyttöjärjestelmien tietoturva-aukkoja. Madoiksi luokitellut haittaohjelmat tarvitsevat ainoastaan toimivan internetyhteyden päästäkseen sisään sopivan haavoittuvuuden sisältävään järjestelmään. Madot kuluttavat vapaata tiedonkäsittelykapasiteettia, ja ne voivat vieraaseen järjestelmään päästyään alkaa levittämään muita haittaohjelmia tehden näin ongelman alkuperäisen lähteen löytämisen vaikeammaksi. (TSK 2004.)

Määritelmän mukaisesti puhtaat verkkomadot eivät tee muutoksia tunkeutuessaan luvattomasti järjestelmään. Ne kuitenkin voivat kuluttaa tiedonsiirtokapasiteettia ja saada aikaan muuta vahinkoa mm. palvelunestohyökkäyksien tai roskapostin

lähtöpisteenä. Erityisen vaarallisia ovat haittaohjelmat, jotka yhdistävät virusten ja matojen ominaisuuksia. (Järvinen 2006.)

2.3.3 Troijan hevoset

Troijan hevonen on hyödylliseksi luultu ohjelma, joka kuitenkin sisältää haitallista toiminnallisuutta. Haitallinen osuus on piilotettu niin, ettei varomaton käyttäjä voi huomata mitään hälyttävää tapahtuneen ohjelmaa asentaessaan. Asennus alkaa aina käyttäjän toimilla. Kun troijalainen on asettunut järjestelmään, voi tunkeutuja saada osittaisen tai täyden hallinnan järjestelmään ja voi käyttää sitä tarpeidensa mukaan hyväkseen. (TSK 2004.)

Aktivoiduttuaan vieraassa tietokoneessa troijalaiset pystyvät suorittamaan itsenäisesti tiettyjä toimia, jotka heikentävät tietoturvaa oleellisesti. Tyypillisesti troijalainen sammuttaa koneessa olevan palomuuriohjelman tai avaa siitä salaa portin, joka avaa hyökkääjälle pääsyn järjestelmään. Useasti troijalaiset muuttavat myös koneeseen asennetun virusohjelman toimintaa vaikeuttaakseen hyökkääjän myöhempien toimien havaitsemista. (Järvinen 2006.)

2.3.4 Vakoiluohjelmat (Spyware)

Vakoiluohjelmat keräävät tietokoneelle päästyään yksityiskohtaista tietoa käyttäjistä ja heidän käyttäytymisestään verkossa. Tietoja voidaan lähettää ohjelman tekijälle tai mille tahansa kolmannelle osapuolelle. Vakoiluohjelmien asettuminen koneelle voi tapahtua monella eri tavalla. Monet ohjelmat tulevat internetistä tarttuneen troijalaisen salaa asentamina tai sivulta, jolla on ensin pyydetty käyttäjää asentamaan esim. ActiveX-komponentti sivulla tarjottavaa toimintoa varten. Vakoiluohjelmia asentuu myös silloin, kun asennetaan internetistä ladattu turvalliselta näyttävä ohjelma, joka kuitenkin on varustettuna vakoiluohjelmalla. (Panda Security 2008.)

Tähän ryhmään voidaan lukea kuuluvaksi muitakin haitakkeita, joita käytetään laajasti tietojen keräämiseksi ja siirtämiseksi jompaankumpaan suuntaan, joko

käyttäjälle tai tietojen kerääjille internetissä. Adware on mainosohjelma, joka useimmiten käyttäjän tahtomatta ponnahtaa näkyville mainostamaan jotakin tiettyä tuotetta tai palvelua. Ohjelma on asettunut koneeseen luvattomasti, se on saattanut muokata järjestelmän asetuksia salaa tai kuten useissa tapauksissa voinut vaihtaa internetselaimen kotisivun, jotta käyttäjä joutuisi näkemään tietyn mainossivun aina aloittaessaan internetissä surffaamisen. (Panda Security 2008.)

Cookiet ovat selaimen välimuistiin kerääntyviä tekstitiedostoja, joihin tallennetaan internet-sivuilla tarvittavia tietoja seuraavia sivuvierailuja varten. Tarkoitus on tehdä toiminta kätevämmäksi, koska kaikkia tietoja ei ole tarpeellista kirjoittaa aina uudelleen. Esim. kirjautumista vaativat sivut saattavat lukea edellisen käyttäjän käyttäjätunnuksen suoraan selaimesta. Cookiet eivät itse ole tietoturvariski, mutta koska niillä voidaan kerätä monenlaista dataa käyttäjän täysin sitä tiedostamatta, ulkopuolisen on niiden avulla mahdollista saada haltuunsa tietoja, jotka loukkaavat yksilön yksityisyyttä. (Panda Security 2008.)

2.3.5 Alataso (Rootkit)

Rootkit ei itse ole varsinainen haittaohjelma, vaan erityinen tekniikka tai ohjelma, jolla haittaohjelman olemassaolo voidaan piilottaa käyttöjärjestelmätasolla suoritetulta etsinnältä. Rootkit on saanut nimensä UNIX-maailmasta, jossa se tarkoittaa käyttöjärjestelmään kuuluvan ohjelman vaihtamista samankaltaiseen ja samannimiseen ohjelmaan, joka piilottaa järjestelmän käyttäjältä osia ohjelman alkuperäisestä toiminnasta. Haittaohjelma piilotetaan käyttöjärjestelmän ytimessä ja sen vuoksi käyttöjärjestelmän tehtävienhallinnassa ei näy ohjelman käynnistämää prosessia eikä resurssienhallinta kykene näyttämään haittaohjelmaan kuuluvia tiedostoja. Rootkit voi piilottaa myös haittaohjelman muokkaamat rekisteriavaimet. (Gizmo 2006.)

Koska rootkit piilottaa käynnissä olevia prosesseja käyttöjärjestelmältä, voidaan sitä hyödyntää mm. tietovarkauksien toteuttamisessa. Rootkitin löytäminen järjestelmästä ei ole helppoa ja pahimmassa tapauksessa se saa rauhassa majailla tietokoneessa pitkiä aikoja käyttäjän tietämättä ja auttaa keräämään arvokkaita tietoja

sekä mahdollistaa tietojen lähettäminen eteenpäin mahdollisimman huomaamattomasti. (Gizmo 2006.)

2.3.6 Roskaposti (Spam)

Roskaposti on sähköpostiin perustuvaa mainontaa, johon on yhdistynyt mahdollisuus saada käyttäjän kone hallintaan liitetiedostoissa piilevien haittaohjelmien avulla tai houkuttelemalla hänet haitallisia ohjelmia sisältäville sivulle. Roskapostin hyödyllisyys lähettäjälle perustuu sen nopeuteen ja olemattomiin lähetyskustannuksiin. Postittajat lähettävät miljoonia viestejä kerrallaan, ja heille riittää se että häviävän pieni osa vastaanottajista reagoi viestiin. (Järvinen 2006.)

Roskaposti käsittää suurimman osan kaikesta lähetetystä sähköpostista. Sen määrän katsotaan olevan tutkimuksesta ja mittausajankohdasta riippuen 70 - 90 prosenttia kokonaisliikenteestä. Vaikka roskapostin lähettäminen on kriminalisoitu monissa maissa, on postittajien kiinnisaaminen edelleen hyvin vaikeata. Roskapostittaja pystyy pysymään tuntemattomana käyttäessään kaapattuja koneita lähetysalustanaan. (Commtouch 2007.)

Sähköpostiohjelmat kehittyvät, ja ne kykenevät seulomaan osan roskapostista pois hyödyllisen postin joukosta. Vastaavasti rikollisten keinot kehittyvät myös koko ajan ja uudella tavalla toteutettu roskaposti läpäisee hetken aikaa hyvänkin suodatuksen. Roskapostisuodattimien toiminta perustui aluksi pelkästään viestin sisältämän tekstin vertaamiseen avainsanalistan sanoihin. Viesteissä olevien avainsanojen hienoinen muokkaaminen esim. erikoismerkeillä riitti hämmentämään suodattimia päästämään roskapostia läpi, vaikka ihminen pystyi vielä vaivatta ymmärtämään viestin sisällön. Myöhemmin suodattimien tultua tehokkaammiksi roskapostittajat siirtyivät käyttämään kuva- ja äänisisältöä viestien välittämisessä. Viimeisin keksintö on piilottaa roskapostittajan mainoksen sisältämiä kuvia ylimääräisenä jonkin tunnetun tahon sivun sisällön joukkoon ja ohjata yhdistelmä vastaanottajan suodatuksen ohi. (Järvinen 2006.)

2.3.7 Tietojen kalastelu (Phishing)

Tietojen kalastelu on käyttäjään kohdistuvaa tietojen urkintaa, jolla hyökkääjä toivoo saavansa kerättyä uhreilta rahanarvoisia tai luottamuksellisia tietoja. Kalastelu toteutetaan mm. vakuuttavan näköisillä väärennetyillä sähköposteilla, jotka on naamioitu näyttämään jonkin luotetun yhtiön tai viranomaisen lähettämiltä. Viesteissä pyydetään usein käyttäjää antamaan henkilökohtaisia tietoja tai kehotetaan vierailemaan jollain tietyllä internetsivulla. Sivu saattaa sisältää esim. väärennetyn kirjautumisikkunan palveluun, jota käyttäjä tavallisestikin käyttää. Tiedot kuitenkin menevät rikollisille, joita he voivat käyttää taloudellisen hyödyn saamiseksi. (Kajava 2003.)

Erityisesti tällaista keinoa käytetään teollisuudessa tietojen urkkimiseen. Termi social engineering, joka voidaan suomentaa vakuuttavaksi käytökseksi, tarkoittaa ulkopuolisen esiintymistä esim. yhtiössä korkeassa asemassa olevana henkilönä tarkoituksenaan saada tärkeitä tietoja tai muuta hyötyä itselleen. Suuressa yrityksessä työntekijät eivät todennäköisesti tunne hyvin toisiaan, ja tämän keinon hyödyntäminen on aivan mahdollista. (Kajava 2003.)

Kalastelu on erilainen menetelmä muihin tietoturvan loukkauksiin verrattuna, koska siinä ei hakkeroida järjestelmää vaan järjestelmän käyttäjää. Tietojen kalastelun mahdollistaa internetin ”uutuus” ja siihen liittyvä tietynlainen lukutaidottomuus, jota rikolliset pystyvät yllättävillä keinoillaan käyttämään hyödyksi.

2.3.8 Huijausviestit (Hoax)

Hoaxeiksi kutsutaan perättömiä virustiedotteita, jotka leviävät internetissä usein sähköpostin välityksellä. Niiden tarkoituksena on aiheuttaa hämminkiä tavallisissa tietokoneen käyttäjissä, laittaa liikkeelle huhuja uusista virusuhkista tai olla vain nopeasti leviävä käytännön pila, jonka nykytekniikka on mahdollistanut. Kuitenkin hoaxin vaikutuksesta kokonaiset sähköpostijärjestelmät saattavat tukkeutua lähetysten määrästä monien käyttäjien lähettäessä viestiä eteenpäin. (Järvinen 2006.)

Sähköpostin välityksellä leviävä kiertokirje on nykyajan versio perinteisestä kirjekuoressa postin välityksellä kiertävästä ilmiöstä. Idea on aivan sama, mutta sähköpostin salamannopea jakelujärjestelmä kasvattaa ansaitsemismahdollisuudet aivan eri tasolle entiseen verrattuna.

2.3.9 Etähallittavat verkot (Botnet)

Botnet on joukko tietokoneita, jotka on otettu yhden pisteen hallintaan ja joihin on asennettu tästä yhdestä pisteestä kontrolloitu etähallintaohjelma. Koneita tällaisessa verkossa voi olla muutamista satoihin tuhansiin, ja yhdessä ne muodostavat merkittävän uhkan minkä tahansa internetissä tarjottavan verkkopalvelun ylläpidolle. Uhka muodostuu siitä, että kaikki bottiverkon koneet voidaan komentaa lähettämään yhtä aikaa yhteyspyyntöjä samaan osoitteeseen. Palvelua ylläpitävä tietokone ruuhkautuu yhteyksien tulvassa, ja on lopulta tavoittamattomissa laillisille käyttäjille. Koska minkään internetpalvelun ylläpitäjä ei voi varata päivittäisiin tarpeisiin verrattuna moninkertaista määrää resursseja vain tällaisen hyökkäyksen varalle, on tämäntyyppisen uhkan aiheuttama vaara hyvin suuri ja hyökkäyksen vastustaminen vaikeaa. (Panda Security 2008.)

Bottiverkkojen kehittyminen näkyy tavassa, jolla verkkojen hallinta on muuttunut. Monen etäkoneen yhtäaikainen hallinta yhdestä pisteestä on pulmallista siinä mielessä, että hyökkääjä ei voi paljastumisen vaaran takia olla yhteydessä bottiverkkoon suoraan omalta koneeltaan. Hyökkääjä voisi käyttää jotain kaappaamistaan koneista välikoneena itsensä ja bottiverkon välillä, mutta tämä aiheuttaisi välikoneeseen epäilyttävän paljon liikennettä. Bottiverkkojen hallintaan on käytetty ns. IRC-kanavia (Internet Relay Chat), jotka poistavat nämä ongelmat. Kaapatut koneen siirtyvät IRC-palvelimen tietylle hyökkääjän käyttämälle kanavalle odottamaan suoritettavia komentoja. Kanavalla oleminen näkyy kaapatun koneen yhteyslistauksessa mutta ei aiheuta jatkuvaa verkkoliikennettä, jonka voisi helposti huomata. IRC-palvelimet on suunniteltu hoitamaan tuhansia asiakkaita samaan aikaan, joten tällä tavalla hyökkääjällä riittää kapasitettia isonkin verkon hallitsemiseksi. (Panda Security 2008.)

Bottiverkolle on epäedullista jos laaja ja muutoin tehokas verkko voidaan sulkea yhden koneen alasajolla. Paljon tehtyä työtä ja suuri hyökkäysvoima menetetään liian helposti. Tästä syystä verkot ovat muuttumassa keskitetystä hajautettuun hallintaan, missä kaikki kaapatut koneet osaavat toimia autonomisesti sekä tarvittaessa ohjata muiden toimintaa. Bottiverkko muistuttaa toiminnaltaan vertaisverkkoa, eikä sitä voi sammuttaa yhdestä pisteestä. Tämän lisäksi joidenkin bottiverkkojen on havaittu suorastaan hyökkäävän niitä tutkivien tai uhkaavien tietokoneiden kimppuun vaikeuttaen niiden toimintaa palvelunestohyökkäyksillä. (Panda Security 2008.)

Tässä luvussa kuvailtiin osa haittatyypeistä, joista tietotekniikan käyttäjät joutuvat olemaan tietoisia päivittäin ja käyttämään näiden takia paljon aikaa järjestelmiensä suojaamisesta huolehtimiseen. Oleellista on että tilanne muuttuu jatkuvasti, ja hyvätkin suojaukset vanhenevat nopeasti hyökkääjien löydettyä uusia keinoja päästä päämääräänsä. Rikollisilla on nykyään apunaan todella taitavia ohjelmoijia, minkä voi päätellä siitä, että heidän käyttämänsä ohjelmat kehittyvät koko ajan ja ne pystyvät reagoimaan nopeasti vaihtuviin olosuhteisiin, esim. uusien haavoittuvuuksien löytymiseen. Tilanne on melko huolestuttava tällä hetkellä ja tulee ilmeisesti käymään yhä huolestuttavammaksi tulevaisuudessa. Kaikkein vaarallisin ajatus on kuitenkin se, mitä tietyntyyppiset valtiot voivat saada aikaan lähes rajattomilla resursseillaan ja kyvyllä suojella laittomilla tavoilla tietoverkoissa toimivia kansalaisiaan.

3 SUOJAUTUMISMENETELMIÄ

3.1 Käyttöjärjestelmän päivittäminen

Sen lisäksi, että tietoverkoissa eniten käytetyssä protokollaperheessä (TCP/IP) on paljon väärinkäytöksille altistavia tietoturvaheikkouksia, ovat käyttöjärjestelmät ja niiden tietoturvapuutteet suurena syynä tietoturvan nykyiseen tilaan. Voidaan sanoa, että käyttöjärjestelmä ei ole tuote, vaan prosessi. Prosessi alkaa kun käyttöjärjestelmän kehittäminen omaksi versiokseen alkaa ja päättyy, kun valmistaja lopettaa järjestelmän päivittämisen. Kun käyttöjärjestelmä julkaistaan, alkaa prosessin julkinen osuus, jonka vaiheita on mahdotonta ennustaa tarkasti edeltä käsin siihen vaikuttavien monien muutosten takia, joita joudutaan tekemään aina uusien haavoittuvuuksien tullessa julki. Kärsijöinä tässä prosessissa ovat järjestelmän käyttäjät, jotka ovat suojattomia haavoittuvuuksien hyväksikäyttäjille siihen asti, kunnes järjestelmän valmistaja on saanut haavoittuvuuden paikatuksi.

Käyttöjärjestelmän päivittäminen on oleellinen osa tietokoneen ylläpitoa. Jos päivittämätön tietokone pitää yllä internetiin näkyvää palvelua, pystyvät hyökkääjät lähes aina selvittämään tarkasti koneen käyttöjärjestelmän ja sen version. Hyökkääjät tietävät erittäin hyvin käyttöjärjestelmien tunnetut haavoittuvuudet, ja heillä on työkaluja käyttää haavoittuvuuksia hyväksi. (Scambray, McClure, Kurtz 2001.)

Henkilökohtaisten tietokoneiden käyttöjärjestelmät voidaan jakaa karkeasti kolmeen ryhmään. Ensimmäisessä on Microsoftin kaikki tuotteet ja toisessa avoimeen lähdekoodiin perustuvat käyttöjärjestelmät, joihin voidaan lukea Linux/Unix-varianttien lisäksi Sunin Solaris-käyttöjärjestelmät. Kumpaankin ryhmään julkaistaan jatkuvasti päivityksiä joko käyttöjärjestelmän muokkaamiseksi tai tietoturvan parantamiseksi. Kolmas ryhmä on Applen käyttöjärjestelmät, mutta koska niiden edustajaa ei ole mukana tämän työn käytännön osassa, ei niihin puututa tämän enempää.

Microsoftilla on säännöllisesti kuukausittain toimiva päivitysaikataulu, jolla korjataan käyttöjärjestelmän vaatimat muutokset. Tietoturvapäivityksiä tulee tarpeen

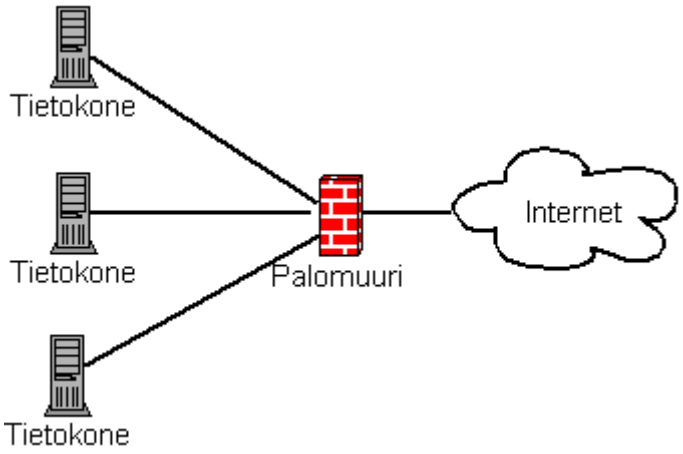
vaatiessa myös useammin. Microsoft-käyttöjärjestelmät voidaan joko konfiguroida huolehtimaan itse päivitysten ylläpidosta tai sitten järjestelmänvalvoja suorittaa päivitykset käsin. Ei ole merkitystä kumpaa tapaa käyttää, kunhan päivittäminen suoritetaan ajallaan. (Microsoft 2008.)

Linux-käyttöjärjestelmät perustuvat avoimeen lähdekoodiin, mikä ylläpidon kannalta tarkoittaa sitä, että ohjelmien uusimmat versiot ovat aina saatavilla. Monet eri Linux-jakelut julkaisevat n. puolivuosittain koko senhetkisen version uudistuksen, jonka jälkeen koko paketti on saatavilla kyseisen jakelun sivuilta internetistä. (Durham 2002.)

Kumpaakin ryhmää vaivaavat lukuisat haavoittuvuudet, joille ei tunnu olevan loppua. Avoimen lähdekoodin ohjelmien korjausten väitetään olevan Microsoftin ohjelmapaikkauksia nopeammin saatavilla. Tämä johtuu siitä, että avoimen lähdekoodin yhteisö voi reagoida välittömästi haavoittuvuuksien löytymiseen, kun taas Microsoft käyttää edeltä käsin suunniteltua aikataulua muiden paitsi todella kriittisten haavoittuvuuksien korjaamiseen.

3.2 Palomuurit

Palomuuri on laite, joka useimmiten asennetaan sisäverkon suojaksi internetin ja sisäverkon rajalinjalle. Palomuuri kykenee tarkkailemaan ja suodattamaan sen läpi kulkevaa liikennettä annettujen palomuurisääntöjen mukaisesti. Koska kaikki liikenne järjestetään kulkemaan tämän yhden väylän kautta, muodostaa palomuuri luonnollisen pisteen, jossa verkkoliikenteen tarkkailu ja estäminen on tehokkainta ja riskit verkon altistamiseksi hyökkäyksen kohteeksi samalla pienimmät. Kuviossa 2 on peruslähiverkko, joka on suojattu palomuurilla. (Zwicky, Cooper, Chapman 2001.)



KUVIO 2. Palomuurin suojaama verkko (Joutsi 2006)

Yksinkertaiset palomuurit toteutetaan pelkästään luomalla sääntöjä, joiden mukaan liikenteen kulkeminen joko sallitaan tai kielletään. Nämä säännöt perustuvat:

- IP-lähdeosoitteeseen
- IP-kohdeosoitteeseen
- Käytettyyn protokollaan
- TCP/UDP-lähdeporttiin
- TCP/UDP-kohdeporttiin
- ICMP-viestityyppiin
- paketin kokoon

Pelkkään liikenteen suodattamiseen ei tarvita vielä varsinaista palomuurilaitetta, vaan tehtävään riittää tavanomainen reititin. Reitittimet yhdistävät liikennettä sellaisten aliverkkojen välillä, joilla on eri verkko-osoite. (Zwicky, Cooper, Chapman 2001.)

Palomuurien monimutkaisemmissa toteutuksissa otetaan huomioon palomuurin läpi kulkevien yhteyksien tila. Perussääntönä on, että vain sellaiset yhteydet sallitaan, joiden lähtöpiste on sisäverkossa tai muulla palomuurin säännöissä määritellyllä turvallisella alueella. Palomuuri päästää läpi internetistä päin tulevan liikenteen, jos liikenteen alkupiste on sisäverkossa. Tarpeen vaatiessa sääntöjä voidaan muuttaa ottamaan huomioon verkon suojatulta puolelta ulospäin tarjottavat palvelut. (Zwicky, Cooper, Chapman 2001.)

Palomuurit voivat valvoa liikennettä myös ns. sovellustasolla, jolloin voidaan rajoittaa liikennettä sen mukaan, mille verkkoliikennöintiohjelmalle kuljetettavat datapaketit ovat menossa. Liikkuvien datapakettien sisältöä voidaan myös tarkkailla jossain määrin, mutta yleisesti ottaen se on melko hankalaa käytännössä. Syynä on se, että lähetettävät tietokokonaisuudet ovat yleensä suuria ja vaativat pilkkomista pienempiin osiin voidakseen päästä erilaisten yhteyksien kautta kohteeseensa internetin läpi. Palomuurin pitäisi ensin koota pilkottu lähetys, pystyä sitten päättelemään lähetyksen sisällön mahdollinen haitallisuus, ja soveltaa kuhunkin tapaukseen oikeat toimenpiteet. (Zwicky, Cooper, Chapman 2001.)

Mitä ylemmällä tasolla liikenteen kontrollointi toimii palomuurissa, sitä yksityiskohtaisemmat rajoitukset voidaan laatia. Sovellustason valvonnalla päästään hyvään tarkkuuteen, mutta saman tietokoneen samalla sovelluksella voi olla lisäksi eri käyttäjät, joita palomuuri ei tällä tasolla pysty kontrolloimaan.

3.3 Virustorjuntaohjelmat

Virusten ja muiden haittaohjelmien pääsyä työasemaan voidaan vaikeuttaa asentamalla koneelle virustorjuntaohjelma. Ohjelmat suorittavat käytön aikana siirtyvien tiedostojen tarkistuksen virusten varalta, ja niillä voidaan tehdä läpikotainen massamuistien virusskannaus, jolla yritetään löytää ne virukset, jotka ovat asentuneet ennen torjuntaohjelman asentamista, tai ovat päässeet pujahtamaan koneelle ennen virustietokannan päivittämistä. Skannaus voidaan asettaa myös alkamaan automaattisesti sopivin väliajoin, jotta voitaisiin pitää tietokone puhtaana ilman käyttäjän toimia. (Järvinen 2006.)

Torjuntaohjelmien toiminta perustuu siihen, että ne vertaavat tiedostojen sisältöä tunnistettujen virusten koodien sormenjälkiin eli ainutlaatuiseen lyhyehköön näytteeseen uuden viruksen koodista. Jos tiedostosta löytyy vastaava merkkiyhdistelmä, tiedoston todetaan olevan saastunut. Virusten etsimisessä käytetään myös erilaisia algoritmeja ja heurististiikkaa, joiden avulla ohjelmat pyrkivät syvällisemmin päättelemään tutkimastaan koodista, onko kyseessä puhdas tiedosto. Heuristiikka tarkoittaa sitä, että torjuntaohjelma ei perusta etsintäänsä pelkästään täy-

den yhdenmukaisuuden varaan, vaan osaa löytää myös merkitystä koodien samankaltaisuuksista. Algoritmit ovat kuitenkin huonoja tekemään lopullisia päätöksiä tältä pohjalta, ja viime kädessä käyttäjän on itse varmistettava poistettavien tiedostojen valinta. Virustorjuntaohjelmat pystyvät myös palauttamaan tiedostoja ennalleen tapauksissa, joissa virukset eivät ole tuhonneet tiedostoja korjauskelvottomiksi. (Järvinen 2006.)

3.4 IPv6

IP-protokollan alkuperäisen version IPv4:n toteutusta suunniteltaessa ei otettu huomioon tietoturvaan liittyviä asioita. Verkko oli tuolloin sotilas- ja yliopistopiirien suljetussa käytössä eikä nykyisenkaltaisia tietoturvaongelmia ollut. Monen protokollan tietoturvallisuus perustui siihen ajatukseen, että kaikki verkkoon kytketyt koneet oli keskitetyssä hallinnassa, eikä IP-osoitetta tai porttinumeroa voinut muuttaa tai väärentää. Toinen heikkous oli se, että autentikointia käyttävät sovellukset lähettivät käyttäjätunnuksia ja salasanoja salaamattomana verkon läpi. Käyttäjien määrän kasvettua internetin käyttö tuli entistä monimuotoisemmaksi ja tarvittiin uusia tietoturvaominaisuuksia. Näitä ominaisuuksia kehitettiin protokollien ohessa toimivina lisäosina, jotka paransivat tietoturvaa, mutta tarjosivat vain puolittaista korjausta ongelmaan. (Hagen 2002.)

IP version 6 (IPv6) on internetprotokollan seuraava kehitysaste. Sen suunnittelu aloitettiin vuonna 1994 IETF:n (Internet Engineering Task Force) toimesta. Tarkoituksena oli muuttaa IP-protokolla vastaamaan tulevaisuuden vaatimuksia ja ottaa käyttöön mm. tietoturvaominaisuudet, jotka ovat mukana alusta asti protokollan oleellisena osana. (Hagen 2002.)

IPv6:n tietoturvaan kuuluu IPSec-tuki (Internet Protocol Security), joka on ollut irrallisena käytössä myös IPv4:ssa. IPSecillä voidaan muodostaa turvallisia yhteyksiä päätelaitteesta toiseen, kahden yhdyskäytävän välille yhdistämään erillisiä lähiverkkoja tai päätelaitteesta yhdyskäytävään, jolloin kyseessä on suojaamattomasta internetistä otettu suojattu etäyhteys lähiverkon resursseihin. IPsec käyttää AH (Authentication Header) ja ESP (Encapsulating Security Payload) mekanis-

meja tietoturvan parantamiseen. AH-kenttä IP-otsikossa varmistaa paketin eheyden ja lähettäjän identiteetin. ESP puolestaan salaa paketin ja varmistaa lähettäjän identiteetin. Ennen liikenteen aloittamista suojatun yhteyden muodostavat laitteet neuvottelevat yhteyden vaatimat SA (Security Association) parametrit. Näihin kuuluvat salausavain, jota yhteys käyttää, salausalgoritmi sekä muutama algoritmin vaatima lisäparametri. (Hagen 2002.)

IPv6 parantaa tietoturvaa mutta valitettavasti sen käyttöön ei ole vielä laajassa mitassa siirrytty. Kehitys on selvästi hidastunut muutaman vuoden takaisesta tilanteesta, ja Suomessa ollaan vielä melkein kokonaan testiverkkovaiheessa. Ratkaisun avaimet ovat tällä hetkellä palveluntarjoajilla, joiden tulisi lähteä rohkeasti lisäämään uuden tekniikan käyttöä. Tilanne on kuitenkin monimutkainen, koska muutos on maailmanlaajuinen kuten internetkin, eikä kokonaisen valtionkaan strategian muuttaminen ehkä yksin riitä sysäämään kehitystä nopeampaan vauhtiin. (Nebula 2007.)

3.5 Uhkien aktiivinen tarkkailu

Käyttöjärjestelmät pystyvät keräämään omilla prosesseillaan lokitietoja tapahtumista, ja niihin voidaan asentaa ulkopuolisia ohjelmia tarvittaessa täydellisempää tarkkailua. Myös palomuureissa käytetyt tunkeutumisesto- ja tarkkailujärjestelmät kykenevät suorittamaan kattavaa liikenteen tarkkailua ja haitallisen liikenteen suodatusta. Nämä toimet ovat kuitenkin luonteeltaan puolustuksellisia. Koska aloite on aina hyökkääjällä, jää verkkojärjestelmän ylläpitäjän tehtäväksi järjestelmänsä päivittäminen uusia haavoittuvuuksia vastaan. Lisäksi sallitun ja hyökkäävän liikenteen kulkiessa samassa verkossa, on lokien seuraaminen ja analysointi vaikeaa liikenteen suuren määrän takia. (The Honeynet Project 2004.)

Hyökkääjien toimia voidaan tarkkailla aktiivisesti asettamalla yksittäinen tietokone tai verkko tarkoituksellisesti alttiiksi hyökkäyksille ja järjestämällä verkon tärkeisiin solmukohtiin laitteet, jotka tallentavat liikennevirtaa ja rajoittavat sitä tarvittaessa. Tällainen verkko voi olla eristetty muusta tuotantoverkosta, jolloin kaiken verkkoliikenteen voidaan katsoa olevan hyökkäävää tai haitallista. Aloite on

tässä tapauksessa siirtynyt verkon hallitsijalle, koska hyökkääjä ei voi mitenkään tietää yhteyden aloittamisvaiheessa olevansa tarkkailun kohteena. Hyökkääjä voi myöhemmin havaita merkkejä tarkkailusta ja päättää lopettaa järjestelmän hyväksikäyttämisen hävitettyään kaikki mahdolliset jäljet tunkeutumisesta. Tunkeutumisessa käytetyt menetelmät ovat kuitenkin jo paljastuneet järjestelmänvalvojalle, ja niistä saatua tietoa voidaan käyttää hyväksi suojautuessa uusilta tunkeutumisyrityksiltä. (The Honeynet Project 2004.)

The Honeynet Project on kehittänyt tätä ideaa ja luonut ohjelmia ja menetelmiä, joiden avulla järjestelmänvalvojat voivat seurata melko kattavasti mitä tapahtuisi, jos hyökkääjän annettaisiin vapaasti toimia verkossa. Tämän opinnäytetyön käytännön osuus on tällaisen verkon rakentaminen ja kokeileminen käytännössä.

4 HONEYNET

4.1 Yleiskuvaus

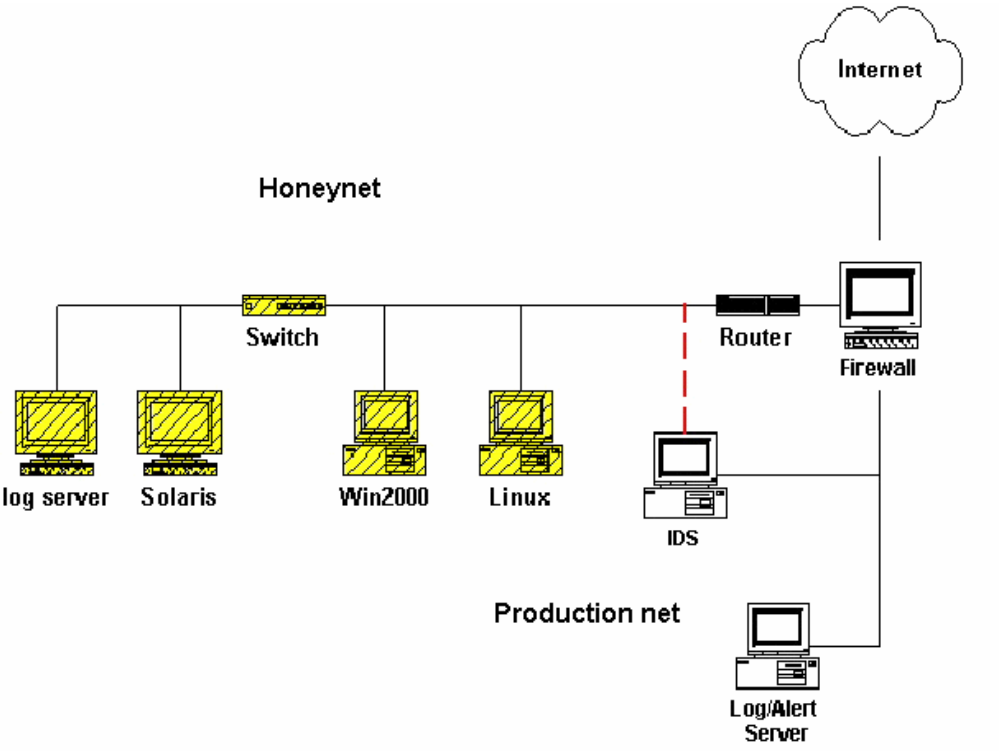
Honeynet on tietyllä tavalla järjestetty tietoverkko, jonka avulla voidaan tarkkailla ulkopuolisia tunkeutumisyrityksiä ja tallentaa niistä kertynyttä dataa sekä järjestää sitä tarkoituksenmukaisesti myöhempää analysointia varten. Verkossa olevat työasemat jätetään tarkoituksella suojattomiksi, mutta niihin kohdistuvaa datavirtaa valvomaan asetetaan erityinen laite, jonka verkkoliitynnät eivät näy IP-osoitteita käyttävälle verkkoliikenteelle. Koska työasemilla ei ole varsinaisesti mitään omaa toimintaa käyttöjärjestelmän lisäksi, on kaikki verkossa havaittu liikenne ulkopuolisen tahon aikaansaamaa. Tästä syystä voidaan katsoa, että kaikki verkossa havaittu liikenne on hyökkäävää. Koska aiheettomia hälytyksiä ei ole kerätyn datan joukossa, tulee tietojen analysointi paljon yksinkertaisemmaksi palomuurityyppisiin ratkaisuihin verrattuna. (The Honeynet Project 2004.)

Honeynetin käyttämiseen sisältyy suuria riskejä. Oman verkkonsa paljastaminen on valvotustikin vaarallista, koska hyökkääjät pystyvät käyttämään järjestelmien haavoittuvuuksia välittömästi hyväkseen saadakseen käyttäjän oikeudet hyökättävään järjestelmään. Vaikka honeynetissä käytetyt laitteet on suunniteltu äärimmäisen tietoturvallisiksi, ei voida olla varmoja siitä, ettei taitava hyökkääjä lopulta voisi huomata olevansa tarkkailtavana, löytää järjestelmästä haavoittuvuuden ja käyttää sitä hyväkseen. Tunkeutuminen tällaisen verkon tietokoneeseen on yhtä vakavaa kuin pääseminen mihin tahansa muuhun verkkoon, koska jos ulospäin suuntautuvaa liikennettä ei rajoiteta, voi hyökkääjä piilottaa itsensä käyttämällä tällaista väliasemaa ja tunkeutua uusiin kohteisiin entistä paremmin suojattuna. (The Honeynet Project 2004.)

4.2 Verkkoarkkitehtuuri

Honeynet koostuu jonkin tyyppisestä palomuurista, joka pystyy suodattamaan, rajoittamaan ja kirjaamaan liikennettä ylös, liikennettä mahdollisimman näkymät-

tömästi tarkkailevasta laitteesta sekä honeynet-verkossa sijaitsevista hyökkäyksen kohteena olevista päätelaitteista, jotka sisältävät halutut käyttöjärjestelmät. Ensimmäisessä honeynet-ratkaisussa kaksi ensin mainittua laitetta olivat erillisiä, mutta the Honeynet Projectin generation II honeynet yhdistää nämä kaksi laitetta, mikä muuttaa verkon arkkitehtuuria ja tehostaa toimintaa. Kuviossa 3 on the Honeynet Projectin esimerkkikuva generation I honeynetistä. (The Honeynet Project 2004.)



KUVIO 3. Generation I honeynet (The Honeynet Project 2004.)

4.3 Liikenteen hallinta

Kaikki verkkoliikenne kuljetetaan honeynetin palomuurin kautta, minkä jälkeen liikenne on tunkeutumisenestojärjestelmän (IDS) luettavissa. Vanhemmassa honeynet-versiossa käytetään tavanomaista palomuurilaitetta, joko kaupallista toteutusta tai jonkin käyttöjärjestelmän mukana tullutta ohjelmaa. Liikennettä voidaan joko estää kokonaan kulkemasta tai rajoittaa se tietyssä ajassa sallittujen yhteyksien määrään. IDS valitsee liikenteen virrasta kaikkein selvimmät tunkeutumisen merkkejä sisältävät yhteydet. (The Honeynet Project 2004.)

Generation II honeynet sisältää verkon kuristuspisteenä toimivan Honeywall-yhdyskäytävän, joka sisältää em. toiminnot monipuolisemmin toteutettuna. Liikenteen rajoittamiseen on kolme eri vaihtoehtoa. Yhteyksien rajoittaminen voi toimia samoin kuin aiemmassa versiossa päästäen liikenteen vapaasti sisäverkkoon, mutta rajoittaen ulospäin suuntautuvia yhteyksiä. Ulospäin suuntautuvia yhteyksiä sallitaan yleensä vain melko pieni määrä tapahtuvaksi tietyssä ajassa, jotta hyökkääjä saisi jonkun verran vasteita, mutta ei voisi huomata helposti liikenteen rajoittamista. Yhteyksiä voidaan myös hidastaa ja estää siten hyökkääjää aiheuttamasta vahinkoa murretusta honeypotista käsin. Toinen vaihtoehto on erotella tunnetut hyökkäysmuodot IDS-sääntöjen perusteella, ja hylätä niiden paketit. Tällä tavalla voidaan pienentää turhan datan määrää ja nopeuttaa tärkeän tiedon käsittelyä. Kolmas mahdollisuus on muokata paketin sisältöä. Haitallista koodia sisältävä paketti tehdään vaarattomaksi poistamalla siitä aktiivinen sisältö. Paketin annetaan kuitenkin jatkaa matkaansa, jotta voitaisiin selvittää sen tarkoitus ja jotta hyökkääjän olisi vaikeampi huomata olevansa tarkkailun kohteena. (The Honeynet Project 2004.)

Tällainen tarkasti valikoiva toiminta saadaan aikaiseksi käyttämällä iptables-palomuurin kanssa Snort-inline IDS-järjestelmää, joka on Snortin muunneltu versio. Snort-inline käyttää libipq-kirjastofunktioita ja niiden avulla paketit saadaan nostettua verkkotasolta ylemmälle tasolle, jossa niiden muokkaaminen on mahdollista. (Morris 2002.)

4.4 Datan kerääminen

Datan keräämisen tarkoitus on saada todisteita honeypotteihin kohdistuneista hyökkäyksistä. Kerätyt tiedot ovat hyödyllisiä tutkittaessa hyökkäysten teknistä taustaa, työkaluja ja ennennäkemättömiä tunkeutujien käyttämiä keinoja päästä sisään järjestelmiin. Datan keräämisen on tapahduttava luotettavasti, ja siksi se on varmistettu monen kerroksen yhtäaikaisella toiminnalla. Datan keräämisen tasot honeynetissä ovat yhteyksien ja verkkoliikenteen tallentaminen, honeypoteissa nähdyn aktiivisuuden tallentaminen ja IDS-hälytysten tallennus. (The Honeynet Project 2004.)

Yhteyksien tarkkailu ja tallentaminen tapahtuu OSI-mallin (Open Systems Interconnect) 3-4 kerroksien tietojen perusteella. Tarkkailtavia tietoja ovat mm. yhteyksien IP-osoitteet, käytetyt protokollat ja porttinumerot. Verkkoliikenne tallennetaan kokonaisuudessaan, jotta saadut tiedot olisivat mahdollisimman kattavat ja jopa päällekkäisiä muiden tasojen keräämien tietojen kanssa. Tietojen päällekkäisyys on hyödyllistä siltä varalta että jonkin kerroksen toiminta estyisi. Honeypoteista tallennetaan järjestelmälokin ja sovellusten kirjaamat tiedot. Jos hyökkääjä käyttää salattua yhteyttä, on näppäilyjen tallentaminen honeypotissa ainoa keino päästä näkemään hyökkääjän toimia. IDS:n hälytykset antavat melko luotettavan varoituksen hyökkäyksen alkamisesta. Kummassakin honeynet-verkossa keinot tietojen keräämiseksi ovat samanlaisia, mutta generation II honeynet käyttää pidemmälle kehitettyjä työkaluja ja kaikki toiminnot tapahtuvat Honeywall-yhdyskäytävän sisällä. (The Honeynet Project 2004.)

4.5 Tietojen analysointi

Tietojen analysointi tarkoittaa mielenkiintoisten osien erottamista kaapatusta kokonaisliikenteestä ja näiden tapahtumien selittämistä ymmärrettävästi. Tallennetun datan määrä on valtava ja koska liikennettä tallennetaan yleensä siinä muodossa, jossa se on verkossa liikkuessaan, tarvitaan analysoinnissa apuna monenlaisia suodattavia ohjelmia. Suodattaminen ja liikenteen jaottelu tapahtuu pitkälti samojen ehtojen mukaan kuin palomuureissa, mutta analysoinnissa on kyettävä myös yhdistämään samaan yhteysjaksoon kuuluvat paketit toisiinsa, jotta niiden sisältämä kokonaisuus voitaisiin saada selville. (The Honeynet Project 2004.)

Generation I honeynet tarjoaa analysointiin vain samat työkalut joilla tietoja kerätään. Ohjelmien lisävalitsimilla voidaan lisäksi vaikuttaa tallennettavan tiedon määrään ja tarkkuuteen sekä tietoa kerättäessä että katseltaessa. Palomuurin tallentamia yhteyksiä on suodatettu jo tallennusvaiheessa ja analysoinnissa voidaan käyttää hyväksi käyttöjärjestelmän omia työkaluja (grep, sort, yms.), jotta saataisiin aikaan mahdollisimman selkeitä kokonaisuuksia. Liikennettä tarkkailtaessa data kerätään ja tallennetaan binäärimuodossa, jotta sen käsittely olisi mahdollisimman nopeata. Generation II honeynet käyttää tietojen keräämiseen Honeywall-

yhdyskäytävää, jonka käyttöliittymä on optimoitu datan analysointia silmällä pitäen.

4.6 Honeywall

Honeywall toimii yhdyskäytävänä honeypotit sisältävän verkon ja internetin välillä generation II honeynetissä. Honeywallissa on vähintään kaksi verkkoliityntää, joiden kautta tarkkailtava liikenne kulkee. Nämä toimivat OSI-mallin tasolla 2 ja ovat huomaamattomia verkkoliikenteessä. Kolmas mahdollinen liityntä on laitteen hallintaa ja datan analysointia varten ja siinä on TCP/IP-pino asennettuna eli laitteen hallintaa voidaan suorittaa etäkäyttönä. Laitteen tarjoama kerätyn datan analysointi on mahdollista vain tätä liityntää käyttämällä, joten sen ottaminen mukaan konfiguraatioon on ehdottomasti suositeltavin vaihtoehto. Järjestelmä asentuu siten, että ulkoverkon liityntä on eth0, sisäverkon eth1 ja hallinnan eth2. (The Honeynet Project 2004.)

Hallintaliittymä toimii verkkokerroksella, mikä mahdollistaa laitteen hallinnoimisen käyttäen tavanomaista selainohjelmaa. Samasta syystä liittymän havaitseminen porttiskannauksella on rutiininomainen tehtävä. Tämän liitynnän on paras sijaita omassa verkossaan, jottei hyökkääjä voisi helposti huomata olevansa tarkkailtavana ja päästä muokkaamaan konfiguraatiota. (The Honeynet Project 2004.)

4.7 Honeypot

Honeypotit voivat olla tavallisia työasemia, joissa on kussakin yksi käyttöjärjestelmä ja niissä tarvittavat ohjelmat ja palvelut asennettuna. Käyttämällä eri käyttöjärjestelmiä (Windows, Linux, Solaris yms.) voidaan tutkia niille ominaisia haavoittuvuuksia ja sitä, miten hyökkääjät pystyvät näitä hyödyntämään. Honeypotit sijaitsevat tarkkailtavan sisäverkon puolella, jossa kaikki tavattu liikenne on hyökkäävää, lukuun ottamatta pientä määrää satunnaisia käyttöjärjestelmien itsenäisesti generoimia signaaleja. (The Honeynet Project 2004.)

Honeypotit voidaan toteuttaa myös virtuaalisesti, jolloin riittää yhden fyysisen työaseman käyttäminen. Kaikki käyttöjärjestelmät asennetaan samaan koneeseen ja ne voivat olla käynnissä yhtä aikaa. Verkkoarkkitehtuurin kannalta nämä vaihtoehdot ovat aivan samanlaiset. Virtualisoinnilla saadaan se hyöty, että testattaessa erilaisia verkon tilanteita käyttöjärjestelmiä ei tarvitse asentaa jokaista testiä varten alusta asti uudelleen. Virtuaalisen kovalevyn voi käyttöjärjestelmän puhtaan asennuksen ja asetusten konfiguroinnin jälkeen tallentaa muualle ja kopioida käyttöön ennen uuden testijakson alkua. (The Honeynet Project 2004.)

4.8 Laajennukset

Honeynetin päätarkoitus on kerätä yksityiskohtaista tietoa verkkoon pyrkivästä luvattomasta liikenteestä ja menetelmistä, joita vieraisiin järjestelmiin tunkeutumisissa käytetään. Yhdessä pisteessä toimivan honeynet-verkon näkemä liikenne on vain häviävän pieni osa kaikesta mahdollisesta liikenteestä ja sen antama kuva internetissä tapahtuvasta toiminnasta on väkisinkin suppea ja perustuu pitkälti sattumanvaraisiin tapahtumiin. Tarkempi kuva saadaan muodostettua yhdistämällä eri puolilla maailmaa sijaitsevia erilaisia kokoonpanoja edustavia honeynet-verkkoja. (The Honeynet Project 2004.)

Usealla honeynetillä voi olla yhteinen lokipalvelin, jonne kaikista yksittäisistä honeyneteistä kerätyt tiedot lähetetään. Palvelin tallentaa kaikkien hallinnoimiensa honeynettien datan ja pystyy näin ollen suorittamaan monipuolisempia tietokantahakuja ja antamaan laajemman kuvan tapahtumista. Tämän menetelmän hyvänä puolena on myös se, että riski joutua hyökkäyksen kohteeksi on hajautettu vaikka päästäänkin samalla käsiksi suurempaan määrään tietoja. Toinen tapa laajentaa honeynettiä maantieteellisesti on yhdistää eri puolilla olevia verkkoja yhteen VPN-tunneleilla (Virtual Private Networking). (The Honeynet Project 2004.)

5 HONEYPOT APUOHJELMAT

5.1 Yleistä apuohjelmista

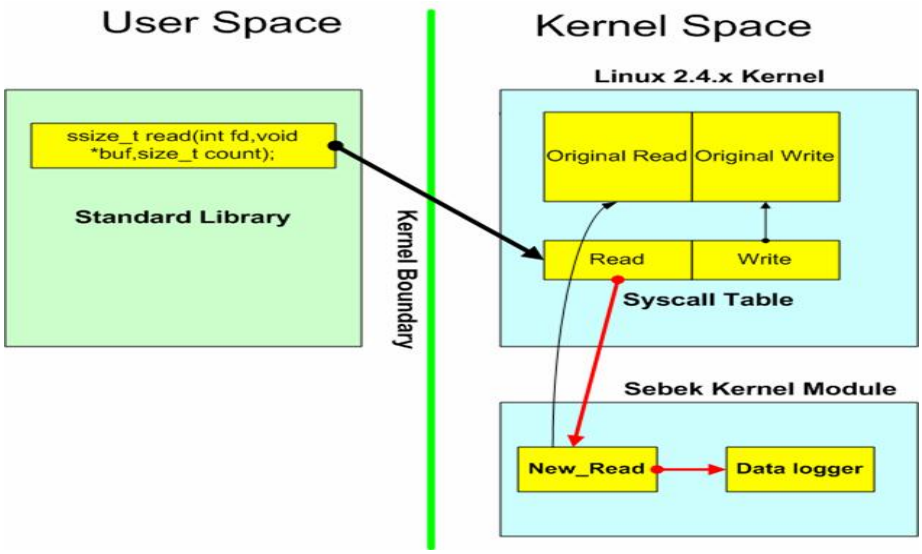
Palomuurin lokit ja verkon tapahtumien seuraaminen antavat täydellisen kuvan honeynetissä kulkevasta liikenteestä ja auttavat tunnistamaan mahdollisia kiinnostavia tapauksia eli melko varmoja tunkeutumisia muun internetin pohjakohinan seasta. Jos hyökkääjä on onnistunut tunkeutumaan järjestelmään ja saanut siirrettyä sinne salausapuohjelmia, ei pelkästä liikenteen seuraamisesta ole enää mitään hyötyä. Hyökkääjän antamat komennot ovat täysin näkymättömissä salatun yhteyden takana. (The Honeynet Project 2004.)

Jotta hyökkääjän tapaa toimia voisi tarkkailla tässäkin tapauksessa, käytetään datan keräämisessä vielä kolmatta tasoa. Tämän tason tarkoituksena on ottaa talteen kaikki hyökkääjän honeypotissa aikaansaamat tapahtumat hyökkääjän sitä huomaamatta. Näiden tietojen avulla voidaan rakentaa täydellinen kuva tapahtuneesta hyökkäyksestä. Tärkeitä tietoja ovat mm. tapa, jolla järjestelmään on tunkeuduttu, tarkat hyökkäyksen ajankohdat ja se, mitä hyökkääjä teki päästyään sisään järjestelmään. Tietoja kerätään mm. järjestelmän- ja sovelluksen lokitiedostoista sekä näppäilyjentallentajalla, joka toimii heti liikenteen salauksen purkamisen jälkeen hyökkääjältä piilotettuna prosessina. Tiedon keräämiseen on myös kehitetty monia tehokkaita apuohjelmia, joilla saadaan näkyviin käyttöjärjestelmien piilotettuja tapahtumia. (The Honeynet Project, 2004.)

5.2 Sebek

Sebek on ohjelma, joka asennetaan honeypottiin poimimaan hyökkääjän suorittamat näppäimen painallukset ja lähettämään ne huomaamattomasti lokia pitävälle palvelimelle. Toiminta on saatu huomaamattomaksi käyttämällä järjestelmän ytimeen ladattavaa osaa, joka piilottaa tietyt toiminnot näkyvistä käyttäjätasolla. Menetelmä on samanlainen kuin hyökkääjien asentamissa rootkiteissä, joita käytetään usein salasanojen varastamiseen. Sebek toimii niin, että se vaihtaa ytimen

read()-funktion muokattuun funktioon. Kuviosta 4 näkyy, kuinka uusi funktio kirjoittaa ensin painallukset lokiin ja vasta sen jälkeen tekee alkuperäisen luku-operaation. (The Honeynet Project 2004.)



KUVIO 4. Sebekin read()-funktio (The Honeynet Project 2004.)

Sebek muodostaa lokipalvelimelle lähetettävän paketin jokaisen lukuoperaation jälkeen. Paketit muistuttavat UDP-protokollan paketteja, mutta niissä on joitain ainutlaatuisia kenttiä mm. magic number, jonka tarkoitus on tehdä paketit näkymättömiksi muille verkossa oleville honeypoteille ja vaikeuttaa näin hyökkääjää havaitsemasta tarkkailua mahdollisesti kaapatuista honeypoteista käsin. Pakettien lähettäminen verkkoon on piilotettu järjestelmältä lähettämällä ne Sebekin oman kernel-moduulin kautta. Lähettämisessä ei käytetä järjestelmän TCP/IP-pinoa, vaan Sebek-moduuli käyttää verkkoliityntää suoraan. (The Honeynet Project 2004.)

Sebekin lähettämiä tuloksia voidaan tarkastella joko konsolissa tai käyttämällä web-pohjaista käyttöliittymää. Konsolia käytettäessä tulokset voidaan purkaa tcpdump-muodossa ja lukea tarvittaessa, tai lukea suoraan konsolista reaaliajassa sitä mukaa kuin komentoja saapuu. Web-käyttöliittymä yhdistää nämä toiminnat, ja sillä on helpompi saada selkeä kokonaiskuva tapahtumista verkossa, jossa on paljon tarkkailtavia honeypotteja. (The Honeynet Project 2004.)

5.3 Capture-Bat

Capture-Bat on apuohjelma, jolla voidaan tarkkailla käyttöjärjestelmän tilanmuutoksia ajettaessa ohjelmia win32-järjestelmissä. Ohjelma auttaa havaitsemaan haitallisen koodin vaikutukset, jotka jäävät helposti huomaamattomiksi järjestelmän muun toiminnan ohessa. Ohjelman antama palaute on yksityiskohtaista ja sen perusteella voidaan ilman lähdekoodia päästä perille hyökkääjän ohjelman sisäisestä toiminnasta. Capture-Bat tarjoaa hyvät suodatusominaisuudet, joiden avulla saadaan pienennettyä tulostuksen määrää käyttöjärjestelmän normaalin taustatoiminnan ja luvallisten sovellusten aiheuttamien tulosteiden jäädessä pois. (The New Zealand Honeynet Project 2007.)

Capture-Bat tarkkailee järjestelmän tiedostojärjestelmää, rekisteriä ja suorituksessa olevia prosesseja sekä tulostaa konsoliin näissä havaittujen tilojen muutoksia. Ohjelma lukee kunkin kolmen monitorin listasta (exclude list) tunnetut tapahtumat, joiden tulostus jätetään pois lopullisesta näytettävästä tulostuksesta. Listat ovat tekstitiedostoja, ja niitä muokkaamalla voi ohjelman toiminnan ja tulosteen määritellä tarkalleen halutun mukaiseksi. Ohjelmaa kehittävältä yhteisöltä on saatavilla valmiita listoja monia eri käyttöjärjestelmiä ja kokoonpanoja varten. (The New Zealand Honeynet Project 2007.)

Capture-Bat lukee kaikki tapahtumat Windowsin kernel-tasolla (kernel level) ja lähettää ne prosessoitavaksi käyttäjätasolle (user space). Ohjelma asentaa käyttöjärjestelmän ytimeen oman ajurin, jonka avulla se pääsee lukemaan kolmen monitorin lukufunktioita suoraan ja kykenee siten tallentamaan tapahtumat. Käyttäjätason vastuulla on listan läpikäyminen ja kaikkien sellaisten tapahtumien tulostaminen, joille ei löydy listasta vastinetta. (The New Zealand Honeynet Project 2007.)

5.4 P0f

P0f on apuohjelma, jolla voidaan suorittaa passiivista yhteyden vastapään tunnustelua (fingerprinting). Tunnusteleva laite vastaanottaa tietoja, joita yhteyden vas-

tapuoli lähettää normaalin yhteyden aikana ja muodostaa niiden pohjalta mahdollisimman täydellisen kuvan vastapuolen laitteistosta, käyttöjärjestelmästä, verkonkomponenteista yms. yksityiskohdista. Passiivisessa tunnustelussa ei liiku ylimääräistä dataa yhteyden ääripäiden välillä, vaan tiedot muodostetaan datapakettien otsikkokenttien arvojen perusteella. Käyttöjärjestelmien verkko-ohjelmien toteutuksissa on pieniä eroja tai suoranaisia virheitä ja näiden erojen vaikutuksesta tiettyjen otsikkokenttien oletusarvot muodostuvat käyttöjärjestelmäriippuvaisesti. Esimerkiksi TTL (Time To Live), joka määrittelee IP-paketin elinajan hyppyinä verkossa, on Windows XP:ssä oletuksena 128 ja VectorLinuxissa 64. (Zalewski 2006.)

P0f pystyy suorittamaan tunnustelua sekä itse aloitetuista että omaan palveluun kytkeytyneistä yhteyksistä. Ohjelmalla on kolme pääasiallista tapaa kerätä tietoja. Sisään tulevien yhteyksien (SYN mode) tunnustelulla saa helposti paljon suoraa tietoa, koska esim. HTTP-palveluun kytkeytyvä selain kertoo itsestään ja isäntälaitteestaan todella paljon yksityiskohtia heti ensimmäisillä siirretyillä datapaketeilla. Itse aloitetuissa yhteyksissä (SYN - ACK mode) yhteydenpito palvelinpuolen kanssa aloitetaan TCP:n mukaisella alkukättelyllä, jonka aikana pakettien otsikot antavat vihjeitä vastapuolen ominaisuuksista. Yhteys jatkuu tämän jälkeen tarvittaessa. Jos vastapuoleinen palvelin kieltäytyy kommunikoinnista (RST+ mode), se lähettää kuitenkin yhteyden aloittajalle tiedon tästä normaalilla TCP-paketilla, josta voidaan lukea mainittuja tietoja. (Zalewski 2006.)

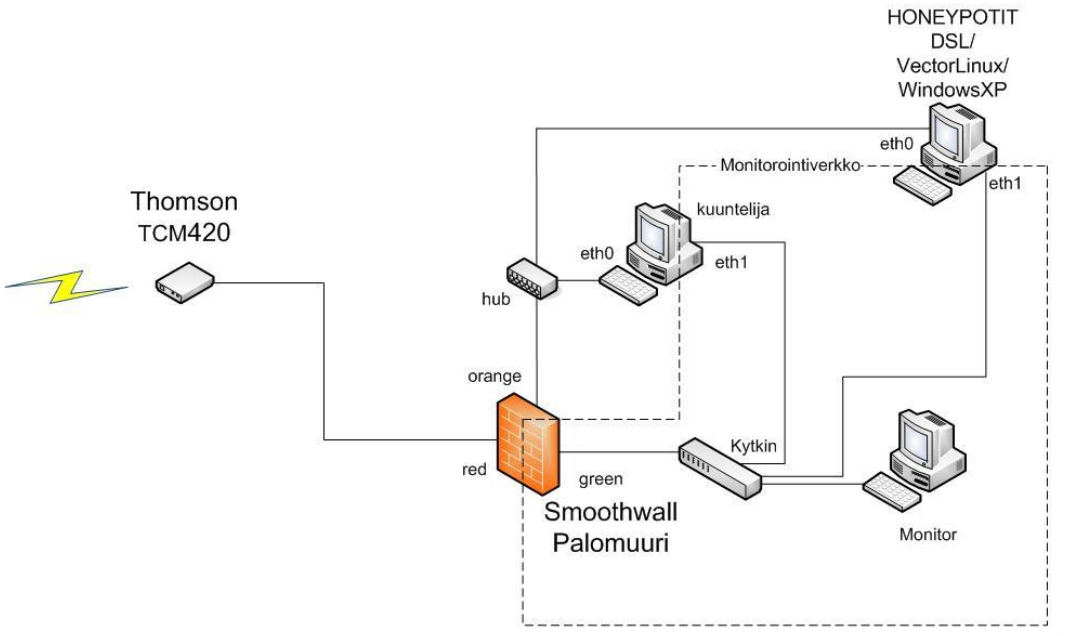
Honeypotteihin on tarjolla paljon muitakin apuohjelmia, jotka osaltaan lisäävät hyökkääjän toimista luettavissa olevan tiedon määrää. Tässä työssä oli honeypotteihin asennettuna ohjelmana käytössä vain Sebek. Muut olivat mukana vain mielenkiinnosta sekä apuna honeypottien testaamisessa ennen testijakson alkua. P0f sisältyy lisäksi generation II honeynet yhdyskäytävälaitteeseen yhteyksien vastapäiden ominaisuuksista tietoa keräävänä ohjelmana.

6 HONEYNET KÄYTÄNNÖSSÄ

6.1 Generation I Honeynet

6.1.1 Testiympäristön kuvaus

Generation I honeynetin testiympäristö sisälsi Thomson TCM420-kaapelimodeemin, kolmiporttisen palomuurin, honeypotit, jotka on toteutettu virtuaalisesti yhteen työasemaan sekä tietokoneen, joka kuuntelee huomaamattomasti palomuurin ja honeypottien välistä verkkoa. Verkko oli kuvion 5 mukainen. Palomuurin red-liityntä vastaanottaa kaikki ulkoapäin tulevat yhteydet, jotka ohjataan orange-liittymästä honeypotteja kohti. Orange-liityntä päästää kaiken liikenteen sisään honeynettiin, mutta ulospäin suuntautuvaa liikennettä on rajoitettu. Kuuntelevan tietokoneen eth0-liityntä on promiscuous-tilassa, eli se pystyy vastaanottamaan kaikki verkossa liikkuvat paketit. Hyökkääjä ei pysty havaitsemaan sen läsnäoloa suoraan, koska palomuuri ohjaa liikenteen vain honeypottien IP-osoitteisiin ja kuuntelijan kaikki verkkopalvelut on kytketty pois päältä.



KUVIO 5. Generation I honeynet-testiympäristö

Sekä virtuaalikoneiden palvelin että kuunteleva tietokone on varustettu kahdella verkkokortilla, jotta niiden hallinta voitaisiin suorittaa turvallisen verkon puolelta ja vältyttäisiin aiheuttamasta ylimääräistä liikennettä tarkkailtavaan verkkoon. Monitor-työasemalla on yhteys kaikkiin verkon laitteisiin ja se voi käyttää internetiä melko suojatusti, koska palomuuri sallii green-verkkoon vain ne yhteydet, joiden alkupiste on lähtöisin green-verkosta.

6.1.2 Asennus ja konfigurointi

Asennus jakautui kolmeen osaan: Smoothwall-palomuurin asennukseen ja konfigurointiin, honeynet-verkkoon liitetyn kuuntelijan asennukseen ja konfigurointiin sekä virtuaalisten honeypot-työasemien asennukseen. Smoothwall-palomuurin asennuspaketin ja ohjeet voi ladata sivulta http://www.smoothwall.org/get/. Smoothwallin alustaksi riittää kolmella verkkokortilla varustettu PC-tietokone, jossa on vähintään 500MHz:n prosessori ja muistia 128Mb. Vaatimattomampaakin laitetta voidaan tarvittaessa käyttää varsinaisten suojaustoimintojen toteuttamiseen, mutta web-pohjainen hallinta toimii sujuvammin kohtalaisen tehokkaalla koneella. Asennus suoritetaan CD-Rom-levyltä, johon on poltettu ladattu isotiedosto. Asennus tapahtuu suoraviivaisesti ilman käyttäjältä vaadittavia vaikeita konfigurointeja. Verkkokorttien MAC-osoitteet (Media Access Control) on hyvä olla tiedossa ennen asennusta, jotta kukin liittymä tulisi valmiiksi kytketyksi oikeaan verkkoon ja internetin suuntainen liittymä voisi saada IP-osoitteen dynaamisesti palveluntarjoajalta. Palomuurin oletusasetukset ovat riittävät suojaamaan sen takana olevia verkkoja heti asennuksen jälkeen. Taulukossa 1 on palomuurin sisältämät laitteet.

TAULUKKO 1. Smoothwall-laitteisto

Prosessori	**Muisti**	**Kiintolevy**	**Verkkokortit**	**Emolevy**
Intel Celeron 700MHz	144MB	Seagate 10GB	Realtek RTL8129	MS-6178 v.1.0
			Realtek RTL8129	
			Realtek RTL8129	

Palomuurissa on mukana myös Snort-niminen tunkeutumisen havaitsemisjärjestelmä. Se on oletuksena pois toiminnasta ja käynnistäminen vaatii ns. Oink-koodin hakemisen www.snort.org-sivulta. Ennen koodien lataamista on hakijan rekisteröidyttävä palveluun. Palomuurin oletusasetuksissa on orange-liittymän liikenne kokonaan kielletty, ja siksi niille menevä reitti on määriteltävä red-liittymään, jotta honeypotit olisivat tavoitettavissa ulkoverkosta. Koska honeypottien on tarkoitus näkyä kokonaan internetin suunnasta, sisäänpäin suuntautuva liikenne sallitaan kaikkiin porttinumeroihin. Ulospäin suuntautuvasta liikenteestä sallitaan turvallisuussyistä vain FTP:n, SSH:n ja HTTP:n porttinumerot.

Kuuntelija on tavallinen työasema, jossa on kaksi verkkokorttia. Käyttöjärjestelmänä on DSL-N, joka on honeypottina käytetyn DSL:n uudemmalla kernelillä varustettu laajennusversio. Asennuksen jälkeen on käyttöjärjestelmään vielä haettava datan keräämisessä käytettävät Snort- ja Tcpdump-ohjelmat MYDSL-Browserilla.

Honeypotit on toteutettu virtuaalisesti, eli niiden käyttöjärjestelmiä ajetaan ohjelmallisesti samassa tietokoneessa alustana toimivan käyttöjärjestelmän päällä ilman suoraa yhteyttä laitetasoon. Alustana on CentOS 4.6 palvelin minimaalisella asennuksella. Minimaalisen asennuksen lisäksi järjestelmästä on poistettu kaikki ylimääräiset palvelut, jotta saataisiin mahdollisimman paljon tietokoneen resursseja virtuaalikoneiden käyttöön. Tietokoneessa on kaksi verkkokorttia joista toisen kautta kulkee kaikkien honeypottien liikenne ja toinen on pelkästään hallintaa ja honeypottien työpöytien ohjausta varten. CentOS 4.6 -asennuksen ja konfiguroinnin vaiheet on kuvattuna liitteessä 1. Taulukossa 2 on listattuna CentOS-palvelimen tärkeimmät komponentit.

TAULUKKO 2. CentOS-palvelimen komponentit

Prosessori	**Muisti**	**Kiintolevy**	**Verkkokortit**	**Emolevy**
AMD 2200+	768MB	Quantum 15GB	3Com 3c905	Soltek SL-75LIV
		WD 160GB	Davicom DM9102	

Seuraavaksi asennetaan virtuaalipalvelin, joka tarjoaa ohjelmallisen laitealustan virtuaalisille työasemille. Ohjelmana käytetään VMware Server 1.0.3 -ohjelmistoa ja se on vapaasti ladattavissa sivulta http://www.vmware.com/download/server/. Ohjelman käyttäminen vaatii rekisteröitymisen VMwaren tietokantaan. Rekisteröitymisen jälkeen saadaan koodiavaimet, joita tarvitaan asennuksen aikana. Ennen VMware Server 1.0.3 -asennusta on asennettava seuraavat CentOS:n minimaalisesta asennustavasta puuttuvat osat järjestelmään.

```
[root]# yum install gcc
[root]# yum install xinetd
[root]# yum install kernel-devel
```

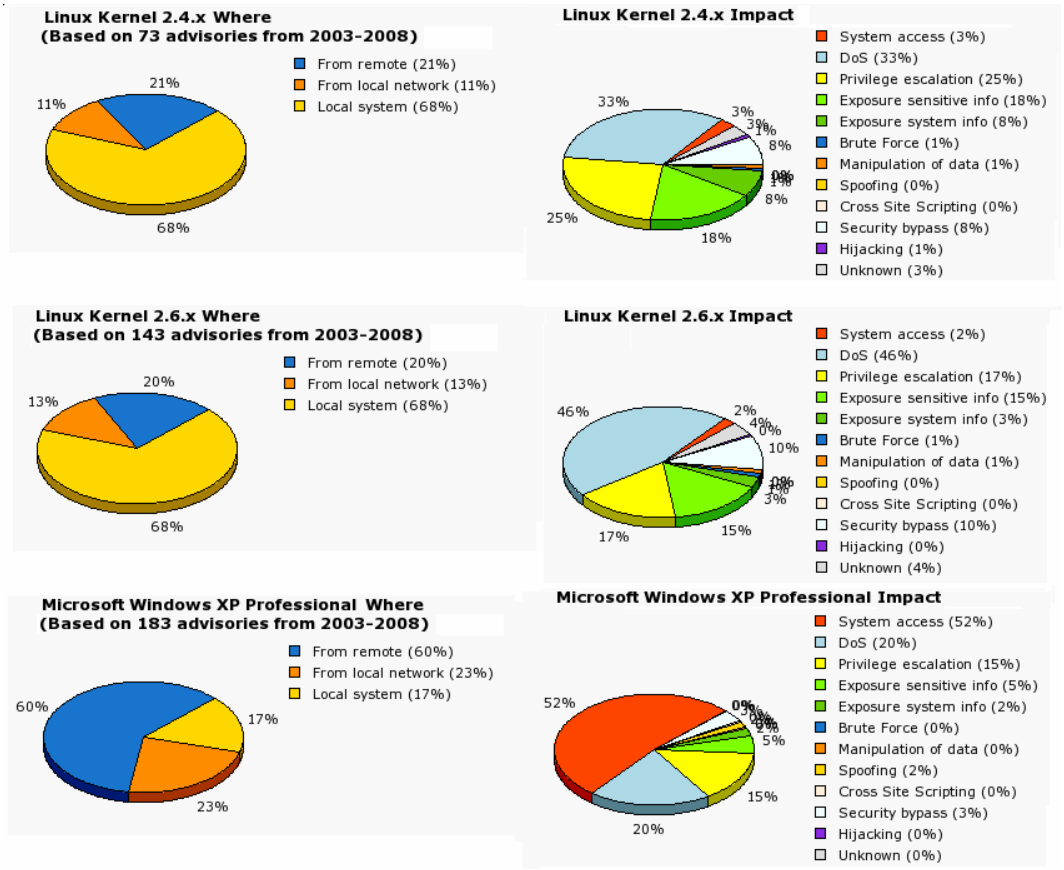
VMware Server 1.0.3 asennus tapahtuu seuraavilla komennoilla.

```
[root]# tar xzvf VMware-server-1.0.3-44356.tar.gz
[root]# cd vmware-server-distrib
[root]# ./vmware-install.pl
```

Asennuksen konfigurointiosassa ohjelma kysyy parametriarvoiksi mm. ohjelman ja sen eri hakemistojen sijainteja sekä muita ohjelman toiminnan kannalta välttämättömiä tietoja. Kaikkien kysymysten kohdalla voidaan käyttää tarjottua oletusvaihtoehtoa. Konfiguroinnin päätteeksi ohjelma pyytää syöttämään rekisteröitymisen yhteydessä saadun koodiavaimen, jonka jälkeen asennus on valmis.

Honeypotteina käytössä olivat DSL 4.2.5 (Damn Small Linux), VectorLinux 5.9 Gold ja Windows XP SP2. Kaksi erilaista Linux-jakelua oli mukana, koska ne pohjaavat eri kernel-versioihin, ja niiden tietoturvallisuudessa on siitä syystä eroja. DSL:n ytimenä olevassa 2.4-kernelissä on tunnettuja tietoturvahaavoittuvuuksia, jotka on korjattu VectorLinuxin 2.6-versiossa. Windows XP tuo mukaan omat tietoturvaominaisuutensa ja täydentää siten kokonpanoa. Kuviossa 6 on www.secunia.com -sivulta havainnollista tilastotietoa honeypotteina käytettyjen käyttöjärjestelmien kohtaamista tietoturvauhkista. Vasemmanpuoleiset kuviot esittävät hyökkäyksien lähtöpaikkojen suhteelliset osuudet ja oikeanpuoleiset jakautumia erilaisten tietoturvaloukkausten kesken. Linux-kerneleiden kuviot ovat silmiinpistävän samanlaisia, ja kummassakin selvästi suurin osa tapahtuneista loukkauksista on lähtöisin paikallisesta järjestelmästä. Kummankin kernel-version

tyypillisin tietoturvauhka on palvelunestohyökkäys, mikä ei ole kovin kriittinen uhka työasemille, vaan suuntautuu enimmäkseen palvelimia kohtaan. Windowsin suurin uhka on järjestelmään tunkeutuminen ja se suoritetaan suurella todennäköisyydellä etäyhteyden takaa. Sivulla on vastaavanlaista tietoa myös eri käyttöjärjestelmien haavoittuvuuksien kriittisyyden tasosta. Windowsin haavoittuvuuksista oli 38 % kriittisiä tai erittäin kriittisiä, samalla kun Linux-kernelien haavoittuvuudet olivat vain keskitasoa tai matalampia.



KUVIO 6. Honeypotit tietuturvauhkien kohteina (Secunia 2008.)

VMware serveriä käytetään monitorikoneesta asiakasohjelman välityksellä. Asiakasohjelma ladataan serveriohjelman ohella samalta WMwaren internet-sivulta. Asiakasohjelmalla otetaan yhteys VMware serverin hallinnointiliittymään, joka oletuksena kuuntelee porttia 902. Ennen virtuaalisten työasemien asentamista on luotava virtuaaliset kovalevyt käyttöjärjestelmiä varten. Levyjen luomisen vaiheet löytyvät liitteestä 2. Kun levyt on luotu, voidaan käyttöjärjestelmät asentaa joko isäntäkoneen (CentOS) CD-Rom-asemasta tai suoraan iso-tiedostoista, jotka

kopioidaan ennen asennusta serverin tiedostojärjestelmään. Ennen asennuksen käynnistämistä valitaan käytetty asennustapa.

Inventory -> [Virtual Machine] -> CD-Rom (IDE 1:0)
Connection (Use physical drive / Use ISO image) (-> Browse)

Käyttöjärjestelmät voidaan tämän jälkeen asentaa tavalliseen tapaan.

Verkon käyttäytymistä ja eri käyttöjärjestelmien vaikutusta hyökkäävän liikenteen kohdistumiseen tiettyyn kohteeseen oli tarkoitus tutkia käyttämällä kaikissa käyttöjärjestelmissä samoja ulospäin näkyviä palveluita. Palvelut olivat FTP, SSH ja HTTP, joiden tavanomaiset porttinumerot jätettiin auki muiden porttien ollessa suljettuina. Windowsissa oli lisäksi käytössä järjestelmän oma palomuuri suojaamassa oletusasennuksessa avoimiksi jääviä porttinumeroita. Nämä ovat tiettyjen haittaohjelmien pääkohde ja mahdollistavat päivittämättömän järjestelmän hyväksikäyttämisen.

Palveluiden ohjelmat ja ohjelmaversiot ovat miltei kaikissa tapauksissa erilaisia eri käyttöjärjestelmien välillä. DSL:ssa kaikki käytetyt palvelut saa käynnistettyä suoraan työpöydältä, mutta Windowsissa ne ovat lisäosia ja vaativat asennuksen käyttöjärjestelmän asennuslevyltä. VectorLinuxiin, jonka versio on toimistokäyttöön tarkoitettu, ei ole suoraan saatavilla FTP- ja HTTP-palveluita, ja ne on siksi rakennettava lähdekoodista. Cruxports4slax-ohjelmaa apuna käyttäen palvelut saadaan toteutettua koneelle melko helposti. FTP (pure-ftpd) asentuu koneelle suoraan toimivana, mutta HTTP:n (lighttpd) asennuksessa muodostuu viallinen konfigurointitiedosto, joka estää palvelun käynnistymisen. Virheen saa korjattua käynnistysyrityksessä saatujen virheilmoitusten perusteella. Lisäksi /var/log/lighttpd/-hakemistossa olevat access.log- ja error.log-tiedostot tarvitsevat laajemmat käyttöoikeudet, ennen kuin palvelu toimii kunnolla. SSH on linux-jakeluissa yleensä mukana automaattisesti, mutta Windowsiin se on asennettava erikseen. Tässä työssä käytetyn version asennusohje löytyy sivulta http://pigtail.net/LRP/printsrv/cygwin-ssh.html. Käytetyt palvelut ja niiden versiot ovat listattuna taulukossa 3.

TAULUKKO 3. Honeypot-palvelut ja niiden versiot

	Windows XP	**VectorLinux**	**DSL**
SSH	OpenSSH 4.7	OpenSSH 4.7	OpenSSH 3.6.1
FTP	XP SP2:n FTP-palvelin	Pure-ftpd 1.0.21	Betafttpd 0.08
HTTP	XP SP2:n HTTP-palvelin	Lighttpd 1.4.18	Monkey 0.9.2

6.1.3 Liikenteen tallennus ja analysointi

Alustavan suunnitelman mukaan liikennettä tallennettaisiin neljän vuorokauden ajan niin, että kaikkien honeypottien liikenne saataisiin kerättyä yhtä aikaa. Käytetty palomuuri tukee kuitenkin vain yhtä IP-osoitetta red-liittymässä, joten lopulta honeypottien liikenne jouduttiin tallentamaan yksitellen, jokaisen näkyessä verkossa yhtäjaksoisesti kahden vuorokauden (48h) ajan.

Verkkoliikenne otettiin talteen kuuntelijaan Tcpdump- ja Snort-ohjelmilla. Tcpdump-tallennus on binäärimuotoista ja snort-tallennus ASCII-muodossa helpomman luettavuuden takia. Lisäksi honeypotit suorittivat tallennusta siltä varalta, että kuuntelija olisi jostain syystä sammunut. Windows XP:ssä tallennus tehtiin Etherealilla ja kummassakin Linuxissa Snort-ohjelmaa käyttäen. Kuuntelijassa komentoriviltä annetut käskyt olivat:

```
snort -dvi eth0 -l snort_hakemisto/
tcpdump -w tallennus_tiedosto
```

Käytetyllä komennolla Snort sekä tallentaa että lähettää kaapattua liikennettä näytölle, joten liikennevirtaa pystyy seuraamaan myös reaaliaikaisesti. Edellämainitulla komennolla Tcpdump kirjaa kaiken näkemänsä liikenteen käynnistysparametrinä saamaansa tiedostoon ilman näytölle tulostamista. Näytön päivittäminen on yksi tietokoneen hitaimpia prosesseja ja kirjaamalla verkkoliikenne tiedostoon ilman tulostamista voidaan välttää liian nopeasta liikenteestä johtuva pakettien pudottaminen.

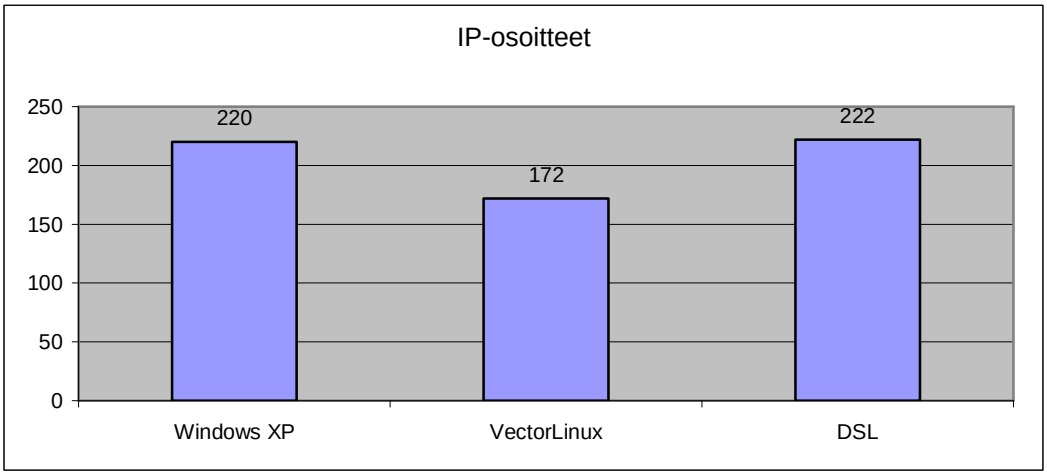
Tallentamisen tuloksena saatiin jokaiselta käyttöjärjestelmältä yksi suuri tcpdump-tiedosto ja hakemisto IP-osoitteita, joiden alla oli kyseisten osoitteiden kanssa vaihdettu data. Datan suodattamiseen tarvittavia valmiita komentoja ei ollut. Seuraavilla yhdistelmäkomennoilla saa tarpeellisia tietoja kaivettua data-möhkäleistä esiin.

IP-osoitteiden määrä	ls snort_hakemisto/ | wc -l
yhteyksien määrä	1. find snort_hakemisto/* | wc-l
	2. luvusta on vähennettävä ip-osoitteiden määrä
pakettien määrä	cat snort_hakemisto/*/* | grep =+=+ | wc -l
toistuvia IP-osoitteita	1. ls honeypotX/snort_hakemisto >> honeypotX
	2. ls honeypotY/snort_hakemisto >> honeypotY
	3. comm honeypotX honeypotY
IP-osoitteet järjestyksessä	ls snort_hakemisto/ | sort –n
IP:t datamäärän mukaan	du snort_hakemisto/ | sort –n
TCP-yhteydet	ls snort_hakemisto/*/ | grep TCP | wc –l
UDP-yhteydet	ls snort_hakemisto/*/ | grep UDP | wc –l
FTP-yhteydet	ls snort_hakemisto/*/ | grep TCP | grep 21 | wc –l
SSH-yhteydet	ls snort_hakemisto/*/ | grep TCP | grep 22 | wc –l
HTTP-yhteydet	ls snort_hakemisto/*/ | grep TCP | grep 80 | wc –l

ASCII-muotoinen liikenteen tallentaminen pakottaa käyttämään analysoinnissa apuna tekstityökaluja, joiden mahdollisuudet ovat rajoitetut. Tcpdump-tallennusta voidaan selata käyttämällä hyväksi ohjelman omia tehokkaita valitsimia, joilla etsinnät voidaan rajata tarkasti haluttuun paketin ominaisuuteen. Valitsimina voidaan käyttää mm. lähde- tai kohdeportteja, IP-osoitteita ja erilaisia hakusanoja. Haku löytää binääritiedostosta osumat erittäin nopeasti.

Kuviossa 7 on testijakson aikana eri honeypottien kanssa yhteydessä olleiden IP-osoitteiden lukumäärät. VectorLinuxilla IP-osoitteiden lukumäärä on pienempi kuin muilla käyttöjärjestelmillä, vaikka loogisesti määrien pitäisi olla lähempänä

toisiaan. Ero on n. 23 % ja se johtui testijaksolla tapahtuneesta yhteyshäiriöstä. Häiriön vaikutus näkyy myös kuviossa 8.

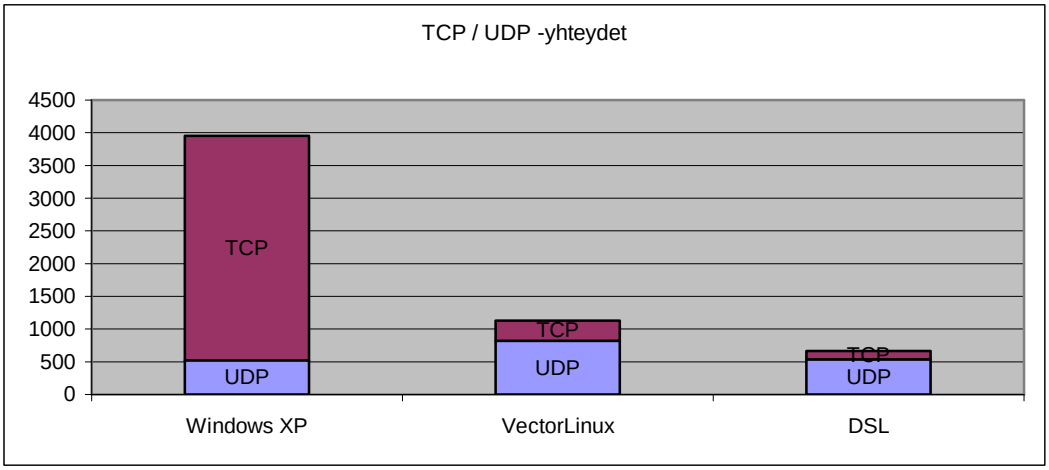


KUVIO 7. Yhteyden muodostaneiden IP-osoitteiden määrät käyttöjärjestelmittäin

IP-osoitteiden joukossa oli 29 osoitetta, joista tuli liikennettä jokaiseen honeypottiin. Kaikki osoitteet olivat kiinalaisoperaattorien hallinnassa. IP-osoitteiden tiedot saadaan esim. Whois-ohjelmalla tai käyttämällä palvelua www.ripe.net-sivulla. Kattava analyysi IP-osoitteiden alkuperistä olisi vaatinut automatisoidun työkalun käyttämistä, mutta sellaista ei tehtävään löytynyt.

Kuviossa 8 on honeypotteihin tulleiden ja niistä lähteneiden TCP- ja UDP-yhteyksien lukumäärät. Yhteyksiä Windows-koneeseen on huomattavasti muita enemmän, koska testijakson aikana sitä kohtaan tehtiin 7 SSH-hyökkäystä, joista pisimmässä oli 1409 yhteyttä. VectorLinuxia kohtaan tehtiin 3 samanlaista hyökkäystä, joista pisimmässä oli 168 yhteyttä. DSL sai vastaansa yhden hyökkäyksen, ja siinä oli vain 22 yhteyttä. Loppuun asti viety TCP-yhteys käsittää kättelyn, datan siirtämisen ja yhteyden purkamisen. Yhden salasanan kokeileminen vaatii SSH-protokollalla vähintään 23 ääripäiden välillä vaihdettua pakettia, ja siihen kuluva aika oli pienimmillään n. 0,5 sekuntia. Muuta TCP-liikennettä tallentui pieni määrä, mutta se oli suuntautunut honeypottien suljettuihin portteihin. Nämä yhteydet alkoivat lähettäjän SYN-paketilla, jonka jälkeen honeypot lähetti takaisin SYN-RST-paketin ja yhteys katkesi.

UDP-liikenteen melko yhtäsuuret määrät selittyvät mm. sillä, että luvuissa on mukana internetoperaattorin kanssa vaihdettu DNS-liikenne porttiin 53. Vector-Linuxin yhteyshäiriö aiheutti sille DNS-liikennettä muita enemmän. Lisäksi tasaista UDP-liikennettä aiheuttaa portteihin 1026 ja 1027 suunnatut paketit, jotka käyttävät Microsoftin messenger-protokollaa ja niiden sisältönä on eräänlainen varoitusviesti. Jos kohdekoneessa on nämä portit auki, ilmestyy työpöydälle ikkuna, jossa varoitetaan Windowsin rekisterin kriittisistä viheistä. Viestin mukaan virheet tulisi korjata välittömästi tietyn internetsivun tarjoaman palvelun avulla. Näitä viestejä liikkuu internetissä säännöllisesti ja ne tulevat monista eri IP-osoitteista. Muu UDP-liikenne sisälsi mm. kuusitoista tallentunutta pakettia Slammer-matoa, joka yrittää järjestelmään sisään portista 1434.



KUVIO 8. Yhteyksien määrän jakaantuminen TCP- ja UDP-protokollien välillä

Taulukossa 4 on listattuna tärkeimpien tutkittujen protokollien yhteyksien lukumäärät. HTTP- ja FTP-yhteyksiä on olemattoman vähän verrattuna SSH-yhteyksien määrään. Vaikka SSH on turvallinen kuljetusprotokolla, on sen eri versioista löydetty lukuisia haavoittuvuuksia, joita yritetään näidenkin tuloksien mukaan hyväksikäyttää. Windows ja VectorLinux käyttivät OpenSSH 4.7, josta ei http://secunia.com mukaan ollut 18.3.2008 mennessä löydetty haavoittuvuuksia. DSL:ssa oli käytössä vanhempi 3.6-versio, mutta sitäkään ei tallennetusta liikenteestä päätellen onnistuttu murtamaan.

TAULUKKO 4. Tutkittujen protokollien osuudet yhteyksistä

	Windows XP	**VectorLinux**	**DSL**
FTP	3	0	4
SSH	3318	230	124
HTTP	67	4	6
TCP	3434	306	158
UDP	521	822	538

Honeypottien lokitiedostoista saa lisätietoja tapahtumista. Verkosta tallennetussa datassa on täydellisimmät tiedot tapahtumista, mutta niiden tulkitseminen ei ole aina aivan suoraviivaista. Käyttöjärjestelmän tallentaman tapahtuman lokilistaukseen kuuluu usein jonkinlainen selvitys, joka saattaa valottaa tapahtumien kulkua. Linux-järjestelmissä lokitiedostot löytyvät /var/log/-hakemistosta. Yksi tärkeimmistä lokitiedostoista on messages, joka tallentaa erilaisia käyttöjärjestelmän tilatietoja käynnistymisestä lähtien. Kuviossa 9 on VectorLinuxin messages-tiedoston tulostusta, josta selviää joukko käyttäjänimiä, joilla hyökkääjä on yrittänyt sisään järjestelmään. Ensimmäisellä rivillä hyökkääjä on käyttänyt järjestelmästä löytyvän käyttäjänimen kanssa väärää salasanaa. Muiden rivien yrityksissä käyttäjänimikin on ollut väärä.

KUVIO 9. Salasanahyökkäys tallentuneena messages-tiedostoon

Windowsin lokeja voi lukea käyttöjärjestelmään ohjelmoidulla työkalulla. Polku Windowsin lokitiedostoihin on:

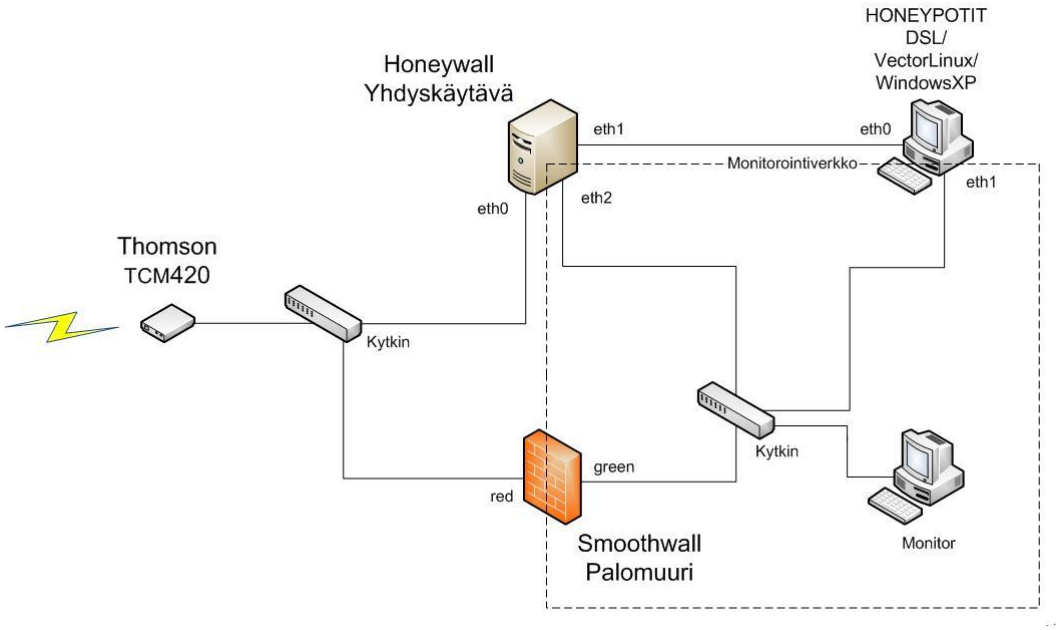
Käynnistä -> Ohjauspaneeli -> Valvontatyökalut -> Tapahtumien valvonta

Tapahtumienvalvonta jaottelee tapahtumia kolmeen tiedostoon, joista ainakin sovellus- ja järjestelmätiedostot sisältävät honeynetin liikenteen tarkkailussa hyödyllistä tietoa. Tapahtumien valvonta tallensi jokaisen Windows-honeypottiin suuntautuneen kirjautumisyrityksen testijakson ajalta. Suurin osa merkinnöistä koskee SSH-tapahtumia, joiden seasta on vaikea löytää muita protokollia koskevia lokeja. Ohjelmassa on kuitenkin hyvät suodatusominaisuudet, joilla muitakin tapahtumia saa nostettua helpommin esiin. Kuviossa 17 on esimerkki Windows-honeypotin tapahtumien valvonnasta.

6.2 Generation II Honeynet

6.2.1 Testiympäristön kuvaus

Generation II honeynet sisälsi Thomson TCM420-kaapelimodeemin, Honeywall-yhdyskäytävän, monitorointiverkkoa suojaavan palomuurin sekä virtuaalisten honeypottien palvelimen. Palomuuri sekä honeypottien palvelin olivat samanlaiset kuin generation I honeynetissä ja niiden rakenne on selvitetty kappaleessa 6.1.2. Kaikilla honeypoteilla on oma julkinen IP-osoite. Tämä mahdollistaa sen, että kaikki honeypotit voivat olla yhtä aikaa toiminnassa ja näkyvissä internettiin. Monitoroivalla työasemalla on suora yhteys Honeywallin tarkkailuliitäntään sekä ulkopuoliseen verkkoon palomuurin green-liitynnän kautta. Kuvio 10 havainnollistaa työssä käytettyä verkkoa.



KUVIO 10. Generation II honeynet-testiympäristö

6.2.2 Honeywallin asentaminen

Honeywall asennetaan CD-Rom-levyltä. Asennustiedoston voi hakea the Honeynet Projectin sivulta https://projects.honeynet.org/honeywall/. Tässä työssä käytetty versio oli roo-1.2.hw-1.iso, mutta tätä niin kuin muitakin Linux-tuotteita päivitetään usein ja versio saattaa vaihtua.

Laitteistovaatimusten mukaan prosessorit i586 Pentium -tasosta ylöspäin ovat tuettuja. Prosessorin puolesta kelpaa siis melko vanhakin laite, mutta muistin määrä tulee olla vähintään 512MB. Honeywallin selaintoiminen pääkäyttöliittymä, jonka kautta saadaan kaikki analysointitulokset helpoimmassa muodossa, vaatii vähintään tämän määrän tietokantojen nopeaa lajittelua varten. Kiintolevyn tulee olla tarpeeksi suuri, jotta kaikki sisä- ja ulkoverkkojen liittymistä tavattu liikenne saadaan talteen. Sovelluksen tekijöiden mukaan 10GB on vähimmäiskoko tuotantoympäristössä. Verkkokortteja tarvitaan kolme kappaletta, ja taulukossa 5 on listattuna testilaitteiston tärkeimmät komponentit.

TAULUKKO 5. Honeywall-laitteisto

Prosessori	Muisti	Kiintolevy	Verkkokortit	Emolevy
AMD 1300+	1GB	Maxtor 80GB	Realtek RTL8129	Soltek SL-75DRV5
			Realtek RTL8129	
			Digital 21x4x tulip	

Asennus tapahtuu CD-Rom-levyltä automaattisesti eikä käyttäjän osallistumista tarvita tässä vaiheessa juuri lainkaan. Asennusohjelma muodostaa kovalevylle oletuksenmukaisen tilanvaraustaulukon, formatoi osiot ja asentaa minimalisoidun ja kovitetun Fedora Core -käyttöjärjestelmän ilman ylimääräisiä osia tai palveluita. Asennus on valmis, kun ruudulla näkyy kehotus painaa enteriä koneen uudelleenkäynnistämiseksi ja CD-Rom työntyy asemasta ulos.

6.2.3 Konfigurointi

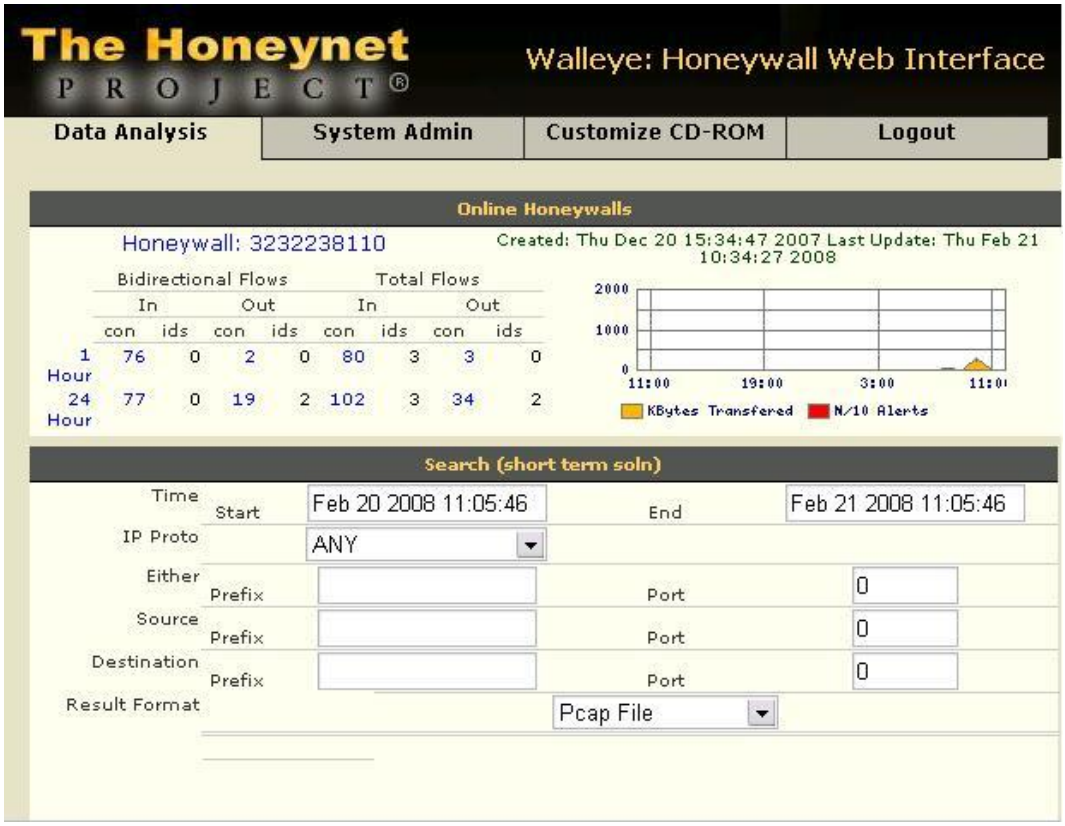
Heti ensiasennuksen jälkeen Honeywall on oletusasetuksilla, eikä sitä voi vielä käyttää mihinkään hyödylliseen. Laitteessa on kaksi oletuskäyttäjää: root järjestelmänvalvojana sekä roo vähäisemmillä oikeuksilla varustettuna. Root-käyttäjä ei voi kirjautua etäyhteyden avulla järjestelmään sisään, joten kirjautumiseen käytetään roo-käyttäjää ja vaihdetaan sisään päästyä itselle järjestelmänvalvojan oikeudet komennolla:

```
[roo@localhost ~]$ su - root
```

Honeywallin konfigurointiohjelma käynnistyy automaattisesti tämän vaihdon jälkeen. Konfigurointiohjelmassa asetetaan kaikki parametrit, joita Honeywall tarvitsee toimiakseen. Näitä ovat mm. verkkoliityntöihin liittyvät parametrit, sallitut portit ja IP-osoitteet, joiden kuuntelu on sallittua, liikenteen kaappaamiseen liittyvät parametrit sekä IP-liikenteen rajoittamisen ja mahdollisten hälytysten asetukset. Täydellinen lista asetuksista on liitteessä 3. Liitteen 3 mukainen konfigurointi voidaan ladata Honeywalliin asennuksen aikana tai myöhemmin SSH-etäyhteyden tai web-hallintaliittymän avulla.

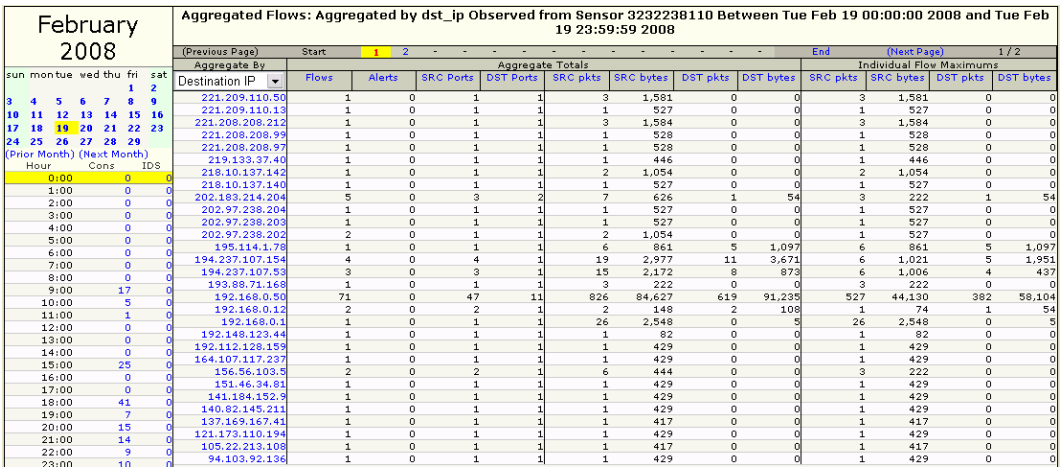
6.2.4 Walleye - Graafinen käyttöliittymä

Web-käyttöliittymään saa yhteyden koneelta, jonka IP-osoite on alkukonfiguroinnissa määritelty sallituksi lähdeosoitteeksi. Osoitteeksi selaimeen kirjoitetaan https://localhost, jossa localhost on Honeywallin hallintaliitynnän osoite. Kirjautumisessa käytetään roo-käyttäjätunnusta, ja ensimmäisellä kerralla oletussalasanaa honey. Kirjautuminen pakottaa vaihtamaan salasanan, joksi kelpaavat roo-1.2-versiossa manuaalin tiedoista poiketen vain 8- tai 9-merkkiset salasanat. Salasanan vaihtamisen jälkeen aukeaa kuvion 11 mukainen hallintaliittymä, jonka etusivulla on pikanäkymä yhdyskäytävän verkkoliityntöjen kautta kulkeneista yhteyksistä ja hakualue, jonka avulla voidaan tehdä yhteyshakuja protokollien, porttinumeroiden tai IP-osoitteiden perusteella. System Admin-sivulla voidaan muokata mm. Honeywallin konfigurointiasetuksia, lokitietokantaa sekä analysoida liikennettä ja hälytyksiä konsolityyppisessä näkymässä. Customize CD-Rom-toiminto ei ole käytettävissä roo-1.2-versiossa.



KUVIO 11. Walleye-hallintaliittymä

Walleye on generation II honeynetin pääasiallisin analyysityökalu. Perusnäytössä voidaan tietoja yhdistellä erilaisten ajanjaksojen perusteella, jolloin kyseisenä aikana otetut yhteydet listataan hallintaikkunaan muiden haluttujen parametrien mukaisesti järjesteltynä. Näitä parametrejä ovat IP-osoitteet ja protokollat kohde- ja lähdesuuntien mukaan eroteltuina. Lisäksi näytettäviä kohteita voidaan suodattaa yhteyden suunnan tai tyypin perusteella. Perusnäytön ominaisuuksiin kuuluu myös mahdollisuus suodattaa yhteyksiä kolmen tärkeän protokollan, TCP:n, UDP:n ja ICMP:n mukaan erillisesti, mikä antaa lisää tarkkuutta ja mahdollisuuksia sopivien yhdistelyjen tekemiseen. Tietyn yhteyden sisältöön päästään valitsemalla haluttu linkki. Honeywallin perusnäyttö listaa kuviossa 12 monipuolisesti yhteyksien kuljettamat tavu- ja pakettimäärät, minkä avulla erottuu selkeästi yhteydet, joissa on liikkunut selvästi muita enemmän dataa.



February 2008

sun	mon	tue	wed	thu	fri	sat
					1	2
3	4	5	6	7	8	9
10	11	12	13	14	15	16
17	18	19	20	21	22	23
24	25	26	27	28	29	

(Prior Month) (Next Month)

Hour	Cons	IDS
0:00	0	0
1:00	0	0
2:00	0	0
3:00	0	0
4:00	0	0
5:00	0	0
6:00	0	0
7:00	0	0
8:00	0	0
9:00	17	0
10:00	5	0
11:00	1	0
12:00	0	0
13:00	0	0
14:00	0	0
15:00	25	0
16:00	0	0
17:00	0	0
18:00	41	0
19:00	7	0
20:00	15	0
21:00	14	0
22:00	9	0
23:00	10	0

Aggregated Flows: Aggregated by dst_ip Observed from Sensor 3232238110 Between Tue Feb 19 00:00:00 2008 and Tue Feb 19 23:59:59 2008

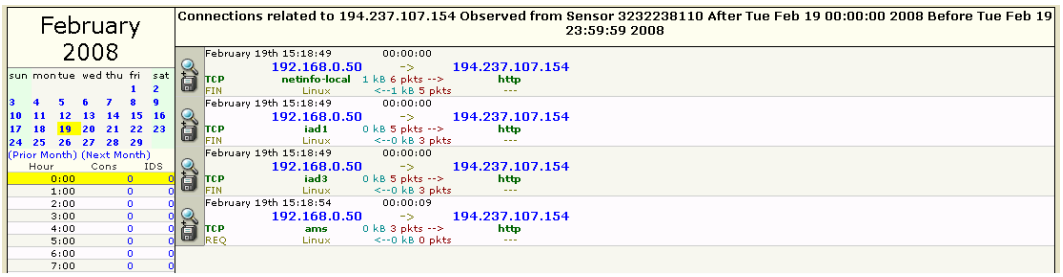
(Previous Page) Start 1 2 - - - - - - - - - End (Next Page) 1 / 2

Aggregate By	Aggregate Totals								Individual Flow Maximums			
Destination IP	Flows	Alerts	SRC Ports	DST Ports	SRC pkts	SRC bytes	DST pkts	DST bytes	SRC pkts	SRC bytes	DST pkts	DST bytes
221.209.110.50	1	0	1	1	3	1,581	0	0	3	1,581	0	0
221.209.110.13	1	0	1	1	1	527	0	0	1	527	0	0
221.208.208.212	1	0	1	1	3	1,584	0	0	3	1,584	0	0
221.208.208.99	1	0	1	1	1	528	0	0	1	528	0	0
221.208.208.97	1	0	1	1	1	528	0	0	1	528	0	0
219.133.37.40	1	0	1	1	1	446	0	0	1	446	0	0
218.10.137.142	1	0	1	1	2	1,054	0	0	2	1,054	0	0
218.10.137.140	1	0	1	1	1	527	0	0	1	527	0	0
202.183.214.204	5	0	3	2	7	626	1	54	3	222	1	54
202.97.238.204	1	0	1	1	1	527	0	0	1	527	0	0
202.97.238.203	1	0	1	1	1	527	0	0	1	527	0	0
202.97.238.202	2	0	1	1	2	1,054	0	0	1	527	0	0
195.114.1.78	1	0	1	1	6	861	5	1,097	6	861	5	1,097
194.237.107.154	4	0	4	1	19	2,977	11	3,671	6	1,021	5	1,951
194.237.107.53	3	0	3	1	15	2,172	8	873	6	1,006	4	437
193.88.71.168	1	0	1	1	3	222	0	0	3	222	0	0
192.168.0.50	71	0	47	11	826	84,627	619	91,235	527	44,130	382	58,104
192.168.0.12	2	0	2	1	2	148	2	108	1	74	1	54
192.168.0.1	1	0	1	1	26	2,548	0	5	26	2,548	0	5
192.148.123.44	1	0	1	1	1	82	0	0	1	82	0	0
192.112.128.159	1	0	1	1	1	429	0	0	1	429	0	0
164.107.117.237	1	0	1	1	1	429	0	0	1	429	0	0
156.56.103.5	2	0	2	1	6	444	0	0	3	222	0	0
151.46.34.81	1	0	1	1	1	429	0	0	1	429	0	0
141.184.152.9	1	0	1	1	1	429	0	0	1	429	0	0
140.82.145.211	1	0	1	1	1	429	0	0	1	429	0	0
137.169.167.41	1	0	1	1	1	417	0	0	1	417	0	0
121.173.110.194	1	0	1	1	1	429	0	0	1	429	0	0
105.22.213.108	1	0	1	1	1	417	0	0	1	417	0	0
94.103.92.136	1	0	1	1	1	429	0	0	1	429	0	0

KUVIO 12. Walleye-perusnäyttö

Kun perusnäytössä valitaan IP-osoite- tai protokollalinkki, päästään katsomaan tarkemmin valitun parametrin rajaamia yhteyksiä honeypotin ja yhteyden vastapäässä olevien päätelaitteiden välillä. Jos linkki on IP-osoite, ovat kaikki haun löydökset kahden tietyn päätelaitteen välisiä määritellyllä aikavälillä sattuneita yhteyksiä. Protokollalinkki antaa tulokseksi monien eri päätelaitteiden välisiä yhteyksiä, joissa yhteyden kohteen porttinumero ja näin ollen yhteyden aikana käytetty protokolla pysyy samana. Kuviossa 12 on suodatusparametrinä käytetty kohteen IP-osoitetta.

Kuviossa 13 on kuvattuna päätelaitteiden välinen yhteyslista. Suurennuslasikuvakkeesta siirrytään tutkimaan yksittäistä kahden päätelaitteen välistä yhteyttä. Tarvittaessa yhteys saadaan purettua kokonaan auki, jolloin se on luettavissa snort-koodattuna bittitasolla. Levykkeen kuvakkeesta voidaan vastaavat tiedot siirtää tietokoneelle luettavaksi muulla tarkoitukseen soveltuvalla ohjelmalla. IP-osoitteista aukeaa linkki ulkopuoliseen whois-tietokantaan, jonka avulla voidaan selvittää osoitteen sijainti maan ja palveluntarjoajan verkkoalueen tarkkuudella. Haitallisia yhteyksiä ottavan IP-osoitteen tiedot voidaan tarvittaessa ilmoittaa puhelimitse tai sähköpostilla suoraan palveluntarjoajan edustajalle.



KUVIO 13. Yhteyksiä kahden päätelaitteen välillä

6.2.5 Apuohjelmien asentaminen honeypotteihin

Sebek haetaan sivulta http://www.honeynet.org/tools/sebek, josta löytyy sopiva ohjelman version jokaista honeypottia varten. Windowsiin tarkoitettu latauspaketti on Sebek-Win32-3.0.4.zip, joka sisältää asennus- ja konfigurointiohjelmien lisäksi lisenssitiedoston sekä asennusohjeen. Asennusohjelma kopioi Sebek-ajurin c:\windows\system32\drivers-kansioon ilman mainittavaa interaktiivisuutta, minkä jälkeen kaikki ohjelman asetukset säädetään erillisen konfigurointiohjelman avulla. Asetuksista tärkeimmät ovat Sebek-pakettien kohteen MAC-osoite, UDP-pakettien kohdeporttinumero sekä Sebek-protokollassa olevan magic-number-kentän arvo. Näitä arvoja käytetään pakettien piilottamiseen verkossa sekä Sebek-liikennettä kuuntelevan serverin paikallistamiseen. Lisäksi valitaan paketit lähettävä ethernet-liityntä sekä kohteen IP-osoite, jonka tulee mahdollisen salakuuntelun hämäämiseksi olla muu kuin kuuntelevan serverin osoite. Asetuksien on vastattava Honeywalliin asetettuja arvoja, jotta paketit olisivat luettavissa.

DSL on pienikokoinen Linux-jakelu ja sen asennuspaketista puuttuu monia Sebekin asennuksen vaatimia osia. DSL:ssä on MYDSL Browser-pakettienhallintajärjestelmä, jonka avulla voidaan puuttuvat osat ladata ja asentaa koneelle. Puuttuvat osat ovat gcc1-withlibs, gcc 9.25 ja kernelsource-2.4.31.dsl.

Kernelin voi hakea myös sivulta www.kernel.org, joka on suositeltavin vaihtoehto. Tässä tapauksessa kernel puretaan ja liitetään järjestelmään komennoilla:

```
tar zxfv linux-2.4.31.tar.gz
ln –s linux-2.4.31 linux
cd /usr/src/linux-2.4.31/
make dep
make symlinks
```

DSL:lle tarkoitettu asennuspaketti on sebek-linux24-3.2.0c.tar.gz. Kun edellä mainitut esivalmistelut on suoritettu, voidaan asennuspaketti purkaa, konfiguroida asennustiedosto ja käynnistää ohjelma. Komennot ovat:

```
tar zxfv sebek-linux24-3.2.0c.tar.gz
cd sebek-linux24-3.2.0c
./configure --with_kernel_source_dir=/usr/src/linux-2.4.31/
make
```

Ennen Sebekin käynnistämistä on suoritettava asetusten konfigurointi sbk_install.sh-tiedostoon. Näiden asetusten avulla honeypot osaa viestiä kuuntelevan palvelimen kanssa sekä salata UDP-paketit muilta verkossa olevilta honeypoteilta. Se osa sbk_install.sh-tiedostoa, jossa on käyttäjän muutettaviksi tarkoitetut parametrit, on listattuna liitteessä 4.

```
vi sbk_install.sh
./sbk_install.sh
```

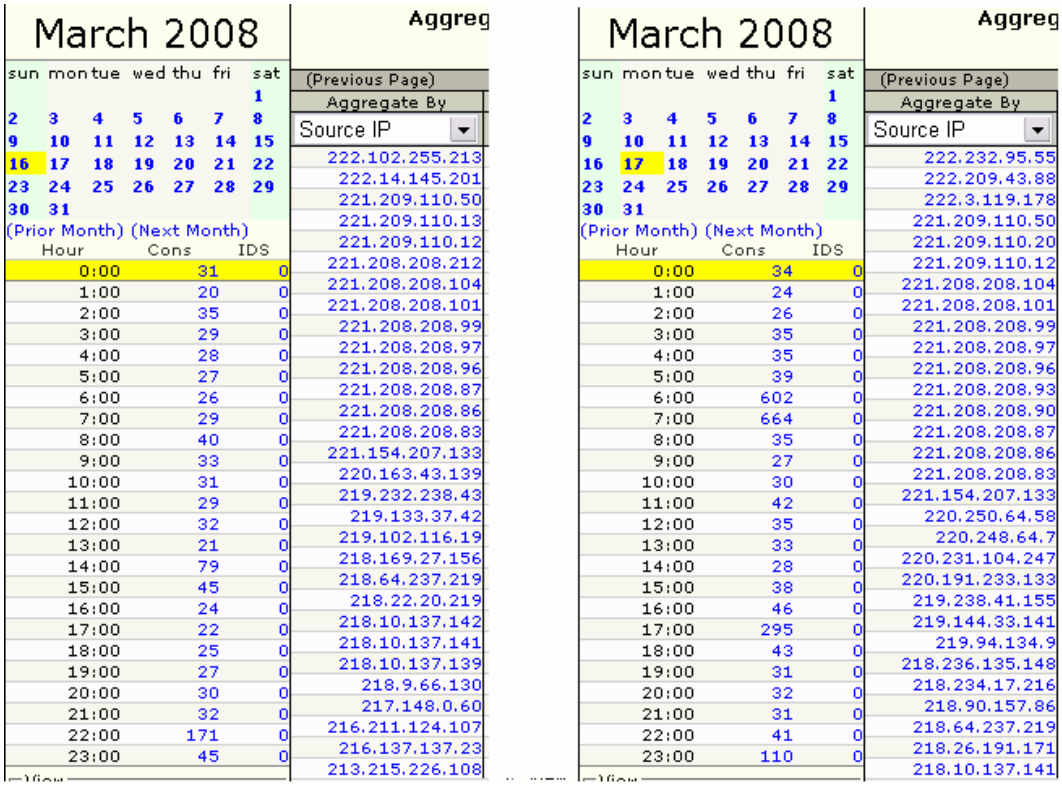
Koska VectorLinux-5.9-Gold käyttää 2.6-pohjaista kerneliä, on sopiva Sebekin versio Linux 2.6 Client 3.2.0b. Jos honeypotteja on samassa verkossa enemmän

kuin yksi, vaatii asennus kernelin lähdekoodin, joka ladataan www.kernel.org-sivulta. Valittavana on joko linux-2.6.22.14.tar.bz2, joka on VectorLinux-5.9-Gold:n pohjalla oleva kernel tai tuoreimman version lähdekoodi ja sen kääntäminen käyttöjärjestelmän ytimeksi. Tässä työssä on käytetty tekohetkellä uusinta Linux-ydintä linux-2.6.24.3. Käännös- ja asennusohje on liitteessä 5.
Sebekin asennuksen vaiheet ovat muuten samat kuin DSL:n tapauksessa.

6.2.6 Liikenteen seuraaminen ja analysointi

Generation II honeynetin liikenteen seuranta on hyvin visuaalista johtuen Walleye-käyttöliittymästä. Näkymää päivitetään minuutin välein, joten tietoja voi lukea melkein reaaliajassa. Sivuilla jokaisesta linkistä saadaan näkyviin erilaisia hakuyhdistelmiä.

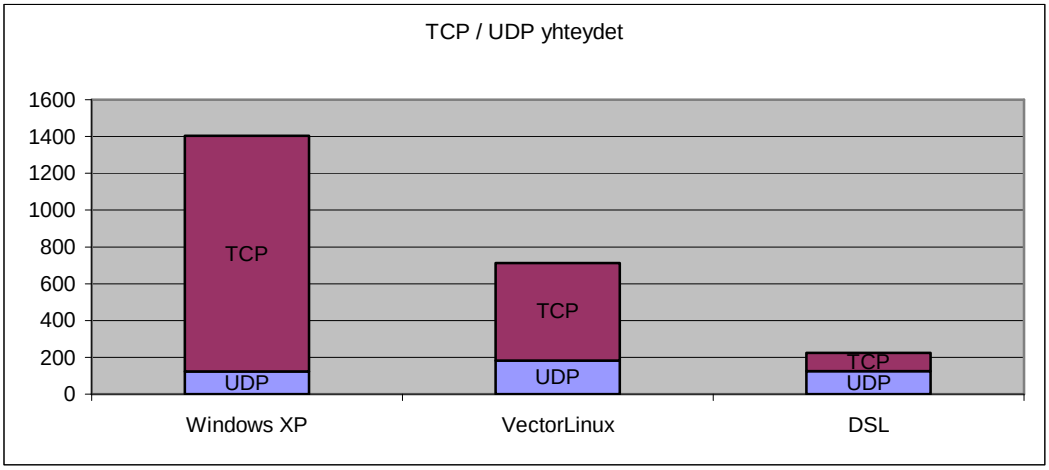
Honeynetin liikennettä tallennettiin kahden vuorokauden ajan niin, että kaikki honeypotit olivat verkossa yhtä aikaa. Kuviossa 14 on testijakson aikana Honeywallin läpi otettujen yhteyksien lukumääriä tunneittain ja IP-osoitteista ensimmäiset kaksi sivua kaikkiaan kymmenestä sivusta. Testijakso oli huomattavasti rauhallisempi verrattuna generation I honeynetin jaksoon. Cons-sarakkeesta nähdään, että jaksolle on sattunut vain viisi sellaista tuntia, joissa yhteyksiä on ollut sata tai enemmän. IDS-sarakkeessa on pelkkiä nollia mikä osoittaa, ettei tunkeutumisenhavainnointijärjestelmä ole havainnut epäilyttäviä yhteyksiä.



KUVIO 14. Generation II honeynetin yhteydet testijakson aikana

Valitsemalla jonkin luvuista Cons-sarakkeessa päästään tutkimaan tarkemmin tämän nimenomaisen tunnin aikana otettuja yhteyksiä. Samalla IP-osoitelista päivittyy näyttämään vain kyseisenä aikana liikennöineet osoitteet. Suuret yhteysmäärät kiinnittävät heti huomiota ja tilanteen tarkasteleminen kannattaakin aloittaa niistä. Esimerkiksi luvun 602 valitseminen paljasti sen sisältävän Windows-honeypottia vastaan tehdyn porttiskannauksen liikenteen sekä hyökkäyksen VectorLinuxin SSH-palvelinta vastaan.

Honeypottien yhteysmäärät voidaan laskea valitsemalla niiden IP-osoite listasta. Osoitelistaan päivittyy silloin vain kyseisen honeypotin kanssa yhteydessä olleet IP-osoitteet. Hakua voidaan tarkentaa esim. sisältämään vain tietyn protokollan liikennettä käyttämällä kuvion 11 alaosassa näkyviä hakukenttiä. IP-osoitteiden laskemiseksi ei ole suoraa tapaa ja osoitteet on tarvittaessa laskettava käsin. Kuviossa 15 on generation II honeynetin TCP- ja UDP-yhteyksien jakautumat.



KUVIO 15. Yhteyksien määrän jakaantuminen TCP- ja UDP-protokollien välillä

Taulukko 6 listaa tärkeimpien tutkittavien protokollien yhteysmäärät. Tulokset ovat samansuuntaiset verrattuna generation I honeynetin tuloksiin. Tälläkin kerralla Windowsiin tuli suurin osa kaikista yhteyksistä. Windowsin ja DSL:n TCP-yhteyksien kokonaismäärä on paljon suurempi kuin tutkittujen protokollien yhteyksien yhteismäärä. Näiden käyttöjärjestelmien avoimia portteja skannattiin testijakson aikana, mikä nostaa paljon yhteyksien yhteismäärää, mutta ei kohdistu sanottavammin tutkittuihin protokolliin. HTTP-yhteydet Windowsiin ja VectorLinuxiin tulivat yhtä aikaa samasta IP-osoitteesta. Vaihe kesti n. 10 minuuttia, jonka aikana kummallekin käyttöjärjestelmälle tuli vuorotellen yhteyksiä eri lähdeporttinumeroilla. Hyökkäyksen tarkoitus jäi epäselväksi, mutta ilmeisesti hyökkääjä oli skannannut osan PHnetin verkosta ja löytänyt honeypottien lähekkäin olleet IP-osoitteet. DSL sijaitsi osoiteavaruuden toisella reunalla, eikä ilmeisesti siksi saanut osaansa yhteyksistä.

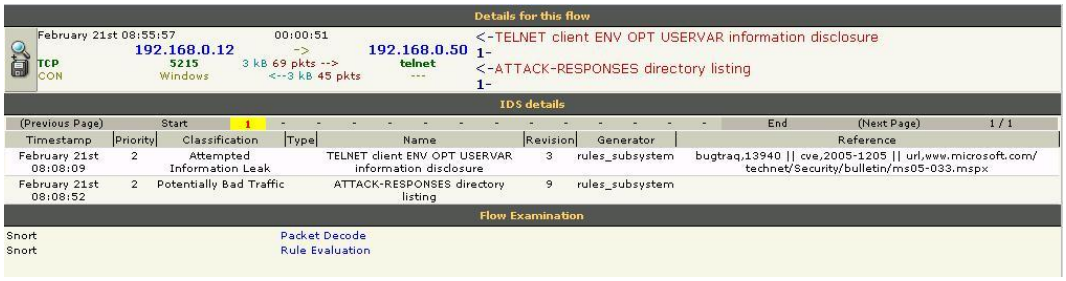
TAULUKKO 6. Tutkittujen protokollien osuudet yhteyksistä

	Windows XP	**VectorLinux**	**DSL**
FTP	3	7	3
SSH	396	369	13
HTTP	75	75	9
TCP	1281	529	99
UDP	123	183	125

Koska honeypotteihin ei tunkeuduttu testijakson aikana, ei Sebekin toimintaa voinut kokea aidossa tilanteessa. Alla on Honeywall-komentorivin tulostus demonstroidusta tunkeutumisesta Windows-honeypotin SSH-palvelimeen käyttäen matkapuhelimeen asennettua SSH-asiakasohjelmaa. Sebek lähettää palvelimelle jokaisen honeypotissa tulostuneen komentorivin, joten sen avulla voidaan lukea sekä hyökkääjän antamat komennot että honeypotin vasteet komentoihin.

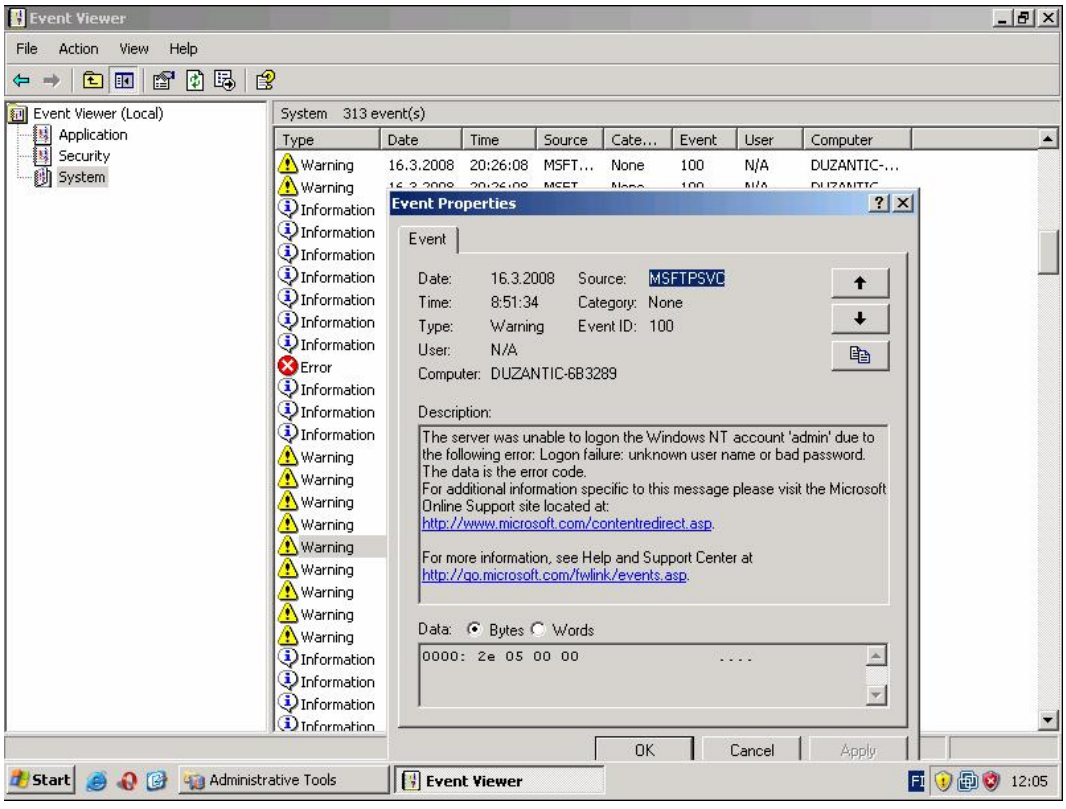
```
monitoring eth1: looking for UDP dst port 1102 81.175.XX.XX  2008/03/15 16:35:54  record 327 received 1 lost 0 (0.00
percent)
[2008-03-15 16:37:50 UID:0 PID:1212 FD:0 INO:0 COM:bash.exe ]# ls
[2008-03-15 16:37:51 UID:0 PID:1212 FD:0 INO:0 COM:bash.exe ]#[ESC]]0;~
[2008-03-15 16:37:51 UID:0 PID:1212 FD:0 INO:0 COM:bash.exe ]#[ESC][32mPavelPloch@duzantic-6b3289
[ESC][33m~[ESC][0m
[2008-03-15 16:38:31 UID:0 PID:1212 FD:0 INO:0 COM:bash.exe ]#$
[2008-03-15 16:38:45 UID:0 PID:1212 FD:0 INO:0 COM:bash.exe ]#$ toimii
[2008-03-15 16:38:46 UID:0 PID:2628 FD:0 INO:0 COM:bash.exe ]#-bash: toimii: command not found
[2008-03-15 16:38:46 UID:0 PID:1212 FD:0 INO:0 COM:bash.exe ]#[ESC]]0;~
[2008-03-15 16:38:46 UID:0 PID:1212 FD:0 INO:0 COM:bash.exe ]#[ESC][32mPavelPloch@duzantic-6b3289
[ESC][33m~[ESC][0m
[2008-03-15 16:38:58 UID:0 PID:1212 FD:0 INO:0 COM:bash.exe ]#$ exit
[2008-03-15 16:38:58 UID:0 PID:1212 FD:0 INO:0 COM:bash.exe ]#logout
```

Ne yhteydet, joissa tunkeutumisenhavainnointijärjestelmä huomaa olevan mahdollisesti haitallista sisältöä, summataan etusivun IDS-sarakkeessa. Kuviossa 16 on esimerkki tällaisesta yhteydestä tarkemmin tarkasteltuna. Vasemmassa yläkulmassa on yhteyden yleiset tiedot, jotka esitetään kaikkien muidenkin yhteyksien tallennuksissa. Tiedoista ilmenee mm. yhteyden katkaisutapa, yhteyksien vastapäiden käyttöjärjestelmät, siirretyn datan määrään kumpaankin suuntaan sekä protokolla ja porttinumerot, joita yhteyden aikana on käytetty. Lohkon tietojen perusteella saa nopeasti käsityksen yhteyden ominaisuuksista. Kaikki muut alueet ilmaisevat tunkeutumisenestojärjestelmän lisätietoja. Tiedoista selviää mm. hyökkäyksen tyyppi, selvitettyjen tapahtumien vaiheet ja internetsivun osoite, josta saa lisätietoja kyseisestä hyökkäystyypistä. Valitsemalla alhaalta packet decode-linkin saadaan yhteyden tiedot avattua kokonaan.



KUVIO 16. Telnet-hyökkäys

Käyttöjärjestelmien lokitiedostoja voi käyttää hyväksi tarvittaessa lisätietoja Honeywallin omien työkalujen antamien tietojen täydennykseksi. Käyttöjärjestelmät eivät eroa eri honeynet-malleissa toisistaan, joten tietojen etsiminenkin tapahtuu niissä samoilla tavoin. Kuviossa 17 on lokilistaus, joka on saatu Windows-honeypotista FTP-yhteysyrityksen seurauksena. Esiin nostetussa varoitusviestissä on ilmoitus sisäänkirjautumisyrityksestä väärällä käyttäjänimellä ja salasanalla.



KUVIO 17. FTP-yhteysyritys Windows-honeypottiin

6.3 Honeynettien vertailu

Eri käyttöjärjestelmiä edustavien honeypottien vaikutus tallennetun liikenteen määrään on selvästi nähtävissä tuloksista. Kummallakin honeynet-mallilla saatiin Windows-honeypottia kohtaan selvästi suurimmat määrät liikennettä. Linux-käyttöjärjestelmillä VectorLinux sai yhteyksiä selvästi DSL:a enemmän, mikä oli hieman odotusten vastaista. Oli odotettavissa, että vanhempi kernel-versio olisi hyökkääjille kiinnostavampi ja sille olisi tullut enemmän yhteyksiä. Kummankin honeynetin tulos oli kuitenkin samansuuntainen Linuxien tapauksessa ja varmasti kuvaa todellisuutta siinä suhteessa oikein. TCP- ja UDP-protokollien lukemia tarkastellen honeynetit käyttäytyivät samalla tavalla, mutta testeissä tutkittujen sovellusprotokollien lukemat paljastavat taustalta erilaista käyttäytymistä. Windows ja VectorLinux saivat generation II honeynetissä melkein tarkalleen samat tulokset. Näissä ei kuitenkaan ole porttiskannauksen seurauksena syntyneitä yhteyksiä. Tulokset vaikuttavat nyt hieman sattumanvaraisilta ja olisivat ehkä tarvinneet pidemmän tallennusjakson, jonka aikana suurempi yhteyksien määrä olisi tasoittanut tuloksia.

Tämäntyyppiset honeynetit keräävät enimmäkseen liikennettä, jonka tarkoitus on kartoittaa haavoittuvuuksia sisältäviä tietokoneita ja sen jälkeen hyökätä haavoittuvuuksien kimppuun käyttämällä jotakin automaattista työkalua. Testijaksojen aikana tapahtuneet salasanahyökkäykset ja porttiskannaukset tulokset puhuvat tämän käsityksen puolesta. Muihin kuin testissä käytetyihin portteihin suuntautuneen TCP- tai UDP-liikenteen eritteleminen ei ole kovinkaan helppoa sen pienen määrän ja satunnaisuuden takia. Varsinkin generation I honeynetin suppeampi työkaluvalikoima vaikeuttaa tehtävää.

Honeynetissä suoritetun liikenteen tarkkailun ja tallentamisen on tarkoitus tapahtua siten, että vaikutettaisiin mahdollisimman vähän itse tapahtumiin verkossa. Tarkkailun olemassaolo vaikuttaa tuloksiin vain silloin, jos hyökkääjä huomaa olevansa tarkkailun kohteena. Kummassakaan honeynet-mallissa hyökkääjä ei voi suoraan havaita merkkejä tarkkailusta. Hyökkääjän on ensin onnistuttava tunkeutumaan johonkin honeypoteista ja havaittava siellä epäilyttäviä poikkeuksia normaaliin konfiguraatioon verrattuna. Toinen vaihtoehto on suorittaa murretusta

honeypotista käsin porttiskannaus muuhun lähiverkkoon ja sillä tavalla havaita tarkkailevan tietokoneen läsnäolo. Hyökkääjä voi generation I honeynetissä havaita tällä tavalla kuuntelevan tietokoneen, mutta uudemmassa mallissa ei ole kuuntelevaa tietokonetta ja Honeywallin verkkoliitynnät toimivat tasolla, jota hyökkääjän on mahdoton havaita omasta verkostaan käsin. Generation II honeynet on tästä syystä paremmin piilotettu ja sillä on mahdollisuudet säilyttää pidempään toimintaedellytyksensä hyökkääjiä vastaan.

Eräs tärkeä peruste arvioitaessa honeynetin hyödyllisyyttä on helppous, jolla kerätystä datasta saadaan eroteltua tärkeät tiedot merkityksettömän datan joukosta. Yksi suurimmista eroista honeynet-toteutusten välillä oli juuri tietojen esitystavassa. Generation I honeynetissä oli pelkästään komentoriviltä käytettäviä ohjelmia tietojen lukemiseen. Ohjelmat pystyvät suodattamaan dataa erittäin hyvin erilaisia lisävalitsimia käyttämällä, mutta oikeiden valitsimien löytäminen voi olla aluksi vaikeaa. Tietojen suodatuksen onnistuminen juuri tarkoitetulla tavalla on varmistettava suodattamalla tiedot myös jollain muulla tavalla. Voi olla vaikeata varmistaa yksityiskohtaisten tietojen esim. lukumäärien tarkkuus.

Datamäärän kasvaessa voi komentoriviltä tapahtuva tietojen analysointi myösmuuttua vaikeaksi. Jos verkossa on paljon honeypotteja tai liikenne vilkasta muusta syystä, voi kerätyn datan määrä kasvaa hankalan suureksi. Tässä työssä honeypoteista tallennettiin vain kahden perättäisen vuorokauden data, mutta generation I honeynetin tallennushakemiston sisällön koko oli jakson lopussa kasvanut 105 MB. Windows-honeypotin aiheuttaman datan määrä oli yli puolet kokonaismäärästä ja aiheutti sen, että Cat-ohjelma ei kyennyt käsittelemään niin suurta hakemistoa. Cat oli käytössä Snortilla kerätyn datan selaamista varten ja ongelma helpottui jakamalla tallennushakemisto kahtia. Tcpdump-ohjelmalla vastaavia ongelmia ei ollut, johon oli syynä ainakin se, että tallennusten tuloksena syntyneet tiedostot olivat paljon pienempiä.

Generation I honeynet kerää ja tallentaa dataa raakamuodossa. Datan analysoijan on oltava hyvin perillä erilaisten protokollien toiminnasta ja otsikkokenttien sisältämien tietojen merkityksestä yms. voidakseen ymmärtää verkon tapahtumia käytettävissä olleilla työkaluilla. Linux-käyttöjärjestelmiin on saatavilla graafisia työ-

kaluja, joilla liikenteen seuraaminen on hieman visuaalisempaa, mutta verkon kuuntelijana olleeseen DSL:n niitä ei käyttöjärjestelmän omalla pakettienhallinnalla voinut asentaa.

Honeywall-yhdyskäytävä on suunniteltu tuottamaan havainnollista dataa verkon liikenteestä ja hoitamaan luotettavasti honeynetin liikenteen kontrollointia edeltä käsin asetettujen sääntöjen mukaisesti. Honeypotteihin asennettujen apuohjelmien avulla Honeywall kykenee suorittamaan kattavasti kaikki toimet, joita tarkkailtavan verkon hallinta vaatii. Honeywalliin on asennettu valmiiksi kaikki tarpeelliset ohjelmat ja niiden hienosäätö voidaan suorittaa selkeän web-käyttöliittymän avulla, mikä pienentää epäonnistuneesta konfiguroinnista johtuvan tietoturvan vaarantamisen mahdollisuutta. Esimerkiksi ulospäin suuntautuvan liikenteen rajoittaminen sallimaan vain tietyn määrän yhteyksiä tietyssä ajassa säädetään Honeywallissa yhdellä parametrilla. Tavanomaisen Linux-palomuurin vastaava toiminnan muuttaminen vaatii useiden rivien varovaisen muokkaamisen palomuurin konfigurointitiedostossa.

7 YHTEENVETO

Testijaksot antoivat selvän käsityksen verkossa liikkuneen liikenteen määrästä. Jo kahdessa vuorokaudessa yhteyksien määrä kasvoi tuhansiin kun kohteina olivat tietokoneet, joiden olemassaoloa ei mainostettu millään tavalla internetin suuntaan. Yksittäinenkin verkkoon liitetty tietokone löytyy siis todella nopeasti, ja sitä voidaan mahdollisesti alkaa hyväksikäyttämään välittömästi haavoittuvuuden löytymisen jälkeen. Kuitenkin muiden kuin testissä tutkittujen protokollien pakettien tarkastelu osoitti, että tietokoneen suojaus paranee pelkästään pitämällä tarpeettomat portit suljettuina. Palomuurin käyttäminen suojaisi tietokoneita hyvin testissä esiintyneitä hyökkäyksiä vastaan. Testissä saatujen tuloksien perusteella hyökkääjät pystyvät selvittämään vastapäisestä tietokoneesta avoimet portit ja käyttöjärjestelmän, päätellen liikenteen painottumisesta Windows-honeypotin suuntaan kummallakin testikerralla.

Työ sisälsi paljon erilaisia käyttöjärjestelmien ja ohjelmien asennuksia. Käyttöjärjestelmän asentaminen on aikaa vievä operaatio, jos käytössä on yhteensopimattomia laitteita. Tässä työssä tarvittiin paljon verkkokortteja, ja niistä aiheutui aluksi ongelmia, jotka ratkesivat kuitenkin vaihtamalla sopimattomat kortit toisenlaisiin. Kaikki ohjelmat saatiin sen jälkeen asennettua, ja ne toimivat käyttöjärjestelmissä oikein. Ainoa poikkeus oli VectorLinuxiin asennettu Sebek-ohjelma, jonka lähettämää tulostetta ei saatu näkymään Honeywall-yhdyskäytävässä. Tämän ongelman olisi voinut ratkaista käyttämällä honeypottina VectorLinuxin sijaan jotain ohjelman tekijöiden mainitsemaa testattua käyttöjärjestelmää. Asian tultua ilmi oli työ edistynyt kuitenkin jo niin pitkälle, että vaihto olisi aiheuttanut lisätyötä kohtuuttoman paljon saavutettuun hyötyyn nähden.

Testeistä saadut otettujen yhteyksien lukumäärät antavat selvän merkin siitä, että Windows on hyökkääjille halutumpi kohde kumpaankin testissä olleeseen Linux-järjestelmään verrattuna. Toisaalta suurin osa hyökkäyksistä kohdistui juuri SSH-palvelinta vastaan, mikä ei ole Windowsin oma ohjelma, vaan peräisin avoimen lähdekoodin puolelta. Tämä yhdistelmä ei ole muita järjestelmiä haavoittuvaisem-

pi, mutta hyökkääjät etsivät todennäköisesti heikkoja salasanoja, joita löytyy helpommin Windows-järjestelmistä.

Tehtyjen testijaksojen perusteella kumpaakin verkkotyyppiä voidaan käyttää apuna selvitettäessä verkkoihin suuntautuvan liikenteen alkuperää ja tarkoitusta. Verkkojen toiminta on hyvin samankaltaista toisiinsa verrattuna, ja erot löytyvät lähinnä asennuksen ja konfiguroinnin helppouden sekä saatujen tulosten analysoinnin selkeyden kautta. Konfiguroinnin onnistuminen on honeynetin käyttämisessä avainasemassa, koska se väärin suoritettuna voi jättää verkon laitteet suojattomiksi ja edesauttaa hyökkääjää tekemään lisää vahinkoa muille kohteille. Honeywallin käyttöliittymä on toteutettu selkeästi ja ottaen erityisesti huomioon liikenteen rajoittamisen vaatimukset. Vastaavanlaisen toiminnan aikaansaaminen konfiguroimalla standardia Linux-palomuuria vaatii syvällistä paneutumista palomuurisääntöjen suunnittelemiseen ja niiden kirjoittamisen onnistumista virheettömästi.

Tietoturvan taso on toinen tärkeä eroavaisuus testissä olleiden toteutusten välillä. Generation I honeynetiin sijoitetun kuuntelevan työaseman havaitseminen on aivan mahdollista honeypotin murtamisen jälkeen. Hyökkääjä pystyy saamaan selville kaikki verkkotasolla toimivat laitteet itsensä ja honeypotin väliltä. Honeywallia ei kuitenkaan voida havaita samalla tavalla, mikä on suuri etu tietoturvan kannalta.

Honeynettiä voi ajatella käytettäväksi yrityksissä, jotka haluavat selvittää heidän tietoverkoihinsa suuntautuvan liikenteen laatua ja tarkoitusta. Yrityksen honeynetissä havaittu aktiivisuus on merkki mahdollisista tuotantoverkkoon kohdistuneista uhkista, jotka voisivat peittyä sallitun verkkoliikenteen joukkoon. Yrityksen tietoturva-asiantuntijoille on hyödyllistä päästä perille heikommin suojattujen honeypottien murtamiseen käytettävistä tekniikoista voidakseen suojata tuotantoverkkonsa paremmin näitä tekniikoita vastaan.

Honeynettiä voidaan käyttää hyväksi tietoturva-alan tutkimuksessa. Tietoturvayhtiöt saavat honeynetin avulla keskimääräisen kuvan internetissä liikkuvasta haitallisesta liikenteestä. Suuria yrityksiä tai muita kiinnostavia kohteita vastaan teh-

dyistä hyökkäyksistä tällainen tiedonkeruu ei kuitenkaan kerro mitään, koska tällöin käytetään tarkasti valittuja IP-osoitteita.

Myös tietoturvasta kiinnostunut yksityishenkilö voi aivan hyvin käyttää honeynettiä parantaakseen oman verkon tietoturvaa ja lisätäkseen oman tietoturvaymmärryksensä tasoa. Kustannukset pysyvät minimaalisella tasolla, koska verkon toteutukseen riittää tavanomaiset laitteet ja ohjelmistot ovat vapaan lähdekoodin tuotteita ja näin ollen ilmaiseksi saatavilla.

Työssä käytetyntyyppiset verkot saavat vastaansa enimmäkseen sattumanvaraisia hyökkäyksiä, joiden tarkoitus on mm. kerätä murrettuja tietokoneita hyökkääjän keskitettyyn hallintaan. Tällä tavalla ei kuitenkaan voida selvittää esim. internetpalvelimien sisältämää hyökkäävää toimintaa, joka vaatisi asiakkaan puolelta otetun yhteyden palvelimeen. The Honeynet Projectilla on kuitenkin käytössään monenlaisia työkaluja ja tähänkin tarkoitukseen löytyy apuohjelmia, joilla voidaan tehdä tutkimusta joko yksityiseen käyttöön tai lähettäen saatuja tuloksia takaisin yhteisölle yhteiseksi tiedoksi. The Honeynet Project on myös laajenemassa hiljalleen maailmanlaajuiseksi tarkkailuverkkojen verkostoksi, jonka tarkoitus on luoda tarkentuva ja nopeasti päivittyvä kuva koko maailman tietoturvatilanteesta. Rikollisten käyttämien uusien hyökkäystapojen löytäminen ja selvittäminen on ehkä tärkein honeyneteistä saatava hyöty ja laajentamalla toimintaa laajenevat myös mahdollisuudet tähän. Rikolliset tulevat olemaan aina askeleen edellä tietoturvaa, mutta sen askeleen pituutta yritetään lyhentää näillä keinoilla.

LÄHTEET

The Honeynet Project, 2004. Know your enemy: learning about security threats (second edition). ISBN 0-321-16646-9

Anonymous, 2002. Hakkerin käsikirja. Edita Publishing OY IT-Press, Helsinki

Durham, J. 2002. Linux Sertifikaatti. ISBN 951-826-652-2

Hagen, S. 2002. IPv6 Essentials. ISBN 0-596-00125-8

Järvinen, P. 2006. Paranna tietoturvaasi. Docendo Finland OY, Jyväskylä

Kerttula, E. 1998. Tietoverkkojen tietoturva. Oy Edita Ab, Helsinki

Scambray, J, McClure, S, Kurtz, G. 2001. Hakkeroinnin Torjunta. ISBN 952-14-0436-1

Zwicky, E.D, Cooper, C, Chapman, D.B. 2001. Internet palomuurien rakentaminen. Satku -Kauppakaari, Helsinki

Commtouch. 2007. Q1 2007 Spam Trends: Botnets continue sending devious spam. [viitattu 9.3.2008] [verkkodokumentti]
saatavissa:
http://www.commtouch.com/documents/Commtouch_2007_Q1_Spam_Trends.pdf

Decker, M. CERT, 1997. Security of the Internet
[viitattu 23.1.2008][verkkodokumentti]
Saatavissa: http://www.cert.org/encyc_article/tocencyc.html

Gizmo. 2006. Rootkit detection and removal. [viitattu 29.1.2008]
[verkkodokumentti]
saatavissa: http://www.pcsupportadvisor.com/rootkits.htm

Helenius, M. 2006. Virusten torjunta. [viitattu 29.1.2008]
[verkkodokumentti]
saatavissa: http://www.cs.uta.fi/hot/luennot_2_ja_3_2006.pdf

Joutsi, J. 2006. WWW-Pähkinänkuoressa. [viitattu 3.4.2008] [verkkodokumentti]
saatavissa: http://www.joutsi.com/palomuuri2.png

Kajantie, S. Verkkorikollisuus tietoturvauhkana. 2006.
[viitattu 26.1.2008] [verkkodokumentti]
saatavissa: http://www.huoltovarmuus.fi/documents/7/TIVA-seminaari_ES_09-10-2006_Kajantie.pdf

Kajava, J. Henkilöturvallisuus osana yrityksen tietoturvaa. 2003
[viitattu 10.3.2008][verkkodokumentti]
saatavissa: http://www.ulapland.fi/files/2004011415044.pdf

Laurio, J-M. 2007. Tietoviikko. Tallinnan mellakat poikivat palvelunestohyökkäyksiä.
[viitattu 27.1.2008][verkkodokumentti]
saatavissa: http://www.tietoviikko.fi/doc.te?f_id/1162280

Lemos, R. , 2007. SecurityFocus. Rootkits headed for bios [viitattu 27.1.2008]
[verkkodokumentti]
saatavissa: http://www.securityfocus.com/news/11372

Microsoft. 2008. Tietokoneen Päivittäminen: Usein kysytyt kysymykset.
[viitattu 3.4.2008] [verkkodokumentti]
saatavissa: http://www.microsoft.com/finland/athome/security/update/faq.mspx

Morris, J. 2002. LIBIPQ. [viitattu 8.2.2008]
[verkkodokumentti]
saatavissa: http://www.cs.princeton.edu/~nakao/libipq.htm

Mustonen, E. Monipalveluverkot Tietoturvauhkia ja Ratkaisuja. 2006.
[viitattu 26.1.2008] [verkkodokumentti]
saatavissa: http://huoltovarmuus.fi/documents/7/Mustonen.pdf

Nebulan ADSL-liittymät IPv6 aikaan
[viitattu 11.3.2008][verkkodokumentti]
saatavissa: http://sektori.com/uutiset/8009/nebulan

Panda Security. 2008. Types of malware. [viitattu 29.1.2008]
[verkkodokumentti]
saatavissa: http://www.pandasecurity.com/homeusers/security-info/types-malware

Secunia. 2008. Vulnerability Report [viitattu 3.4.2008] [verkkodokumentti]
saatavissa: http://secunia.com

The New Zealand Honeynet Project. 2007. Capture BAT (Behavioral Analysis Tool) for applications and documents [viitattu 8.3.2008]
[verkkodokumentti]
saatavissa: http://newzealand.honeynet.org/cbatabout.html

TSK. 2004. Sanastokeskus TSK ry. Tiivis tietoturvasanasto. [viitattu 29.1.2008]
[verkkodokumentti]
saatavissa: http://www.tsk.fi/fi/info/TiivisTietoturvasanasto.pdf

Zalewski, M. 2006. The new P0f. [viitattu 13.2.2008] [verkkodokumentti]
saatavissa: lcamtuf.coredump.cx/p0f.shtml

LIITE 1

CentOS asennus

CD found	Skip
Welcome to CentOS_Server_CD	Next
Language Selection	English -> Next
Keyboard Configuration	Finnish -> Next
Disk Partitioning Setup	Automatically partition -> Next
Warning	Yes
Automatic Partition	Remove all partitions on this system -> Next
Warning	Yes
Disk Setup	Next
Boot Loader Configuration	Next
Network Configuration	Kaikki kortit aktiivisiksi -> Next
Firewall Configuration	Enable firewall -> Next
Additional Language Support	English -> Next
Time Zone Selection	Europe/Helsinki -> Next
Set Root Password	******** -> ******** -> Next
Package Installion Defaults	Customize Software Packages to be installed -> Next
Package Group Selection	Minimal -> Next
About to Install	Next

Komennot, joilla poistetaan tarpeettomat palvelut järjestelmästä asennuksen ja uudelleenkäynnistyksen jälkeen

```
[root]# chkconfig acpid off
[root]# chkconfig atd off
[root]# chkconfig autofs off
[root]# chkconfig cups off
[root]# chkconfig gpm off
[root]# chkconfig isdn off
[root]# chkconfig mdmonitor off
[root]# chkconfig netfs off
[root]# chkconfig nfslock off
[root]# chkconfig openibd off
[root]# chkconfig pcmcia off
[root]# chkconfig portmap off
[root]# chkconfig rawdevices off
[root]# chkconfig rpcgssd off
[root]# chkconfig rpcidmapd off
[root]# chkconfig smartd off
```

/etc/inittab-tiedostosta poistetaan tarpeettomat virtuaalikonsolit käytöstä

```
[root]# vi /etc/inittab
```

LIITE 1 (jatkuu)

```
# Run gettys in standard runlevels
1:2345:respawn:/sbin/mingetty tty1
#2:2345:respawn:/sbin/mingetty tty2
#3:2345:respawn:/sbin/mingetty tty3
#4:2345:respawn:/sbin/mingetty tty4
#5:2345:respawn:/sbin/mingetty tty5
#6:2345:respawn:/sbin/mingetty tty6
```

```
[root]# reboot
```

uudelleenkäynnistyksen jälkeen päivitetään järjestelmä

```
[root]# yum update
```

LIITE 2

Virtuaalisen kovalevyn luominen

File -> New -> Virtual Machine

Welcome to New Virtual Machine Wizard	Next
Select the Appropriate Configuration	Custom -> Next
Select a Guest Operating System	

Windows XP:lle	Microsoft Windows ja Windows XP Professional
DSL:lle	Linux ja Other Linux 2.4.x kernel
VectorLinuxille	Linux ja Other Linux 2.6.x kernel

Name the Virtual Machine	Valitaan sopiva nimi -> Next
Set Access Rights	Jos ei ole muita käyttäjiä -> Next
Startup/Shutdown Options	Next
Processor Configuration	Next
Memory for the Virtual Machine	Valitaan suositusten mukaan -> Next
Network Type	Use bridged networking -> Next
Select I/O Adapter Types	Next
Select a Disk	Create a new virtual disk -> Next
Select a Disk Type	IDE (kaikille) -> Next
Specify Disk Capacity	Disk size (GB) mahd. pieni -> Next
SpecifyDisk File	Valitaan sopiva nimi -> Finnish

LIITE 3

Honeywall konfigurointiparametrit

xxx -parametrit konfiguriotava toteutettavan verkon mukaisesti

HwHOSTNAME = **localhost**
HwLAN_BCAST_ADDRESS = **xxx**
HwSUMNET = **xxx**
HwUDPRATE = **19**
HwSEBEK_DST_IP = **xxx**
HwROACHMOTEL_ENABLE = **no**
HwALERT = **yes**
HwRULE_DAY = **sat**
HwINET_IFACE = **eth0**
HwQUEUE = **yes**
HwTIME_SVR =
HwMANAGE_NETMASK = **xxx**
HwSEBEK_DST_PORT = **1102**
HwSEBEK_LOG = **yes**
HwHWPARMOPTS =
HwFWFENCE = **/etc/fencelist.txt**
HwSCALE = **hour**
HwALLOWED_TCP_IN = **443**
HwNICMODLIST =
HwMANAGE_IP = **xxx**
HwFWBLACK = **/etc/blacklist.txt**
HwSSHD_PORT = **22**
HwHONEYWALL_RUN = **yes**
HwLAN_IFACE = **eth1**
HwBPF_DISABLE = **no**
HwRULE_ENABLE = **no**
HwMANAGE_GATEWAY = **xxx**
HwDOMAIN = **localdomain**
HwMANAGE_IFACE = **eth2**
HwICMPRATE = **16**
HwALERT_EMAIL =
HwDNS_SVRS = **xxx**
HwOTHERRATE = **37**
HwFWWHITE = **/etc/whitelist.txt**
HwMANAGE_DNS = **xxx**
HwPCAPDAYS = **120**
HwOINKCODE = **xxx**
HwRESTRICT = **yes**
HwSNORT_RESTART = **no**
HwMANAGE_STARTUP = **yes**
HwDBDAYS = **180**
HwSEBEK = **yes**

LIITE 3 (jatkuu)

HwSSHD_REMOTE_ROOT_LOGIN = **yes**
HwHPOT_PUBLIC_IP = **xxx**
HwHEADLESS = **no**
HwWALLEYE = **yes**
HwALLOWED_UDP_OUT = **53 123**
HwFENCELIST_ENABLE = **no**
HwALLOWED_TCP_OUT = **22 43 80 443**
HwHFLOW_DB = **1.1**
HwDNS_HOST = **xxx**
HwSEBEK_FATE = **ACCEPT**
HwLAN_IP_RANGE = **xxx**
HwBWLIST_ENABLE = **no**
HwTCPRATE = **25**
HwRULE_HOUR = **3**
HwSWAP_CAPSLOCK_CONTROL = **no**
HwMANAGER = **xxx**
HwMANAGE_DIALOG = **yes**

LIITE 4

sbk_install.sh-tiedoston konfigurointi

xxx -parametrit konfiguriotava toteutettavan verkon mukaisesti

#----- SEBEK LINUX CLIENT INSTALL SCRIPT ----------

FILTER="**./filter.txt**"

INTERFACE="**eth0**"

DESTINATION_IP=**xxx** (ei saa olla sebek-palvelimen osoite)

DESTINATION_MAC=**xxx** (honeywallin eth1:n mac)

SOURCE_PORT=**1101**

DESTINATION_PORT=**1102**

MAGIC_VAL=**1524**

KEYSTROKE_ONLY=**0**

SOCKET_TRACKING=**1**

TESTING=**0**

MODULE_NAME=

WRITE_TRACKING=**1**

#----- !! END OF USER CONFIGURABLE OPTIONS !!----------

LIITE 5

Kernelin päivittäminen VectorLinuxissa

Download kernel source, linux-2.6.x.tar.bz2 from www.kernel.org

tar xvjpf linux-2.6.x.tar.bz2 -C /usr/src/

ln -sf /usr/src/linux-2.6.x /usr/src/linux

1 - "cd /usr/src/linux-2.6.x"

2 - "make mrproper"

3 - "cp /boot/config ./.config" (please notice the .'s carefully, there is no .config in /boot and the file config will be useless to your kernel...it needs to be .config)

4 - "make menuconfig" or "make xconfig" or "make gconfig" -whichever you prefer,
(I like gconfig because its a snazzy gui that has little notes on the options and tells you what they mean and if you might want them or not)

5 - now change the options you want to change

6 - exit and it will save the config. (or hit save and exit in gconfig)

7 - "make bzImage && make modules && make modules_install" - the && means that
the previous command must finish successfully before going on with the next.

Once this completes you you will have bzImage in /usr/src/linux-2.6.x/arch/i386/boot and in /usr/src/linux-2.6.x/ you will have System.map.

8 - "cp /usr/src/linux-2.6.x/System.map /boot/System.map-2.6.x
9 - "cp /usr/src/linux-2.6.x/arch/i386/boot/bzImage /boot/bzImage-2.6.x

now in Vectorlinux /boot/System.map is a symlink which points to a real system.map file, if you ls -al System.map you will see that it point to your old System.map

10 - "rm System.map" removing the symlink

11 - "ln -sf /boot/System.map-2.6.x System.map" which replaces it.

LIITE 5 (jatkuu)

now we need to edit /etc/lilo.conf with your favorite editor...
if you're afraid of vi then you can always "nedit /etc/lilo.conf"
but I prefer vi.
You should have an entry that looks "something" like this:

image=/boot/vmlinuz
label=linux
root=/dev/hda2
read-only

12 - copy all four lines and paste them with a one line break below the original.

13 - in the new entry you just made to /etc/lilo.conf change the line image=/boot/vmlinuz to
image=/boot/bzImage-2.6.x. Also change label=linux to label=linux-2.6.x. Save and quit.

Be sure that you have TWO similar paragraphs, one starting with line image=/boot/vmlinuz and another beginning with the line image=/boot/bzImage-2.6.x, this way you will have a "backup kernel" in case this one borks.

14 - as root run "lilo"
15 - cross your fingers, reboot.
16 - enjoy your new kernel

Common Problem often you reboot to a black screen, and it appears nothing is happening, you will even think your computer is broken it is NOT!
This is a framebuffer problem that is being worked on right now, just reboot again and at lilo prompt instead of Linux-2.6.x, highlight linux-2.6.x and append "vga=normal' then
enter. Or you can just type linux-2.6.x vga=normal at the lilo prompt and you should be home free.

http://www.vectorlinux.com/Docs/vl50/vlfaq/kernel2.6.x.htm