hyväksymispäivä arvosana

arvostelija

Bottiverkot ja niiltä suojautuminen

Visa Röyskö

Helsinki 24.9.2020

HELSINGIN YLIOPISTO
Tietojenkäsittelytieteen laitos

HELSINGIN YLIOPISTO – HELSINGFORS UNIVERSITET – UNIVERSITY OF HELSINKI

Tiedekunta – Fakultet –Faculty	Laitos – Institution – Department
Matemaattis-luonnontieteellinen tiedekunta	Tietojenkäsittelytieteen laitos

Tekijä – Författare – Author

Visa Röyskö

Työn nimi – Arbetets titel – Title

Bottiverkot ja niiltä suojautuminen

Oppiaine – Läroämne – Subject

Tietojenkäsittelytiede

Työn laji – Arbetets art – Level	Aika – Datum – Month and year	Sivumäärä – Sidoantal – Number of pages
Gradu	23.9.2020	42 sivua + 7 liitesivua

Tiivistelmä – Referat – Abstract

Bottiverkot ovat sellaisten laitteiden verkkoja, jota on saastutettu haittaohjelmalla. Verkon hallitsija voi antaa näille koneille komentoja ja komentaa ne tekemään hyökkäyksiä. Hyökkäyksiä ovat muun muassa hajautetut palvelunestohyökkäykset ja roskapostin lähettäminen.

Tässä opinnäytetyössä verrataan kolmea erilaista bottiverkko-ohjelmistoa. Vertailussa käydään läpi niiden topologiaa, sekä erityispiirteitä. Lopuksi käydään läpi erilaisia tapoja bottiverkoilta suojautumiseen.

ACM Computing Classification System (CCS):

General and reference ~Document types ~Surveys and overviews

Security and privacy ~Network security

Security and privacy ~Network security ~Denial-of-service attacks

Security and privacy ~Network security ~Firewalls

Avainsanat – Nyckelord – Keywords

bottiverkot, haittaohjelmat, tietoturva, palveluestohyökkäys, roskaposti

Säilytyspaikka – Förvaringställe – Where deposited

Muita tietoja – Övriga uppgifter – Additional information

Sisältö

1 Johdanto

Bottiverkko on joukko koneita, jotka on saastutettu bottiverkko-ohjelmiston haittaohjelmalla. Saastutetut koneet muodostavat verkon, jolle hyökkääjä voi antaa komentoja.

Bottiverkkoja käytetään koneiden käyttäjien tietämättä erilaisiin hyökkäyksiin. Erilaisia hyökkäyksiä ovat muun muassa roskapostin lähettäminen, kalastus sekä hajautetut palvelunestohyökkäykset. Lisäksi koneiden ei välttämättä tarvitse tehdä hyökkäyksiä muihin koneisiin, vaan ohjelmisto voi vain tarkkailla koneen käyttöä ja esimerkiksi tallentaa näppäimistön käyttö ja ohjata ne varastoon verkon kautta.

Näihin tarkoituksiin on hyödyllistä, että hyökkäys saadaan lähtemään monesta lähteestä. Bottiverkoilla voidaan toteuttaa todella laajoja palvelunestohyökkäyksiä. joilla voidaan joko tukkia yksittäinen palvelu, tai pahimmillaan hyökkäyksen kohdistuessa laajalti nimipalvelimiin saada koko internet jumiin.

Aikaisemmin bottiverkkoon kuului vain tietokoneita. Nykyään kun mutta nykyään siihen voi kuulua myös muita Internetissä olevia laitteita, kuten esimerkiksi webbikameroja.

Luvussa 2 selvitetään perustietoja botnet-verkoista ja niiden luomisesta sekä verkkojen erilaisista topologioista. Luku 3 käsittelee tietoturvaa. Siinä tarkastellaan erilaisia verkoilla tehtäviä hyökkäyksiä, keskittyen tarkimmin hajautettuihin palvelunestohyökkäyksiin. Luvussa 4 käydään yksityiskohtaisesti läpi kolmen erilaisen botnet-ohjelmiston toiminnallisuus. Käsiteltävät ohjelmistot ovat Storm Worm, Zeus ja TDL4. Storm Worm oli 2000 -luvun lopupuolella yksi pahimmista roskapostittajista. Zeus on tullut kuuluisaksi verkkopankkeihin kohdistuneista hyökkäyksistä. TDL4 oli aikanaan kaikkein edistynein ohjelmisto, joka hyödynsi tietoliikenteessään julkista KAD-verkkoa.

Luku 5 keskittyy tapoihin, joilla puolustaudutaan tai tarkkaillaan bottiverkkoja. Bottiverkoilta suojautuminen on jaettu kolmeen tasoon: saastumisen estämiseen, verkkojen toiminnan havaitsemiseen sekä hyökkäyksiltä puolustautumiseen.

2 Yleistietoja bottiverkoista

Tämä luku sisältää perustietoja bottiverkoista. Siinä käsitellään bottiverkkojen muodostamista ja elinkaarta, niiden topologiaa. Lisäksi tarkastellaan tapoja, joilla hyökkääjä voi niistä hyötyä.

2.1 Perustietoa bottiverkoista ja niiden historia

Termi bottiverkko tulee sanoista bot ja net. Botit ovat koneita, joihon on asennettu haittaohjelma. Muista haittaohjelmista bottiverkko-ohjelman erottaa se, että ne hyökkääjä voi luoda saastuneista koneista verkon ja hallita niitä ohjelmiston avulla. Näitä koneita hallitaan niihin asennetun takaoven kautta. Verkon koneita kutsutaan myös zombeiksi. Verkon hallitsijaksi (botmaster) kutsutaan henkilöä, joka hallitsee tätä verkkoa.

Bottiverkkoja hallitaan komentokeskus-infrastruktuurin (command-and-control, C&C) avulla. Näiden kautta verkon hallitsija voi antaa verkon koneille komentoja, joita voivat olla esimerkiksi roskapostittaminen tai palveluestohyökkäyksen aloittaminen. Bottiverkoissa on C&C-palvelimia, joiden kautta verkon komentaminen suoritetaan. Nämä voivat olla joka erillisiä palvelimia, tai sitten verkon koneet voivat toimia tällaisina. [BB+07]

Bottiverkkoja ei alunperin suunniteltu hyökkäyksiä ja vahingon tekemistä varten. Vuonna 1993 kehitettiin Eggdrop-niminen botti, joka oli laajalti käytössä. Se oli IRC-botti, jota käytettiin esimerkiksi IRC-keskustelun seuraamiseen, kun tilin omistaja ei ollut paikalla.
Ensimmäinen vihamielinen bottiverkko-ohjelmisto oli GT-bot, joka tuli käyttöön 1998. Vuonna 1999 ilmestynyt PrettyPark worm oli ensimmäinen mato käyttäjien IRC-tilien vihamieliseen hallintaan. Alkuperäiset bottiverkko-ohjelmistot olivat yksinkertaisia. Nykyään ne ovat todella monimutkaisia ja niissä on monenlaista toiminnallisuutta. [JLZ09]

2.2 Bottiverkkojen topologiat ja viestintä

Bottiverkoissa yhteys hyökkääjältä verkon koneisiin voidaan toteuttaa erilaisilla tavoilla. Tässä luvussa käydään läpi erilaisia topologioita botnet-verkkojen viestintään. Näitä ovat keskitetty malli, Peer-to-Peer -verkot. Lisäksi käydään läpi myös hybridimallia, joka on yhdistelmä näistä kahdesta mallista.

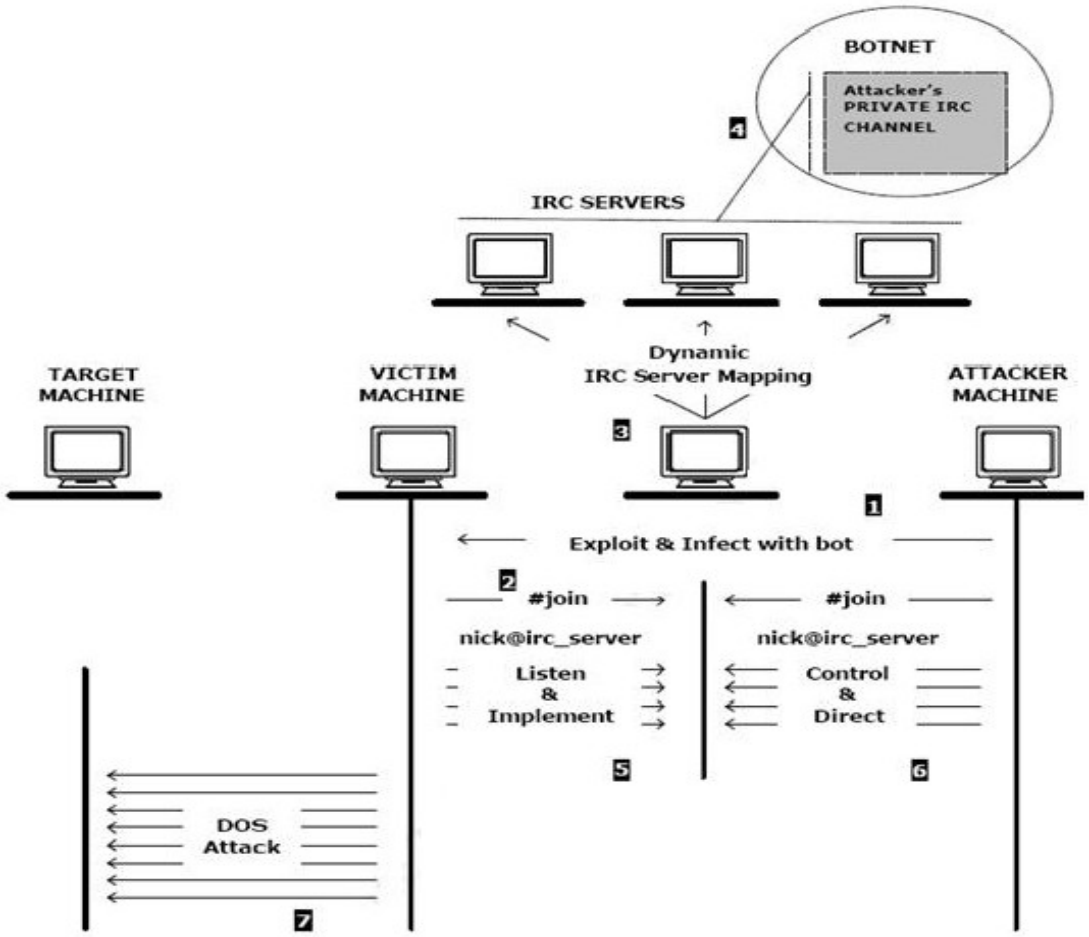
2.2.1 Viestintään käytetyt protokollat

C&C -palvelimet käyttävät tietojen toimittamiseen yleensä IRC (Internet Relay Chat) - ja HTTP -protokollia. Näiden toiminnallisuuden suurin ero on se, että IRC-protokollaa käyttämällä tiedot toimitetaan PUSH-periaatteella, kun taas HTTP-protokollalla PULL-periaatteella.

Vanhimmat bottiverkot hoitivat viestinnän IRC-protokollalla. on monia etuja, esimerkiksi kommunikaation viive on pieni ja se mahdollistaa anonyymin kommunikaation. Hyökkääjä voi joko komentaa kaikkia botteja tai sitten vain osaa niistä. Useimmiten hyökkääjä luo kanavan C&C -palvelimiin, johon kaikki botit liittyvät ja jota kautta ne saavat ohjeensa toimia. [Pur03]

Kuvassa 1 on kuvattu IRC-protokollaan perustuvan Botnetin toiminta [Pur03].

1. Haitallinen koodi asennetaan koneeseen
2. Botti pyrkii saamaan yhteyden IRC-palvelimeen käyttäen satunnaista lempinimeä (nick name) hyökkääjän yksityisellä kanavalla.
3. Hyökkääjä voi tämän estääkseen käyttää erilaisia palveluntuottajia (esim dyndns.com tai no-ip.com) useampien IRC-palvelimien avulla siten, että ne dynaamisesti ohjaavat (map) botteja useammalta IRC-palvelimelta. Tämä estää IRC-administraattoreita antamasta porttikieltoa hyökkääjälle, joka käyttää toistuvasti julkista IRC-palvelinta hyökkäysten tekemiseen.
4. Botti liittyy hyökkääjän yksityiseen kanavaan.
5. Botti kuuluu nyt botnet-verkkoon ja odottaa ohjeita hyökkääjältä.
6. Hyökkääjä lähettää botille ohjeet hyökkäyksen suorittamiseen.
7. Botti aloittaa hyökkäyksen.



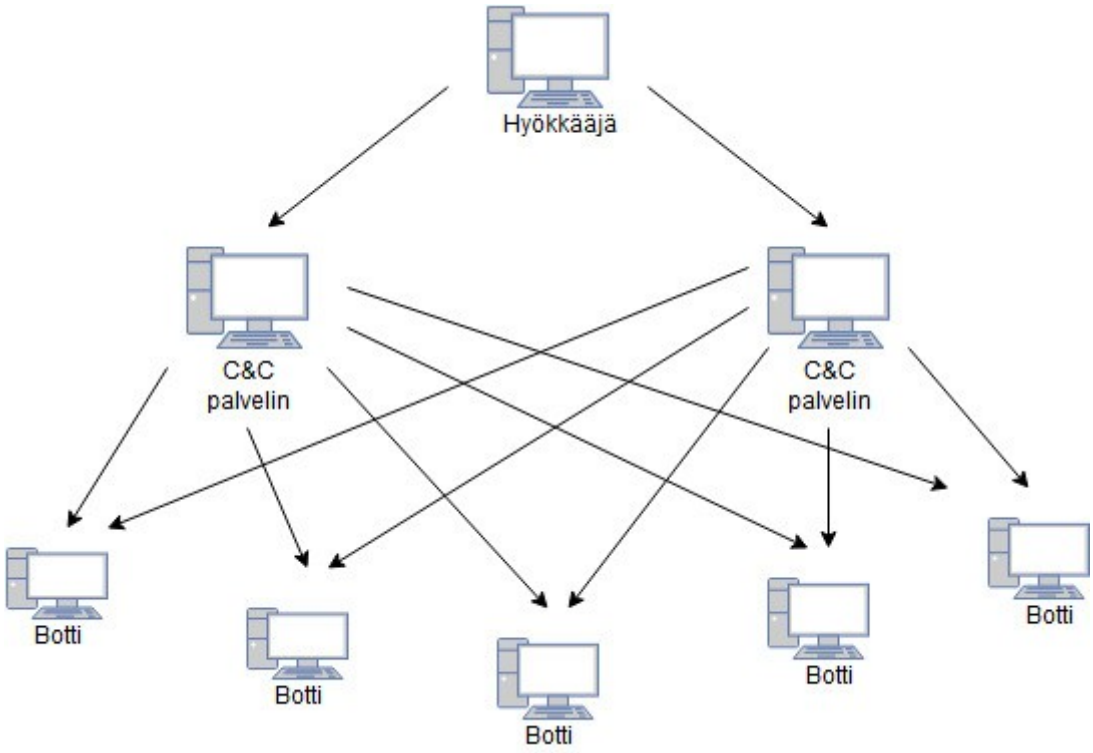
Kuva 1: Bottiverkon koneiden toiminta IRC-protokollaa käyttäen [Pur03]

IRC-protokollan käyttämisen huono puoli bottiverkoissa on se, että tietoturvatutkijat alkoivat monitoroida IRC-liikennettä bottiverkkojen havaitsemiseksi. Lisäksi palomuurit on usein säädetty estämään IRC-protokollan yleensä käyttämät portit. Tähän vastauksena hyökkääjät alkoivat käyttämään HTTP-protokollaa välittämään tietoja C&C-palvelimilta verkon boteille. HTTP-protokollaa käyttäen bottiverkon tietoliikenne sulautuu normaaliin web-liikenteeseen, ja näin ollen sitä on vaikeampi havaita. Normaalin verkkoliikenteen seassa se myös läpäisee helpommin palomuurit. [StS09]

2.2.2 Keskitetty malli (centralized model)

Keskitetty malli, josta käytetään myös termiä hierarkinen malli, on bottiverkkojen topologioista vanhin. Mallin ideana on se, että verkon haltija komentaa verkon botteja

yksittäisten C&C palvelimien kautta. Nämä ovat yleensä sellaisia koneita, joilla on käytössä paljon kaistaa (bandwith). Esimerkki keskitetyn mallin mukaisesti järjestetystä bottiverkossa kuvassa 2.



Kuva 2: Kuvaus keskitetyn mallin mukaisesti toimivasta bottiverkosta. Hyökkääjä antaa komentoja C&C-palvelimille, joka välittää ne verkon boteille.

Keskitetystä mallista on erilaisia topologioita, kuten tähti-, monipalvelin sekä hierarkinen malli. Tähtimallissa on yksi C&C -palvelin, joka hallitsee kaikkia botteja. Monipalvelinmallissa (multiserver) on useampi C&C -palvelin, jotka palvelevat kaikkia botteja. Hierarkisessa mallissa eri C&C-palvelimet hallitsevat kukin omaa osaa verkosta. [Oll09A]

Tähtimallissa huonoin puoli on se, että mikäli ainoa C&C-palvelin saadaan pois pelistä, koko verkko halvaantuu täysin. Monipalvelinmalli kestää yhden C&C-palvelimen kaatumisen. Lisäksi se, että on useampi maantieteellisesti hajautettu C&C-palvelin yhden sijaan, nopeuttaa yhteyksiä verkon botteihin. [Oll09A]

Hierarkisessa mallissa verkko on järjestetty siten, että liikenne kulkee joidenkin

valittujen bottien kautta, joka jakaa ne edelleen muille verkon koneille. Hierarkisessa mallissa etuna on se, että bottiverkko on helppo myydä eteenpäin. Haittana taas on se, että tiedon liikkuminen kaikille verkon koneille kestää kauemmin, mikä hankaloittaa hyökkäämistä. [Oll09A]

Paras puoli keskitetyssä mallissa on se, että viestien viive on pieni, ja verkon hallinta helppoa. Tiedot liikkuvat nopeasti kaikille verkon koneille, jolloin hyökkääjä voi helposti järjestää botnet-verkon ja tehdä hyökkäyksiä.

Keskitetyn mallin suurin heikkous on se, että C&C -palvelin on mallin heikko kohta. Mikäli se löydetään ja saadaan eliminoitua, koko verkon toiminta halvaantuu. Tämän vuoksi nykyään bottiverkoilla on lista ip-osoitteita vaihtoehtoisiin C&C -palvelimiin. Lisäksi keskitetyssä mallissa hyökkääjän identiteetin selviäminen on ei-keskitettyä (decentralized) mallia todennäköisempää. [Oll09A]

2.2.3 Peer-to-Peer -verkot (P2P)

P2P-bottiverkoilla ei ole yksittäisiä C&C-palvelimia, vaan kaikki verkon koneet toimivat sellaisina. Tästä johtuen verkon havaitseminen on keskitetyn mallin mukaan organisoitua bottiverkkoa hankalampaa, ja sen eliminointi on todella vaikeaa.

Jokainen verkon kone pitää yhteyksiä muihin verkon botteihin. Verkon botit toimivat sekä asiakkaina, että palvelimina. Jokaisen uuden botin täytyy tietää joitakin osoitteita verkosta, jotta verkon hallitsijan käskyt saadaan perille kaikkiin verkon koneisiin. Jotkut P2P bottiverkot ovat joltain osin epäkeskitettyjä ja toiset ovat täysin epäkeskitettyjä.

P2P-mallin tärkein etu bottiverkoille on se, että koska siinä ei ole yksittäisiä C&C-palvelimia, ei myöskään ole sellaisia yksittäisiä kohteita, jotka sulkemalla saataisiin koko verkko käyttökelvottomaksi. Toisaalta huonona puolena on se, että viestien välitys koko verkolle on hitaampaa, sillä viestin pitää kulkea monen botin kautta, ennen kuin se saavuttaa koko verkon, toisin kuin keskitetyssä mallissa, jossa viesti lähtee C&C-palvelimelta suoraan boteille. [Oll09A]

2.2.4 Hybridi peer-to-peer bottiverkko

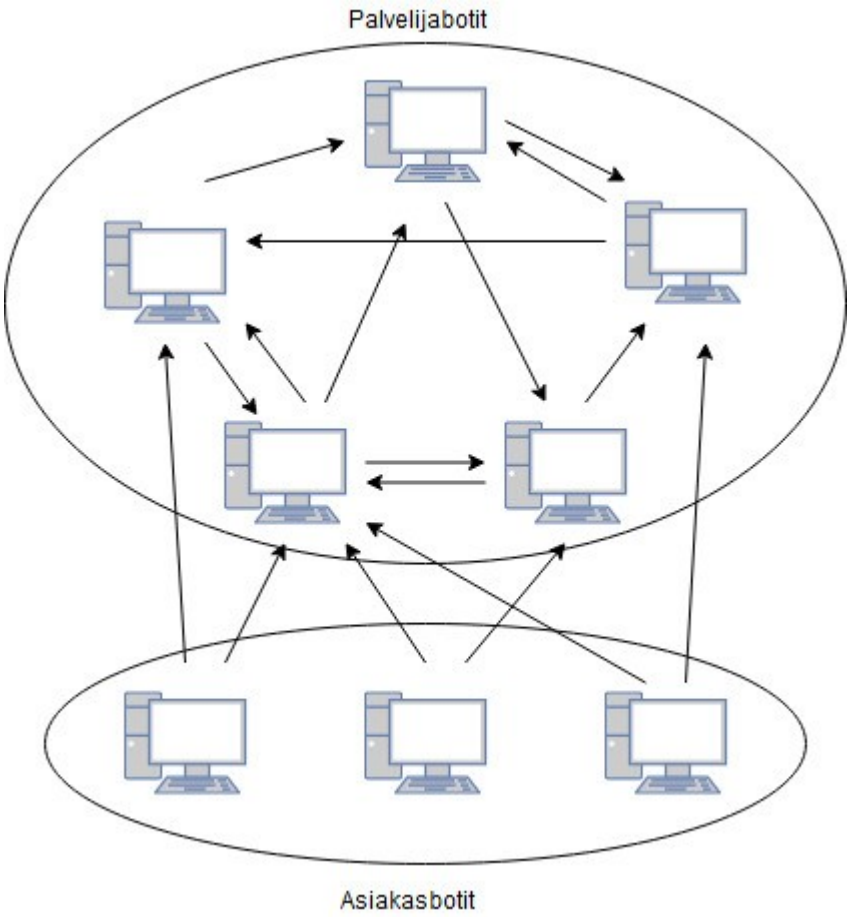
Lähteessä [SWZ08] esitellään bottiverkolle sellainen topologia, joka olisi yhdistelmä . Hybridiverkko yhdistäisi hierarkisen ja epäjärjestäytyneen verkon hyvät puolet. Tässä

mallissa on kahdenlaisia botteja, palvelijabotit ja asiakasbotit.

Palvelijabotit (servant bots) toimivat sekä palvelimina, että asiakkaina. Niillä on staattinen ei-yksityinen IP-osoite ja ne ovat saavutettavissa julkisesta Internetistä. Vain palvelijabotit ovat ehdokkaita peer-listoissa.

Asiakasbotteja voi olla kolmenlaisia: 1) sellaisia, joilla on dynaamisesti varattu osoite, 2) sellaisia, joilla on yksityinen IP-osoite ja 3) botit jotka ovat palomuurin takana, jolloin niihin ei saa yhteyttä globaalista Internetistä. Asiakasbotit eivät hyväksy sisään tulevia yhteyksiä. Sekä asiakasbotit että palvelijabotit käyvät tasaisin väliajoin kyselemässä palvelinboteilta omilla peer-listoillaan uusia komentoja, joita botmaster on antanut.

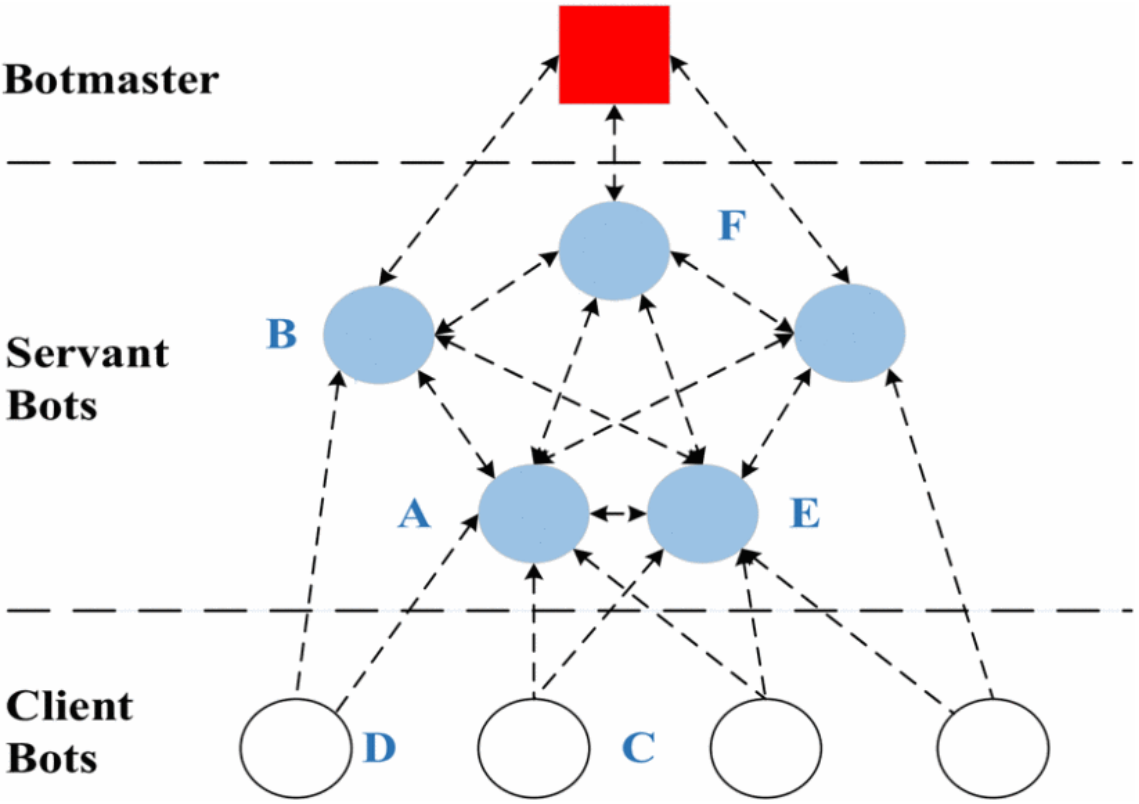
Hybridiverkko on laajennettu versio keskitetystä mallista, jossa palvelijabotit voidaan nähdä C&C -palvelimina. Tällaista verkkoa on kuitenkin vaikeampi saada suljettua, sillä palvelijabotteja on suurempi määrä kuin keskitetyssä mallissa, ja ne ovat yhteydessä toisiinsa. [SWZ08]



Kuva 3: Hybridin bottiverkon topologia [SWZ08]

Tämänkaltaista topologiaa käyttäviä bottiverkkoja on toteutettu. TRBot-bottiverkon botit jakautuvat palvelija- ja asiakasbotteihin. Tätä on hahmotettu kuvassa 4. Tässä mallissa palvelijabotit ovat saavutettavissa globaalista Internetistä. Niillä on staattiset IP-osoitteet. Vain palvelijabotiti voivat olla kandidaatteja peer-listoilla.

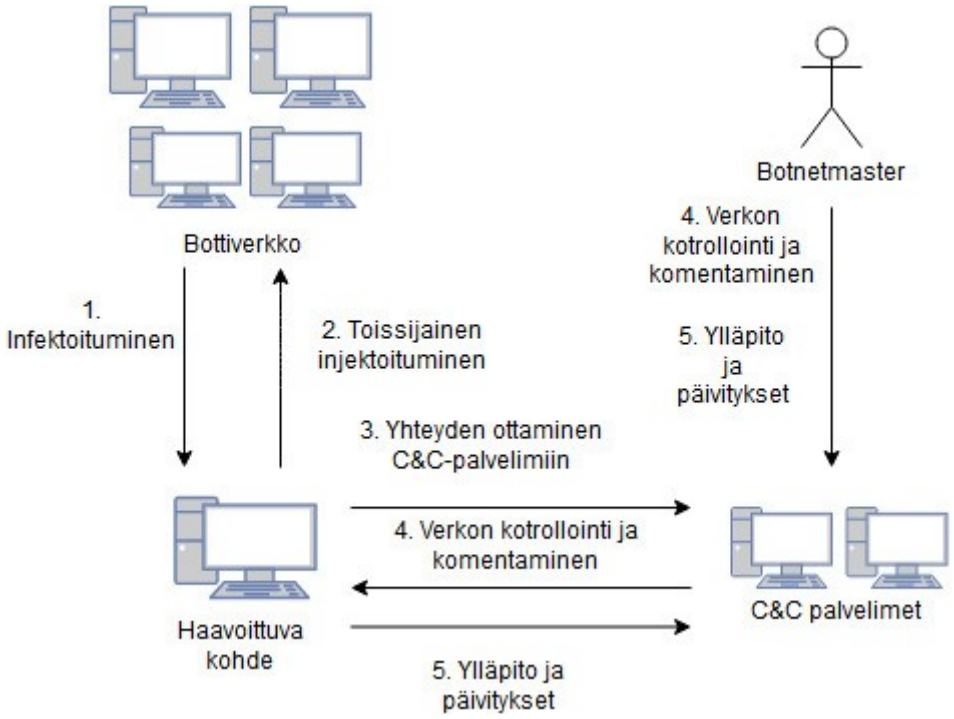
Palvelijabotit eivät ole saavutettavissa globaalisti, vaan niillä on joko privaatit IP-osoitteet, dynaamiset IP-osoitteet tai sitten julkiset IP-osoitteet, mutta niissä on palomuurit, jotka estävät liikenteen niihin. Asiakasbotteja ei ole peer-listoilla. [CLY17]



Kuva 4: TRBot topologia.

2.3 Bottiverkon luominen

Bottiverkkojen tavallinen elämänkaari voidaan jakaa viiteen vaiheeseen. Tätä on kuvattu kuvassa 4. Ensimmäinen vaihe bottiverkon luomisessa on se, että saadaan haittaohjelma asennettua koneisiin ja kohde saastumaan. Siinä hyökkääjä skannaa kohdekoneen haavoittuvuudet ja infektoi koneen.



Kuva 5: Bottiverkon tyypillinen elinkaari [FRS09]

Tämän jälkeen suoritetaan shell-code -niminen skripti, joka käy hakemassa määritellystä paikasta varsinaisen bot-binäärikoodin. Sitten binäärikoodi installoi itsensä uhrikoneelle ja näin muuttaa koneen botiksi ja liittää sen osaksi bottiverkkoa. Bottiohjelma käynnistyy aina automaattisesti koneen käynnistyessä. [FRS09]

Tämän jälkeen perustetaan C&C-kanava koneen ja C&C-palvelimen välille. Tässä vaiheessa hyökkääjä voi alkaa suorittaa hyökkäyksiä myös tämän koneen kautta. Seuraavassa vaiheessa hyökkääjä lähettää C&C-palvelimen kautta koneelle komentoja. Näitä voivat olla muun muassa palvelunestohyökkäyksen käynnistäminen tai roskapostin lähettäminen. Viimeisessä vaiheessa kone käy hakemassa ajantasaisen version binäärikoodeista. Tämä tarvitaan esimerkiksi sen vuoksi, että päivityksessä on uusia keinoja välttää niiden havaitsemista. [FRS09]

2.4 Bottiverkoilla tehty bisnes

Kyberkriminaalit voivat hyödyntää bottiverkkoja monella tavalla. ”Bottiverkkoja pyörittävät suuret rikollisorganisaatiot” kommentoi Bradley Anstis, VP of Technology Strategy for M86 Security. [Vau10] Tässä luvussa käydään läpi erilaisia tapoja tehdä

bisnestä bottiverkoilla.

2.4.1 Roskaposti

Bottiverkot ovat hyviä välineitä roskapostin lähettämiseen. Sen lisäksi, että botnet-verkoilla voidaan lähettää paljon roskaposti, tässä vältetään myös lähettäjäosoitteen joutuminen mustalle listalle, joka on vaarana kun lähettää paljon postia samasta osoitteesta. Bottiverkoissa lähetys hajautetaan satoihin tuhansiin osoitteisiin. Rikollinen voi myös myydä osoitteet muille kyberkriminaaleille. Kaspersky Laboratoryn tietojen mukaan keskiverto roskapostittaja tienaa 50 000 – 100 000 dollaria vuodessa. [Kam08]

2.4.2 Hajautettu palvelunestohyökkäys

Bottiverkot ovat äärimmäisen tehokkaita hajautetussa palvelunestohyökkäyksissä, sillä suurella verkolla saa aikaan todella tehokkaan hyökkäyksen. Hyökkääjä voi tienata rahaa toteuttamalla laajan palvelunestohyökkäyksen yrityksen palvelua vastaan ja tämän jälkeen kiristää tältä rahaa siitä, että lopettaa hyökkäyksen. Suuria hajautettuja palvelunestohyökkäyksiä vastaan on melkein mahdotonta puolustautua ja usein yritys maksaakin hyökkääjälle. Hajautettuja palvelunestohyökkäyksiä käytetään myös hyökkäyksiin valtioita vastaan. Näitä hyökkäyksiä käsitellään tarkemmin luvussa 3. [Kam08]

2.4.3 Internetin käyttäminen anonyymisti (Anonymous Internet access)

Hyökkääjät voivat käyttää saastuneita koneita internetin käyttämiseen anonyymisti. Tällöin he voivat toteuttaa verkon koneiden avulla esimerkiksi hyökkäyksiä verkkopankkia vastaan. [Kam08]

2.4.4 Botnet-verkkojen tai niillä hankitun tiedon myyminen

Botnet-verkon kasaajan ei välttämättä tarvitse käyttää verkkoa ollenkaan itse. Hyökkääjä voi myös toteuttaa verkon, ja myydä sen eteenpäin jollekin toiselle kyberkriminaalille joka voi käyttää sitä haluamallaan tavalla. [Kam08]

Hyökkääjä voi myös kaupata botnet-verkolla saamiaan tietoja, esimerkiksi pankkitunnuksia eniten tarjoavalle. Rahamuuliksi kutsutaan henkilöä, jonka tilille haittaohjelmien tekemät tilisiirrot tallennetaan. Muulin tehtävä on siirtää nämä rahat

eteenpäin rikollisille. Muulin täytyy olla valmis nostamaan rahat tililtä nopealla aikataululla, jotta ei jäisi kiinni. Rahamuuli värvätään yleensä sähköpostilla tai esimerkiksi Facebook-palvelusta. Koska rahat tulevat muulin tilille, hänellä on vaara jäädä kiinni, mutta varsinaiset rikolliset ovat turvassa. [DDM11]

2.4.5 Bitcoin-virtuaalivaluutan louhiminen

Bottiverkoilla voi tehdä rahaa myös bitcoin-valuutan avulla. Miner bottiverkko, joka on saanut nimensäkin virtuaalivaluutan louhimisesta, tuli 2011 kuuluisaksi laajasta palvelinestohyökkäyksestä verkkopalveluita vastaan. Ohjelmistossa on ominaisuuksina myös valuutan louhiminen saastutetulta koneelta. [GeP12]

2.5 Verkon koko

Aikaisemmin bottiverkkojen haluttiin olevan mahdollisimman suuria. Pienissä verkoissa on kuitenkin se etu, että palveluntarjoajien on vaikeampi havaita niitä ja näin niiden elinikä kasvaa. Tästä johtuen hakkerit skaalaavat verkot aiempaa pienemmiksi nykyään. Kun vuonna 2004 botnet-verkon keskimääräinen koko oli noin 100 000 saastunutta konetta, niin vuonna 2006 enää n 20 000 konetta. [Vau10]

Tietoturvayhtiö Damballa tutki 600 erilaista bottiverkkoa, Näiden tutkimuksen mukaan pieniä (1-100 bottia) oli 57% verkoista, keskikokoisia (101-500 bottia) 21%, suuria (501 -10 000 bottia) 17% verkoista ja todella suuria (yli 10000 bottia) 5 prosenttia verkoista.

Eri kokoisilla verkoilla on erilaiset tehtävät. Suuret bottiverkot ovat hyviä laajoissa hyökkäyksissä, kuten roskapostin lähettämisessä ja laajojen palvelunestohyökkäysten toteuttamisessa, jolloin hyökkäystä on toteuttamassa useita koneita.

Pienet verkot on usein viritetty jotain tiettyä ympäristöä vastaan, jolloin ne ovat myös vaarallisia kohteelle. Niitä käytetään usein passiiviseen tarkkailuun ja seurantaan. Uhri ei tällöin välttämättä edes tiedä olevansa hyökkäyksen kohteena. Vaikuttaa siltä, että pienet bottiverkot ovat usein ammattimaisemmin hallittuja. [Dan09]

3 Bottiverkkojen suhde muihin haittaohjelmiin ja niillä tehtyjä hyökkäyksiä

Tässä luvussa käydään läpi tietoturvan perustietoja bottiverkkoihin liittyen. Ensiksi käydään läpi erilaisia haittaohjelmia. Sitten käydään läpi erilaisia hyökkäyksiä perehtyen tarkimmin hajautettuun palvelunestohyökkäykseen, sekä roskapostiin.

3.1 Haittaohjelmat

Haittaohjelma (malware) on ohjelma, joka on tehty aiheuttamaan tietojärjestelmissä ja tietokoneissa toiminnallisuutta. Bottiverkot eroavat muista haittaohjelmista C&C -palvelinten vuoksi, joiden avulla verkon koneita voidaan hallita.

3.1.1 Madot

Mato on itsestään kopioita tekevä ja niitä levittävä ohjelma. Viruksista poiketen siinä ei ole varsinaista isäntäohjelmaa. Matoja levitetään eri tavalla, esimerkiksi sähköpostin liitetiedostoissa. Kun mato on asentunut koneeseen, se tekee itsestään kopion. Uudemmat ja hienommat madot tekevät pieniä muutoksia kopioon, jolloin se ei ole täysin samanlainen ja se hankaloittaa madon toiminnan havaitsemista. Kopioiden levittämisessä mato hyödyntää yleensä joko käyttöjärjestelmän tai muun ohjelman haavoittuvuutta.

Joitakin matoja käytetään verkon tai asennettujen ohjelmien heikkouksien tutkimiseen, joten se mahdollistaa uusien koneiden saastuttamisen automaattisesti. Madoilla voi saada myös aikaiseksi valtavasti liikennettä verkkoon.

Useat madot on suunniteltu leviämisen lisäksi myös tekemään muita haitallisia toimintoja. Tällaisia voi olla esimerkiksi koneen taustakuvan muuttuminen, mutta myös vakavammat asiat, kuten muunlaisten haittaohjelmien asentaminen.

[Wor]

3.1.2 Virukset

Virukst eroavat madoista siinä, että ne tarvitsevat erillisen isäntäohjelman, johon se kopioi ohjelmakoodinsa. Kun virus on saatu koneeseen, se pyrkii infektoimaan

järjestelmän tiedostoja. Virukset voivat infektoitua usean tyyppisiin tiedostoihin, esimerkiksi Office dokumenttitiedosoihin (word, excel ym.), mutta myös ohjelmiin, joita koneen rauta (hardware) käyttää tiiviisti, esimerkiksi Master Boot Record (MBR). Virukset nimetäänkin usein sen mukaan minköä tyyppisiin tiedostoihin ne infektoituvat, esimerkiksi tiedosto infektoijat (file infectors) ja Word virukset ym.

Viruksia voi levittää joko huijaamalla uhria asentamaan virus koneelle jonkin tiedoston mukana, esimerkiksi sähköpostin liitetiedosto. Toinen vaihtoehto on se, että hyödynnetään jotakin koneen haavoittuvuutta. Infektointi voidaan myös suorittaa näiden yhdistelmänä.

Yleisimmin virus replikoituu joka kerta, kun haavoittunut ohjelma ajetaan tai haavoittunutta tiedostoa käsitellään. Tällöin se asentaa uutta koodia samaan tai samantapaiseen tiedostoon. Tällöin se myös usein vahingoittaa tätät tiedostoa ja pahimmillaan saa sen korruptoitua täysin. Lisäksi virus voi tehdä muutakin tuhoa koneella, esimerkiksi tuhota tiedostoja tai varastaa sensitiivistä dataa. Jotkut virukset voivat saada koko käyttöjärjestelmän sekaisin.

1990-luvulla virukset olivat yleisimmin esiintyvä haittaohjelma. Nykyään troijalaiset, madot ja muunkaltaiset haittaohjelmat ovat yleisempiä. Kuitenkin virukset ovat edelleen uhka eteenkin vanhoille huonosti suojatuille järjestelmille.

[Vir]

3.1.3 Troijalaiset

Troijalaiset houkuttelevat uhrin asentaamaan itsensä koneeseen esittämällä jotakin muuta ohjelmaa. Kuitenkin ne taustalla tekevät toisenlaista toimintaa, joka on haitallista uhrille. Troijalaiset usein on naamioitu hyvin, esimerkiksi suosituksi peliksi, elokuvaksi, musiikkitiedostoksi tai päivitykseksi johonkin oikeaan ohjelmaan.

Troijalaisia levitetään sähköpostin liitetiedostoissa, erilaisilla vihamielisillä (malicious) tai vaaraantuneilla sivustoilla, tai sosiaalisessa mediassa. Niitä voi myös asentaa uhrin koneeseen sen haavoittuvuuksia hyödyntäen laittamalla kone asentamaan troijalainen uhrin tietämättä. Tätä kutsutaan driveby lataukseksi (driveby download).

Toimintansa perusteella troijalaiset voi ryhmitellä kahteen ryhmään: datan jakajat (data-dealers) ja kontrollin varastajat (control stealers). Datan jakajat varastavat ja jakavat

eteenpäin yksityisiä tietoja haavoittuneelta koneelta, kuten salasanoja ja pankkikortin tunnuksia. Kontrollin varastajat taas pyrkivät saamaan koneen hallintaansa esimerkiksi takaovea hyödyntäen.

[Tro]

3.1.4 Takaovi

Takaovi on ohjelma, jonka kautta vieras taho voi päästä tietokoneelle ohittaen koneen suojaukset. Takaovet voidaan jakaa systeemi-, ohjelma- ja kryptotakaoviin.

Systeemitakaovet avaavat hyökkääjälle pääsyn systeemitason prosesseihin. Systeemitakaovia hyökkääjä voi normaalisti hyöödyntää silloinkin kun varsinainen haavoittuvuus, jonka kautta ohjelma on koneelle päässyt on korjattu.

Ohjelmatason takaovi on normaalissa ohjelmistossa oleva aukko. Tällainen takaovi on joskus lisätty ohjelmistoon valmistajan toimesta tarkoituksella. Mutta välillä ne ovat jonkun ulkopuolisen lisäämiä, muiden toimesta. Tällöin muokkaaja on päässyt päässyt käsittelemään ohjelmiston ohjelmakoodeja joko käsin tai jonkin ohjelman avulla.

Kolmas kategoria ovat kryptotakaovet, jota ovat kryptaussysteemeihin tarkoituksella jätettyjä heikkouksia. Ne tarjoavat hyökkääjälle mahdollisuuden päästä käsiksi salattuihin tietoihin. [EnW07]

3.1.5 Rootkit ja bootkit

Rootkit-haittaohjelma pääsee muokkaamaan tiedostoja root-oikeuksilla. Se pääsee käsiksi tärkeisiin käyttöjärjestelmän tiedostoihin. Kun rootkit-ohjelma on installoitu koneeseen, hyökkääjällä on salainen pääsy koneen tietoihin aina kun kone on verkossa. Rootkit-ohjelmaa käytetään usein muiden ohjelmien asentamiseen ja takaovien luomiseen. [Mal13]

Jos hyökkääjä haluaa tietoja, hän voi asentaa koneeseen Keylogger-ohjelman, joka tallentaa lokitiedostoon kaiken mitä käyttäjä kirjoittaa. Tällä ohjelmalla hyökkääjä voidaan muun muassa varastaa salasanoja, pankkikorttien numeroita sekä verkkopankin tunnuksia. Tietojen kalastelun lisäksi rootkit-ohjelman saastuttaneista koneista voi luoda bottiverkon, joiden avulla voi tehdä erilaisia hyökkäyksiä. [Mal13]

Bootkit on rootkit-ohjelma, joka ladataan ennen pääkäyttöjärjestelmää. Tämä tekee

bootkit-ohjelmiston vaikeasti havaittavaksi ja myös hankalaksi poistaa. Se on tallennettu pääkäynnistystietueeseen (master boot record, MBR). [Mal13]

3.2 Palvelunestohyökkäys (Denial of Service, DOS) ja hajautettu palvelunestohyökkäys (Distributed Denial of Service)

Palvelunestohyökkäyksessä on tarkoitus lamauttaa hyökkäyksen kohteena oleva järjestelmä niin, että se ei enää pysty palvelemaan asiakkaita. Bottiverkoilla palvelunestohyökkäykset voidaan toteuttaa hajautetusti, jolloin hyökkäys tehdään useasta kohteesta.

Palvelunestohyökkäykset voidaan jakaa seuraaviin kolmeen kategoriaan: [MeP04]

- Tuhoava (destructive): Hyökkäyksen tavoite on estää kohteena olevaa laitetta toteuttamaan tehtäväänsä. Hyökkäykseen kuuluvat kohteen konfigurointitiedostojen tuhoaminen, sekä myös kohtteen virran saannin estäminen (power interruption).
- Resurssien kulutus (recource consumption): Kohteen resurssien kuluttaminen niin, että se ei pysty palvelemaan asiakkaita.
- Kaistan kulutus (bandwith consumption): verkon kapasiteetin kuluttaminen turhilla pyynnöillä (request).

Ensimmäinen hajautettu palvelunestohyökkäys tehtiin 1998. Silloin hyökkäys oli tehoton, mutta nykyään tilanne on toinen. [MeP04] Bottiverkot voivat olla valtavia ja voivat sisältää miljoonia koneita, joten niillä saadaan tehtyä todella voimallisia hyökkäyksiä. Luku perustuu pitkälti lähteeseen [LPR07].

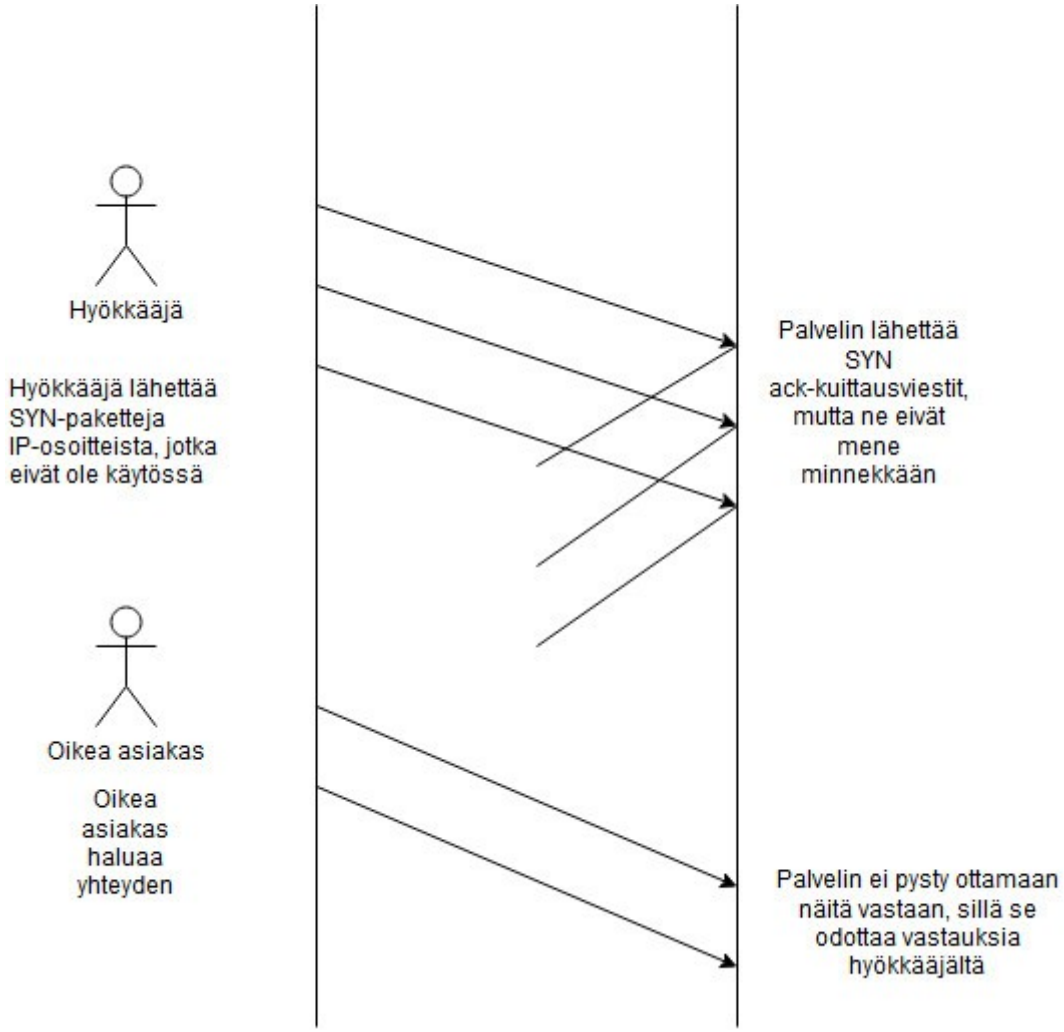
3.2.1 Protokollaan perustuvat kaistahyökkäykset (Protocol-Based Bandwith Attacks)

Protokollaan perustuvat kaistahyökkäykset perustuvat internet-protololliен tiettyihin haavoittuvaisuuksiin. Nämä hyökkäykset voidaan tehdä myös yhdestä lähteestä tehokkaasti.

SYN tulvitus (SYN flood) hyökkää TCP-tason kolmen suunnan kädenpuristuksen (three way handshake) haavoittuvuuteen. Kolmen suunnan kädenpuristus tehdään jokaisen

TCP-yhteyden alussa. Asiakas lähettää SYN-paketin palvelimelle, joka lähettää takaisin SYN-ACK -viestin. Palvelin tallentaa tiedot pyynnöstä muistikekoon (memory stack). Sitten asiakas hyväksyy tämän ja avaa yhteyden.

Hyökkäyksessä hyökkääjä lähettää SYN-paketteja palvelimelle sellaisista IP-osoitteista, jotka eivät ole käytössä. Palvelin lähettää SYN-ACK viestit, mutta ei saa niihin vastausta. Kesken olevat yhteydenotot jäävät kuluttamaan muistikekoa, jolloin oikeat asiakkaat eivät saa palvelua. SYN-tulvitushyökkäys on kuvattuna kuvassa 5. [EdW06]



Kuva 6: Kolmen suunnan kättelyyn perustuva SYN-tulvitushyökkäys [EdW06]

ICMP tulvitus (ICMP Flood) on hyökkäys, joka perustuu IP-protokollaan perustuvaan ICMP-protokollaan. Se on ICMP-paketteja käyttävä kaistan kulutus -hyökkäys. Hyökkäyksessä hyödynnetään IP-protokollan massalähetys (broadcast)-osoitteita, joiden avulla voi lähettää paketteja koko lähiverkkoon.

Yksi ICMP tulvitushyökkäys on Smurf-hyökkäys (Smurf attack). Siinä hyökkääjä lähettää verkkoon viestejä toisen koneen nimissä, joka on hyökkäyksen uhri. Muut koneet vastaavat uhrin ICMP echo -pyyntöön, jolloin uhrin koneelle tulee paljon liikennettä. Nykyään ICMP tulvitushyökkäykset ovat harvinaisia. [LPR07]

3.2.2 Sovelluksiin perustuvat kaistahyökkäykset (Application-Based Bandwith Attacks)

Sovelluksiin perustuvat hyökkäykset jonkin sovelluksen (esimerkiksi sivuston hakukoneen) tulvittamiseen. HTTP-tulvitus perustuu verkkopalvelinten palvelinten kuormittaminen HTTP-pyynnöillä. Verkkopalvelut käyttävät yleisimmin HTTP-protokollaa TCP:n yli portissa 80. Palomuurit on useimmiten konfiguroitu blokkaamaan muut portit, mutta jättämään sen vapaaksi.

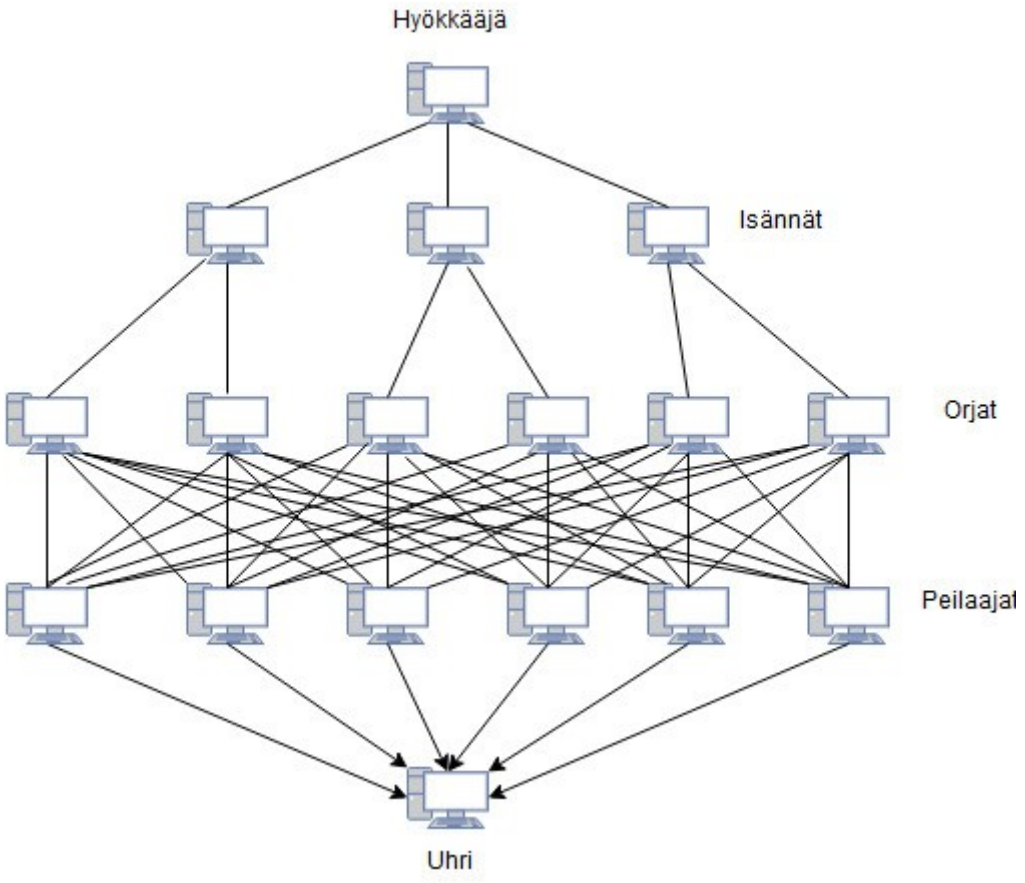
Botnet-verkot sopivat hyvin HTTP-tulvitukseen. HTTP-pyynnön lähettämiseen tarvitaan aito IP-osoite, joten hyökkääjät voivat käyttää verkon saastuneen koneen IP-osoitetta. [LPR07]

Hyökkäyksiä voidaan säätää monella tavalla. Yksi tehokas tapa on lähettää lukuisia HTTP-pyyntöjä, ladata suuri tiedostoja sivustolta, jolloin kohteena oleva palvelin joutuu lataamaan sen muistiin, jolloin siitä tulee paljon kuormaa palvelimelle. Tässä huono puoli on se, että tällainen hyökkäys voidaan havaita ja sulkea.

Parempi tapa on tehdä sivustolle lukuisia pyyntöjä lataamalla sivulta löytyviä linkkejä. Tällaista on hankala havaita, sillä se sekoittuu normaalin HTTP-liikenteen sekaan. [LPR07]

3.2.3 Hajautetut peilaushyökkäykset (Distributed Reflector Attacks, DrDos)

Hajautettu peilaushyökkäys eroaa tavallisesta hajautetusta palvelunestohyökkäyksestä siten, että siinä käytetään apuna bottiverkkoon kuulumattomia laitteita (reflector). Laaja hyökkäys voidaan saada aikaan pienelläkin bottiverkolla. Siinä käytetään IP Spoofing -tekniikkaa, eli hyökkääjä lähettää bottiverkon kautta peilaajalle (reflector) viestejä uhrin



Kuva 7: Kuvaus hajautetusta peilaushyökkäyksestä [MPZ04]

ip-osoitteella. Nämä koneet käsittelevät pyynnön ja lähettävät reply-viestin uhrille. Uhrin koneelle tuleva liikenne on suurempi, sillä nämä palvelimet pystyvät käsittelemään valtavan määrän sivupyyntöjä, jolloin valtava määrä vastausviestejä palautuu uhrille. Hyökkäyksen toimintaa kuvattu kuvassa 7. [MPZ04]

3.3 Roskaposti (spam)

Roskaposti on sähköpostia, jota lähetettään moneen kohteeseen tilaamatta/pyytämättä. Roskapostitus on halpaa, eikä maksa juuri mitään. Roskapostitus on valtava ongelma. Bottiverkot ovat roskapostin lähetyksessä tehokkaita. Viestit saadaan lähtemään monista eri osoitteista, parhaimmillaan kymmenistä tuhansista osoitteista, jolloin viestejä saadaan lähtemään valtava määrä. Tämä myös hämää roskapostisuodattimia, sillä viestejä lähtee vähemmän yhdestä paikasta. Lisäksi viestit lähtevät suoraan ilman sähköpostipalvelimia. [CC+13]

Roskapostia varten on olemassa roskapostin suodattimet, jotka erilaisilla tekniikoilla

koittavat päätellä, mikä posti on roskapostia. Suodattimeen tulevalle sähköpostille lasketaan pistearvo, joka kertoo sen epäilyttävyydestä. Pistearvoa nostaa esimerkiksi viestin koko, kuvien määrä ja se, miten moneen osoitteeseen sama viesti on lähtenyt. Kun pistearvo ylittää määritellyn rajan, viesti suodatetaan pois.[CC+13]

3.4 Kalastus (phishing)

Kalastushyökkäyksessä hyökkääjä yrittää saada uhrin huijattua antamaan arkaluontoista tietoa itsestään, esimerkiksi luottokortin numeron tai verkkopankin tunnukset. Kalastuksella saatiin vuonna 2007 huijattua 3,2 miljardia dollaria [Ira08]. Kaupalliset instituutiot (pankit, paypal, eBay) ovat hyökkäysten pääkohteita.

Yleisimmin kalastus tehdään sähköpostilla, mutta sitä tehdään nykyään myös muussa mediassa, kuten pikaviesteihin (instant message). Tyypillinen kalastusviesti on sellainen, että se esittää olevansa joltakin oikealta organisaatiolta, esimerkiksi pankilta. Viestissä usein ilmoitetaan, että hyökkäyksen kohteen tunnuksessa on ongelmaa ja ohjaa eteenpäin antamaan tietojaan.

Kalastusviestit eroavat tavallisesta roskapostista siten, että sen tarkoitus yrittää esittää kuin olisi legitiimistä organisaatiosta. Koska kalastusviestit on tehty siten, että ne näyttävät uskottavasti laillisilta viesteiltä, niitä ei voi havaita suoraan roskapostin havaitsemiseen käytettävillä tavoilla. [GI+08]

4 Erilaisia bottiverkko-ohjelmistoja

Tässä luvussa tarkastellaan tarkemmin erilaisia bottiverkko-ohjelmistoja. Tarkasteltaviksi on valittu Storm (Worm), Zeus ja TDL4. Käsittelyssä kerrotaan näiden ohjelmistojen historiaa, ominaisuuksia, verkon luomista ja kontrollointia.

4.1 Storm (worm)

Storm worm tunnetaan myös nimillä Nuwar, Peacomm, Tibs, ja Zhelatin (W32/Small.DAM) [Ste07]. Nimestä huolimatta se ei ole pelkkä mato, vaan laajempi kokonaisuus erilaisia haittaohjelmia.

4.1.1 Historia

Storm worm ilmestyi ensimmäisen kerran elokuussa 2006. Se tuli kuuluisaksi 19.1.2007, kun se laukaisi laajoja hyökkäyksiä, jotka levittivät roskapostia 20-kertaisesti normaaliin määrään verrattuna. [Smi08]

4.1.2 Ominaisuudet

Storm bottiverkko on P2P-verkko, jossa tietoliikenne suoritettaan Overnet-verkossa ja käyttää verkon viestittelyyn muokattua versiota Overnet P2P -protokollasta. Overnet on tiedostojen jakamiseen tarkoitettu overlay-verkkoprotokolla ja on Kandemlia-pohjainen hajautettu hajautustaulu (distributed hash table, DHT), jossa jokaisella solmulla (peer) on oma yksilöllinen tunnuksensa.

Alunperin haittaohjelmaa levitettiin roskapostin liitetiedostona olevissa tiedostoissa, jotka näyttivät PDF-tiedostoilta. Haittaohjelman levittäjät kuitenkin muuttivat säännöllisesti ohjelman levitystapaa. Myöhemmin liitetiedostot korvattiin viestissä olevilla linkeillä, jotka johtavat saastuneille sivuille. [Smi08]

Storm-haittaohjelman koodi mutatoituu ja sitä tapahtuu usein, parhaimmillaan jopa tunnin välein. Toisin kuin polymorfisissa viruksissa, mutaatiot tapahtuvat palvelimen päässä (mekanismia kutsutaan palvelinpuolen polymorfisuudeksi, server-side polymorphism). Tämä johtaa siihen, että monille käyttäjille antivirus-tietokannan päivitys on tehotonta. [Ste07]

Haittaohjelma pyrkii olemaan mahdollisimman huomaamaton saastuneessa koneessa. Koneen resursseja pyritään käyttämään rajallisesti, eikä sitä laiteta hyökkäämään jatkuvasti. [Ste07]

Kun kohde saastutetaan, siihen ladataan suoritettavia (.exe) tiedostoja. Nämä tiedostot on useimmiten nimetty nimillä game0.exe – game5.exe. Jokaisella on oma tehtävänsä:

- game0.exe – takaovi/lataaja (downloader)
- game1.exe – SMTP välittäjä (relay)
- game2.exe – Sähköpostiosoitteiden varastaja (stealer)
- game3.exe – Sähköpostivirusten levittäjä
- game4.exe – DDOS hyökkäystyökalu

- game5.exe – päivitetty kopio Storm worm dropperista [Ste07]

Kernel rootkit -ajuri (%windir%\system32\wincom.sys) sisällyttää haittaohjelman koodit services.exe -prosessiin, ja täten huolehtii P2P-linkityksisprosessista. P2P-komponentti on %windir%\system32\wincom.ini -tiedosto, johon on tallennettu kovakoodattu lista yli sadasta verkon solmusta. [Ste07]

4.1.3 Kontrollointi (Command and Control)

Overnet-verkossa jokaisella verkon solmulla on oma 128-bittinen tunnus, jonka verkkoon liittyvä kone luo itselleen. Tämä ID muodostaa yhdessä solmun IP-osoitteen ja portin, jota se kuuntelee, kanssa tripletin, joka on yksikäsitteinen tunniste (identifier).

Jokainen overnet-verkon kone ylläpitää peer-listaa, jossa on muiden verkon koneiden tietoja. Kademlian tapauksessa peer-listat on jaettu alilistoihin joita kutsuteen termillä k-bucket. Nämä sisältävät verkon solmujen triplettejä, jotka etäisyys koneesta on välillä 2^i ja 2^i+1 (0<=i<=128).

Kademlia-protokolla tarjoaa kommunikointiin neljän tyyppisiä viestejä [DF+09].

- PING: Selvitetään onko solmu aktiivinen (on-line).
- STORE: Ohjaa solmun tallentamaan (avain, arvo) -parin myöhempää hakua (retrieval) varten.
- FIND NODE: Tällä viestillä etsitään solmujen tunnuksia. Saadessaan tällaisen viestin solmu palauttaa kysyjälle IP-osoitteensa, porttinsa ja tunnuksensa.
- FIND VALUE: Viestillä etsitään jotakin arvoa

Kone liittyy verkkoon lisäämällä oman triplettinsä omaan k-koriin (k-bucket) ja sitten lähettämällä FIND-NODE -viestin muille verkon solmuille, joiden tunnus on lähinnä koneen omaa tunnusta.

Overnet-verkon solmut eivät salaa toisilleen lähettämiä viestejä, mutta Storm-verkon botit salaavat lähettämänsä viestit 40-bittisellä avaimella. Kun Storm-haittakoodi asentuu uhrin koneeseen, se generoi sille 128-bittisen tunnuksen. Näin kone liittyy bottiverkkoon. Lista päivittyy aina tarpeen mukaan koneen saadessa uusia tietoja muilta bottiverkon koneilta. [DF+09]

Jokaisena annettuna päivänä jokainen verkon kone generoi bottiverkko-ohjelmiston

salaisella avaimen genererointi -funktiolla yhden 32:sta 128-bittisestä mahdollisesta avaimesta. Bottiverkkojen operaattorit voivat siis tallentaa näihin 32 avaimeen liittyviä arvoja tietyissä verkon solmuissa.

Bottiverkon koneet voivat ladata (pull) itselleen uudet komennot ja päivitykset käyttämällä FIND VALUE -hakukyselyjä, joilla ne etsivät sellaista avainta, jonka ne generoivat sinä päivänä. Näin ne voivat laskea (compute) näiden säilytyspaikkojen (repository) yhteystiedot, jolloin niihin voidaan myöhemmin olla yhteydessä HTTP-yhteydellä. [DF+09]

4.2 Zeus

Zeus (kutsutaan myös nimellä Zbot) on monipuolinen työkalupaketti, jota voi käyttää monella tavalla. Alunperin Zeus kehitettiin alunperin pankkitietojen varastamiseen ja on saanut maineensa verkkopankkeihin kohdistuneista hyökkäyksistä.

4.2.1 Historia

ZBot-perheen troijalaiset ilmestyivät 2007. Aluksi Zeus-ohjelmisto levisi ainoastaan tekijän kautta. Tekijä lopetti samoihin aikoihin sen myynnin. Tämän jälkeen kyberkriminaalit rupesivat muokkaamaan sen koodia ja levittivät sitä edelleen.

Ohjelmiston leviäminen kiihtyi huomattavasti loppuvuodesta 2008. Kasperskyn asiantuntija Dimitry Tarakanov epäilee, että yksi syy tähän on taloudellinen laskukausi, jonka takia ohjelmistotyöntekijöitä jäi paljon työttömäksi. Tällöin osa heistä rupesi levittämään ohjelmistoa. Huippukausi ohjelmiston leviämisessä oli maaliskuussa 2009, jolloin liikkeellä oli 5079 kappaletta erilaisia versioita Zeus-troijalaisesta. [Tar10] Toukokuussa 2011 Zeus-työkalupaketin lähdekoodi vuosi täydellisenä internetiin. Tämä toi räjähdysmäisesti lisää uusia variantteja haittaohjelmasta. [AEW15]

4.2.2 Ominaisuudet

Zeus-ohjelmiston saastuttamien koneiden verkkoa tutkitaan koko ajan, ja sen koneista yritetään saada tietoa esiin. Seuraavassa on lueteltu troijalaisen mahdollisia toiminnallisuuksia. [Tar10]

Zeus-haittaohjelma pääsee käsiksi kaikkiin sellaisiin tietoihin, jotka käyttäjä on

tallentanut, esimerkiksi mikäli käyttäjä tallentaa jollekin verkkosivulle tunnuksensa ja salasanansa. Lisäksi troijalainen myös tallentaa lokiin näppäimien painallukset, jolloin se saa kirjautumistiedot selville. Jotkut verkkosivut estävät näppäinten painallusten tallentamisen omalla virtuaalisella näppäimistöllä. Tätä varten Zeus sisältää ominaisuuden, että aina kun käyttäjä painaa hiiren vasenta nappia, ZeuS ottaa ruutukaappauksen, josta painallukset on helppo nähdä.

Mikäli saastuneen koneen käyttäjä menee verkkosivulle, joka on listattu Zeus-ohjelman konfigurointitiedostoon, troijalainen voi muokata verkkosivun koodia ennen kuin se latautuu ruudulle. Lähes aina sivulle ujutetaan osio, jossa kysytään jotakin lisätietoa, jota sivustolla ei muuten kysyttäisi. Esimerkki tästä on pankin sivulla pyyntö antaa PIN-koodi. Jos käyttäjä erehtyy sen antamaan, se lähetetään heti verkon hallitsijalle.

Alempana esimerkkilomake ennen muutosta:

```
<TR>
  <TD>Username:</TD>
  <TD><INPUT id=username name=username></TD></TR>
 <TR>
  <TD>Password:</TD>
  <TD><INPUT type=password name=password></TD></TR>
 <TR>
  <TD colSpan=2><INPUT type=submit value=Submit></TD></TR>
```

Lomake muutoksen jälkeen:

```
<TR>
  <TD>Username:</TD>
  <TD><INPUT id=username name=username></TD></TR>
 <TR>
  <TD>Password:</TD>
  <TD><INPUT type=password name=password></TD></TR>
```

<TR>

<TD>PIN:</TD>

<TD><INPUT id=pinnumber name=pinnumber></TD></TR>

<TR>

<TD colSpan=2><INPUT type=submit value=Submit></TD></TR>

Hyökkääjä voi myös lähettää roskapostia tai tehdä muita hyökkäyksiä verkon koneiden avulla. ZeuS ei sisällä työkaluja näihin, mutta se voi tarvittaessa asentaa hyökkäyksiin tarvittavia ohjelmistoja koneelle. Troijalainen myös varastaa sertifikaatteja, joita tarvitaan joillekin sivustoille kirjautumiseen. [Tar10]

Keskitettyä mallia käyttäneiden versioiden verkon välinen liikenne kryptattiin käyttäen XOR-algoritmia ”visuaalisella kryptauksella”. Vuodesta 2013 lähtien ruvettiin käyttämään sen sijaan RC4-algoritmia. Avaimena tässä käytetään vastaanottajan botin identifioijaa (recipient bot identifier). Tässä on se ongelma, että botti ei välttämättä aina tiedä, mikä on sen oma identifioija muille verkon boteille. Tämä voi estää sitä avaamasta saamiensa viestien salausta. [AB+13]

Mobiililaitteet eivät perinteisesti ole olleet bottiverkoille kovinkaan merkittävässä roolissa. Niiden resurssit ovat pienemmät, akun kesto on rajoitettu. Lisäksi ne eivät ole aina yhteydessä Internettiin. Zeus-in-the-mobile (ZitMo) on yksi esimerkki mobiililaitteissa olevasta haittaohjelmasta. Kohteena on verkkopankkien käyttäminen. Tarkoituksena on varastaa mobiilin transaktion autorisointinumerot (mTAN) lähettämällä uhrille SMS-viesti, joka väittää olevansa pankilta. Viestissä on mukana väärennetty URL-osoite, joka pyytää uhria muka lataamaan tietoturvakyselyn (security question), mutta todellisuudessa se on Zeus-verkon botti. [AB+13]

Zeus ei sisällä troijalaisen levittämiseen tarkoitettuja komponentteja. Sitä levitetään yleisimmin roskapostituksen avulla, joissa yritetään erilaisilla tavoilla saada henkilö asentamaan troijalainen koneeseensa. Se, ettei troijalainen leviä hallitsemattomasti tekee siitä vaikeammin havaittavan ja torjuttavan.

Koneen saastuttamisessa ensimmäisenä on muokattava konfiguraatiotiedostoa, joka koostuu staattisesta ja dynaamisesta osasta. Staattisen konfiguraatiotiedoston (StaticConfig) kääntää (compile) rakennustyökalu (Builder Tool). Verkon botteja

komennetaan kahdella tavalla. (1) botit ottvat yhteyttä proxy-botteihin saadakseen komentoja. (2) Konfiguraatiotiedostojen päivityksiä voi myös käyttää bottien komentamiseen. [AB+13]

Alunperin hallinta toteutettiin keskitetyn mallin mukaan. Vuonna 2011 se muokattiin käyttämään peer to peer -mallia. Kuitenkin zombi-koneet olivat vielä yhteydessä (http-protokollaa käyttäen) erillisiin C&C -palvelimiin. Verkon koneet luovat ensin yhteyden UDP-kädenpuristuksella ja sen jälkeen koneet jakavat esimerkiksi verkon solmujen osoitelistat ja konfiguraatiotiedostot TCP-kerroksella.

Uudessa päivityksessä verkon koneiden aikaisemmin TCP:n päällä tehtyjen tietojen jakaminen tapahtuu UDP-yhteyksillä. TCP-yhteydet ovat helpompi jäljittää ja katkaista ja pakettien jakamisessa ei ole autentikointia, jolloin kuka tahansa voi onnistuneesti kommunikoida verkon botin kanssa. UDP-yhteydet on vaikeampia havaita.[Lel11]

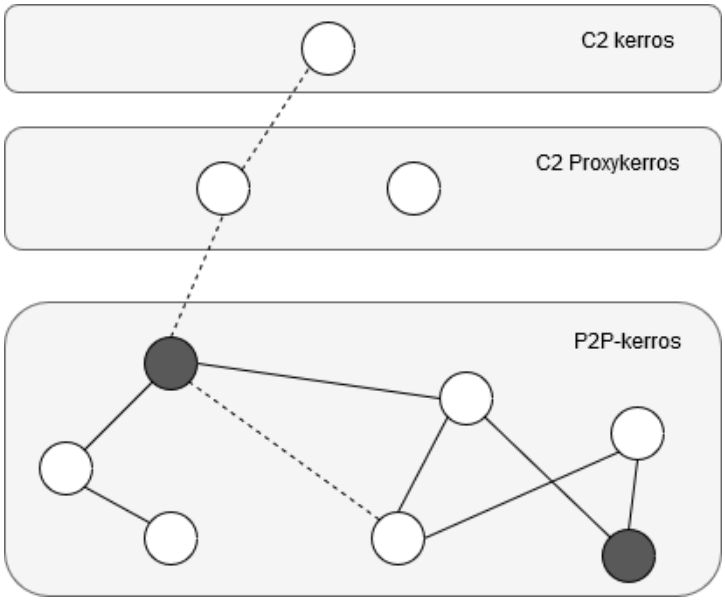
4.2.3 Kontrollointi

Sellaisissa Zeus-varianteissa, joiden tietoliikenne on toteutettu keskitetyn mallin mukaan, tarjotaan usein käyttäjälle mahdollisuus luoda oma yksityinen bottiverkko. P2P -versiossa tätä mahdollisuutta ei enää tarjota. P2P Zeus perustuu yhteen koherenttiin P2P-verkkoon. Se on jaettu lukuisiin virtuaalisiin alibottiverkkoihin. Nämä aliverkot ovat itsenäisiä, ja niiden hallintaan omat verkon hallitsijat. [AB+13]

Gameover Zeus (P2P Zeus) on pisimmälle kehitetty Zeus toijalainen, joka on rakennettu P2P-infrastruktuuriin. Sen kehittäjät tekevät lähdekoodiin vuosittain päivityksiä, jotta se saadaan suojattua.

Gameover Zeus -bottiverkko on organisoitu kolmelle kerrokselle. Tätä on kuvattu kuvassa 8. Alin kerros on P2P-kerros, jossa verkon koneet toimivat. Välillä jokin osa verkon koneita nostetaan verkon hallitsijan toimesta proxy-botiksi. Verkon hallitsija ilmoittaa tästä verkolle proxy-ilmoituksella (proxy announcement). [AB+13]

Proxy-bottien tehtävä on kerätä verkosta saatuja komentoja ja toimittaa varastettua dataa etteenpäin. Proxy botit toimivat yhteydessä toiseen kerrokseen, jota kutsutaan C2 Proxy kerrokseksi. Tämä taso koostuu HTTP-palvelimista jotka eivät ole botteja.



Kuva 8: Gameover Zeus -verkon topologia. [AB+13]

Ylin taso on C2 taso. Se toimii komentojen lähteenä ja varastetun datan varastona. Komennot jaetaan C2 proxy -kerrokselle, josta ne välittyvät proxy bottien kautta muulle verkolle.

Bottiverkon koneet vaihtavat keskenään peer-listoja, pitääkseen verkon yhtenäisenä. Botit myös vaihtavat binääri- ja konfigurointipäivityksiä keskenään. Ne myös jakavat toisilleen listoja verkon proxy boteista, jonne ne voivat käydä pudottamassa varastetun datan. [AB+13]

Jokasella Zeus-verkon botilla on passiivinen- ja aktiivinen säie. Passiivinen säie on sitä varten, että jokanen verkon botti kuuntelee toistensa viestejä. Aktiivisessa säikeessä botti käy puolen tunnin välein pitämässä tietonsa (esimerkiksi peer-listat, proxy botti -listat ym.) ajan tasalla.

P2P-malli aiheutta hyvien puoliensa lisäksi myös hankaluuksia. Bottiverkon peer-listoja rekursiivisesti läpikäymällä (crawling) on verkkoa mahdollista tutkia. On myös mahdollista myrkyttää bottiverkko väärennetyllä peer-listalla, jolloin verkko voidaan neutralisoida. [AEW15]

4.3 TDL4

TDL4 on laadukas ja monipuolinen bottiverkko-ohjelmisto. Kaspersky Lab:in asiantuntijat Sergei Golovanov ja Igor Soumenkov kuvailivat sitä vuonna 2012 tuhoutumattomaksi bottiverkoksi. Tosin saman yhtiön asiantuntija Ram Herkanaidu pehmentää tätä ilmausta sillä, että TDL4:n tekijät ovat halunneet tehdä tuhoutumattoman Bottiverkko-haittaohjelman. Hän myös huomauttaa, että Kaspersky Lab:in haittaohjelmien torjunta kykenee sen poistamaan koneelta. [Her12].

4.3.1 Historia

TDL-4 on osa TDSS-haittaohjelmaperhettä. Ohjelmiston kehittäjä kutsuu sitä TDL:ksi ja Microsoft Aureoniksi. Nykyinen versio on TDL4, joka ilmestyi kesällä 2010.

Kaspersky Lab löysi ensimmäisen version (TDL1) toukokuussa 2008. Tämä oli huomattavasti uudempia versioita yksinkertaisempi ja nykyiset antivirus -ohjelmat osaavat neutraloida sen. Toinen versio ohjelmasta julkaistiin allkuvuodesta 2009, ja kolmas saman vuoden syksynä. Siihen on koko ajan lisätty toiminnallisuuksia. [GoS11] Bottiverkon koko oli vuonna 2011 4,5 miljoonaa PC-bottia. [Hug11]

4.3.2 Ominaisuudet

TDSS infektoi tietokoneen ajureita, eli se on bootkit-ohjelma. Se ladataan hyvin aikaisessa vaiheessa käyttöjärjestelmän latauksessa. Tämä tekee sen havaitsemisen ja poistamisen hankalaksi.

TDL4 käyttää tietoliikenteen salauksiin omaa XOR swap -teknologiaan perustuvaa salausta (varhaisemmat versiot ohjelmistosta käyttivät salaukseen RC4-teknologiaa). Tällä estetään muita kyberkriminaaleja kaappaamasta verkkoa ja suojelee verkon koneita tietoliikenteen analysoinnilta.

TDL4 oli ensimmäinen bottiverkko-ohjelmisto, joka sisälsi mahdollisuuden infektoida myös 64-bittisiin järjestelmiä. Se toimii myös antivirus-ohjelmana. Se etsii haittaohjelmia saastuneen koneen rekistereitä ja tiedostojärjestelmästä ja poistaa ne. TDL voi poistaa likimain 20 yleistä haittaohjelmaa (näitä ovat muun muassa Zeus ja Gbot). TDL hyötyy tästä kahdella tavalla. Toisaalta se kilpailee muiden haittaohjelmien kanssa, ja samalla estetään mahdolliset haitat siitä, että koneessa on kaksi erillistä

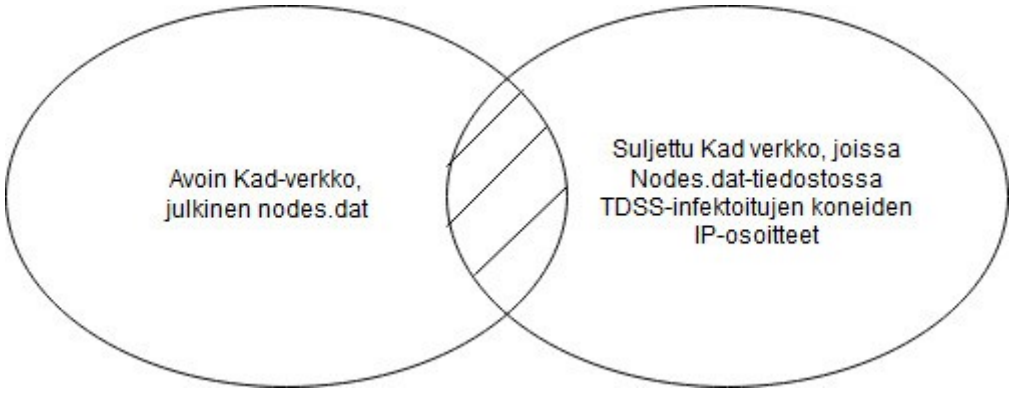
haittaohjelmaa.

TDL4:n innovaatio on julkisen kad-verkon (Kademlia) käyttäminen verkon käskyjen jakamiseen. Kademlia on hajautustaulu (hash table), jota käytetään epäkeskitetyissä P2P-verkoissa.

Ensimmäiseksi luodaan kad-verkon saataville ktzerules-niminen kryptattu tiedosto. TDSS:n saastuttamat koneet käsketään lataamaan ja asentamaan moduuli nimeltä kad.dll. Sen tehtävä on hyökkääjän käskyjen välittäminen. Asennuksen jälkeen kad.dll käy lataamassa nodes.dat -tiedoston, sisältää listan kad-verkon palvelinten ja asiakkaiden ip-osoitteita ja on julkisesti saatavilla.

Tämän jälkeen kad.dll -moduuli lähettää pyynnön löytää ktzerules-tiedosto kad-verkkosta. Kun ktzerules-tiedosto on asennettu koneeseen, kad.dll-moduuli voi ajaa sen sisältämiä komentoja.

Kad-verkon käyttämisessä on myös haittapuolia. Koska ktzerules-tiedosto sijaitsee julkisessa kad-verkossa, se mahdollistaa sen, että muutkin kyberkriminaaleilla on mahdollista muokata näitä komentoja ja saada bottiverkko toimimaan toisin kuin botmaster haluaisi. Kuitenkin vain osa verkon koneista kuuluu julkiseen Kad-verkkoon ja suurin osa on kuuluu suljettuun Kad-verkossa. Julkisessa Kad-verkossa ei ole kuin 10 TDSS-infektoitunutta konetta, mikä tekee ktzerules-tiedoston korvaamisen tai muokkaamisen mahdollisimman epätehokkaaksi. Tämä myös estää muita kyberkriminaaleja kaappaamasta verkkoa itselleen. [GoS11]



Kuva 9: Vain osa bottiverkon koneista kuuluu avoimeen Kad-verkkoon. [GoS11]

Toinen ongelma on GPL-lisenssiehtojen rikkominen. Kehittäessään kad.dll-moduulia tekijät rikkoivat GPL-lisenssin suojaamaa koodia, muokkaamalla sitä tavalla johon heillä ei olisi ollut oikeutta. Tämä tarkoittaa, että tekijät rikkoivat lisenssiehtoja. [GoS11]

4.3.3 Kontrollointi

Ktzerules-tiedosto sisältää seuraavat komennot:

- SearchCfg: etsi Kad-verkosta uusi ktzerules-tiedosto
- LoadExe: lataa ja suorita ajotiedosto
- ConfigWrite: kirjoita cfg.ini -tiedostoon
- Search: etsi tiedostoa Kad-verkosta
- Publish: julkaise tiedosto kad-verkossa
- Knock: lataa uusi nodes.dat -tiedosto C&C -palvelimelle

Knock-komennon avulla hyökkääjä voi luoda oman Kad P2P-verkon TDSS:n saastuttamista koneista.

Bottiverkon ylläpitäjät voivat helposti lisätä verkon koneita julkiseen Kad-verkkoon, ja myös poistaa niitä sieltä. Kad-verkkoon pääseminen mahdollistaa hyökkäjien ladata kaikkia bottien tiedostoja ja laittaa ne jakoon koko verkolle. [GoS11]

4.4 Yhteenveto ohjelmistoista

Yhteenvetoa luvussa käsitellyistä bottiverkoista Alhaalla olevassa taulukossa:

	Storm	Zeus	TDL4
Topologia	P2P, Overnet-verkon hyödyntäminen	Erilaisia versioita, osassa keskitetty, osassa P2P.	P2P, Avoimen Kad-verkon hyödyntäminen
Tärkeimmät hyökkäykset	Roskapostitus	Verkkopankkihyök-käykset	Roskaposti, DOS
Tullut kuuluisaksi	Kuuluisa roskapostituksesta, sillä lähetettiin vuonna 2007 20 kertaisen määrä roskapostia.	Tunnetuin verkkopankki-hyökkäyksiin erikoistunut bottiverkko-ohjelmisto	Erittäin vaikeasti tuhottava bottiverkko

5 Bottiverkkojen havaitseminen ja niiden uhkilta suojautuminen

Tässä luvussa käsitellään erilaisia tapoja puolustautua bottiverkkojen uhkaa vastaan. Ensimmäiseksi käsitellään se, miten on mahdollista välttyä bottiverkkoon joutumiselta, ja sitten käsitellään bottiverkkojen havaitsemista. Lopuksi käsitellään niillä tehdyiltä hyökkäyksiltä puolustautumista.

Tietoturvassa usein ihminen on heikoin lenkki. Tämän takia käyttäjien tietoturvakoulutukseen on tärkeää panostaa, jolloin käyttäjät tulevat tietoisiksi uhista. Oikeanlaisella koulutuksella voidaan nostaa käyttäjien vastuuntuntoisuuta, joka samalla lisää myös tietoturvaa.

5.1 Suojautuminen saastumiselta

Ensimmäinen taso puolustautumisessa on se, että estetään konetta joutumasta osaksi bottiverkkoa, eli konetta ei päästetä infektoitumaan. Tällöin koneen (tai verkon) puolustuksen täytyy olla kunnossa. Koneen puhtaana pitämiseen on monia keinoja:

- käyttöjärjestelmän ja muun ohjelmiston säännöllinen päivittäminen
- tuntemattomista lähteistä tulleiden sähköpostien liitetiedostojen avaamatta jättäminen
- skriptikielten (kuten Javascript, ActiveX) tuki pois päältä tai ainakin niiden

oikeuksien kontrollointi

- selaimen tietoturvataso korkeaksi
- palomuurin ja virustorjuntaohjelman käyttäminen
- käyttöoikeuksien rajoittaminen online-tilassa

[StS09]

5.2 Bottiverkkojen havaitseminen

Bottiverkkoja voidaan havaita joko aktiivisesti tai passiivisesti. Passiivisessa havaitsemisessa pyritään havaitsemaan liikenteestä epäilyttäviä asioita. Sen sijaan aktiivisessa havaitsemisessa pyritään tekemään. Bottiverkkojen havaitseminen voidaan jakaa aktiiviseen ja passiiviseen.

5.2.1 Aktiiviset havaitsemismenetelmät

Hunajapurkit ovat hyvä esimerkki aktiivisesta havaitsemisesta. Ne ovat ansoja, joiden on tarkoitus kerätä epämääräistä liikennettä. Ne on usein tarkoituksellisesti huonosti suojattuja ja sijaitsevat paikassa, missä verkkoliikennettä ei normaalisti pitäisi olla ollenkaan. Kaikki hunajapurkkiin päätyvä liikenne on epäilyttävää ja sitä täytyy tutkia.

Hunajapurkeilla voidaan saada selville tietoa seuraavista asioista [AM+10]:

- bottien signatuureja sisältö-pohjaiseen havaitsemiseen
- tietoa bottiverkon C&C-palvelimesta ja mekanismeista
- tietoa tuntemattomista rei'istä, jotka mahdollistavat järjestelmään tunkeutumisen
- tietoa hyökkääjän käyttämistä hyökkäystyökaluista ja tekniikoista
- tietoa hyökkääjän motivaatiosta

Hunajapurkin toteutukseen on monenlaisia tapoja. Matalan interaktion (Low-interaction) -hunajapurkit tarjoavat emuloituja palveluita ja käyttöjärjestelmiä. Korkean interaktion (High-interaction) hunajapurkit, joita kutsutaan myös hunajaverkoiksi (honeynet) taas ovat kokonaisia fyysisiä järjestelmiä, jotka tarjoavat palveluita ja ohjelmia joihin hyökkääjä voi hyökätä. [HON06]

Hunajapurkeilla on myös rajoitteita. Hunajapurkit eivät varsinaisesti suojaa verkkoa, tosin kun esimerkiksi palomuurit, IDS- (Industrion Detection System) ja IPS- (Intrustion Prevention System) -laitteet. Koska hunajapurkkiin kerääntyy dataa vain skannauksen avulla leviävistä boteista, muunlaisia leviämismenetelmiä käyttävät botnet-ohjelmistot jäävät niiden ulkopuolelle. Hunajapurkit eivät voi mitään esimerkiksi roskapostin tai verkkosivuilta tapahtuvan saastuneen tiedoston lataamisen kautta infektoitumiselle. Lisäksi hunajapurkeilla voi seurata vain rajallista joukkoa erilaisia hyökkäysaktiviteetteja. [AM+10]

5.2.2 Passiiviset havaitsemismenetelmät

Passiiviset havaitsemismenetelmät voidaan ajakaa kahteen perustyyppiin, allekirjoitus-pohjaisiin (Signature-based Detection) sekä anomalioihin perustuviin (Anomaly-based Detection) havaitsemismenetelmiin. [AM+10]

Allekirjoituksiin perustuvassa havaitsemisessa käytetään tietoja tunnetuista bottiverkko-ohjelmistoista. Huono puoli tässä tekniikassa on se, että havainnoinnissa saadaan kiinni vain tunnetut ohjelmistot: uudesta ohjelmistosta kun ei vielä ole havaintoja. Lisäksi jos kahdessa ohjelmistossa on hyvin samankaltainen allekirjoitus (signature), voi toinen näistä jäädä huomaamatta. Esimerkki allekirjoituksiin perustuvasta havainnoimisesta on Snort, joka havaitsee bottiverkko-ohjelmiston toiminnan verkkoliikenteestä. [AM+10]

Anomalioihin perustuvat menetelmissä keskitytään tietoverkon toiminnan poikkeavuuksiin, kuten epäilyttävän suureen liikenteen määrään, väärissä porteissa kulkevaan liikenteeseen, suureen viiveeseen tai muuhun verkon käyttäytymiseen, joka laittaa epäilemään saastumista. [AM+10]

Nimipalvelimiin perustuva havaitseminen (DNS-based Detection) perustuu botnet-verkon tuottamaan DNS-liikenteen monitorointiin. Juuri bottiverkkoon liittetyn koneen on jossakin vaiheessa otettava yhteys C&C-palvelimeen (tätä kutsutaan termillä rallying) [BB+07]. Useimmat botit tekevät tämän DNS-palvelimien avulla. Yleisesti bottiverkko-ohjelmistoilla on DNS-liikenteessä yksilölliset piirteet. [CK+07]

Louhintaan perustuva havaitseminen perustuu bottiverkkojen C&C-palvelinten liikenteen havaitsemiseen. Se on onnistuessaan tehokas tapa havaita bottiverkkojen liikennettä, mutta se ei ole helppoa. Koska liikenne ei ole erityisen suurta ja se usein

muistuttaa normaalia liikennettä, niin anomalioihin perustuvat tekniikat eivät ole tehokkaita. On kuitenkin erilaisia tekniikoita C&C -liikenteen havaitsemiseen. Näihin kuuluu koneoppiminen (machine learning), liikenteen luokittelu (classification) ja klusterointi. [AM+10]

Botsniffer on esimerkki louhintaan perustuvasta havaitsemisesta. Se klusteroi samanlaista liikennettä ja samankaltaista vihamielistä liikennettä. Sitten se suorittaa klusterien ristiintarkistuksen (cross cluster correlation) sellaisten isäntäkoneiden selvittämiseksi, jotka jakavat samanlaiset kommunikaatiomallit ja samanlaiset mallit vihamieliseen toimintaan (malicious activity pattern). [AM+10]

5.3 Hyökkäyksiltä puolustautuminen

Kolmas taso puolustautumisessa on hyökkäyksiltä suojautuminen. Tässä aliluvussa käydään läpi erilaisia suojautumisjärjestelmiä, sekä puolustusta erilaisia hyökkäyksiä vastaan.

5.3.1 Hyökkäyksiltä puolustautuminen

Useimmissa tapauksissa bottiverkoilla tehtävien hyökkäysten kohteena ovat web-palvelut [CJM10]. Suojautumiseen on erilaisia tapoja.

Erityisen tärkeää on ohjelmakoodin pitäminen mahdollisimman turvallisena. Tällöin hyökkääjä ei pääse tunkeutumaan koneelle koodissa olevan haavoittuvuuden kautta. Tietoliikenteessä protokollana kannatta pitää suojattua TLS-protokollaa, jolloin liikenne on paremmin suojassa.

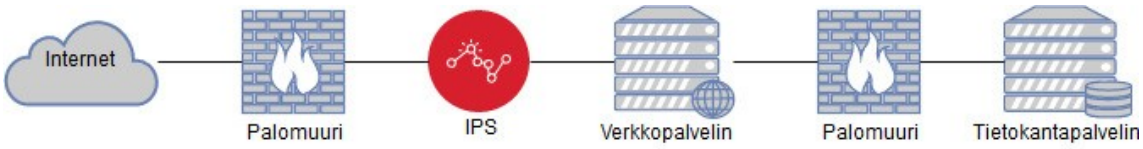
Vihamielistä liikennettä voidaan torjua palomuurin ja erilaisten hyökkäyksen havaitsemijärjestelmien (Intrustion Detection System, IDS) ja hyökkäyksen toimintajärjestelmien (Intrustion of Prevention System, IPS) avulla.

Sisältöön perustuva IPS (Content-based IPS) perustuu siihen, että IPS kontrolloi pakettien sisältöä etsimällä niistä tunnettujen haittaohjelmien allekirjoituksia, ja blokkaa liikenteen jos jonkin se havaitaan olevan epäluotettavasta lähteestä.

IPS pystyy torjumaan sovellusten haavoittuvuuksia hyväksikäyttävät palvelunestohyökkäykset, joiden osuus nykyisin esiintyvistä palvelunestohyökkäyksistä on merkittävä. Verkkoa kuormittavien palvelunestohyökkäyksien torjunnassa IPS-

järjestelmät toimivat hyvin operaattorien tekemän suodatukseen lisänä. [StS09]

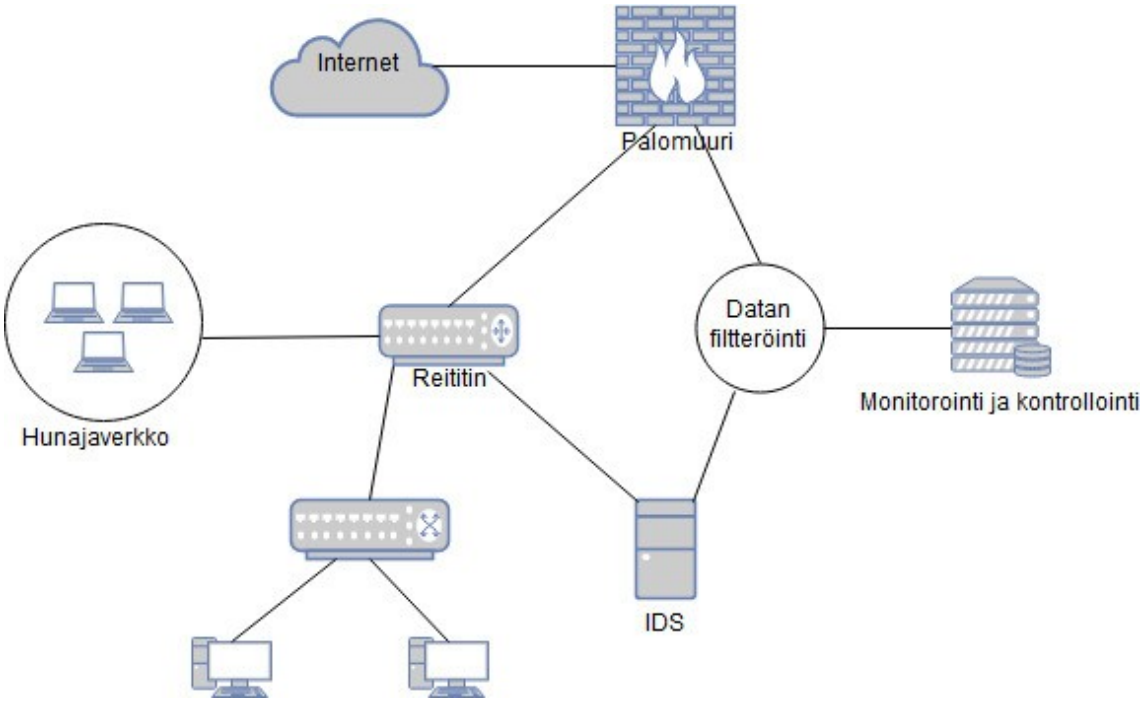
Kuvassa 10 on yhdenlainen versio verkkosivuston suojaamisesta käyttäen apuna palomuureja ja IPS-järjestelmää.



Kuva 10: Verkkopalvelun puolustus, joka on toteutettu palomuurien ja IPS-systeemin avulla. [StS09]

Kuvassa 11 esitellään tapa verkon suojaamiseen yhdistämällä, jossa on mukana palomuuri, IDS-järjestelmä ja hunajapurkki. Koska hunajapurkit eivät varsinaisesti suojaa verkkoa, tarvitaan myös palomuurin ja tunkeutujan havaitsemisjärjestelmän yhteinen puolustus.

Palomuuri on ensimmäinen taso puolustuksessa, joka torjuu osan liikenteestä. Siitä läpipääsevästä liikennettä vartioi hunajapurkki/hunajaverkko raportoi epäilyttävän liikenteen tunkeutujan havaitsemisjärjestelmälle, joka tekee liikenteen torjumisen.



Kuva 11: Suojaus rakennettu käyttäen palomuuria, IDS-systeemiä ja hunajapurkkia. [CJM10]

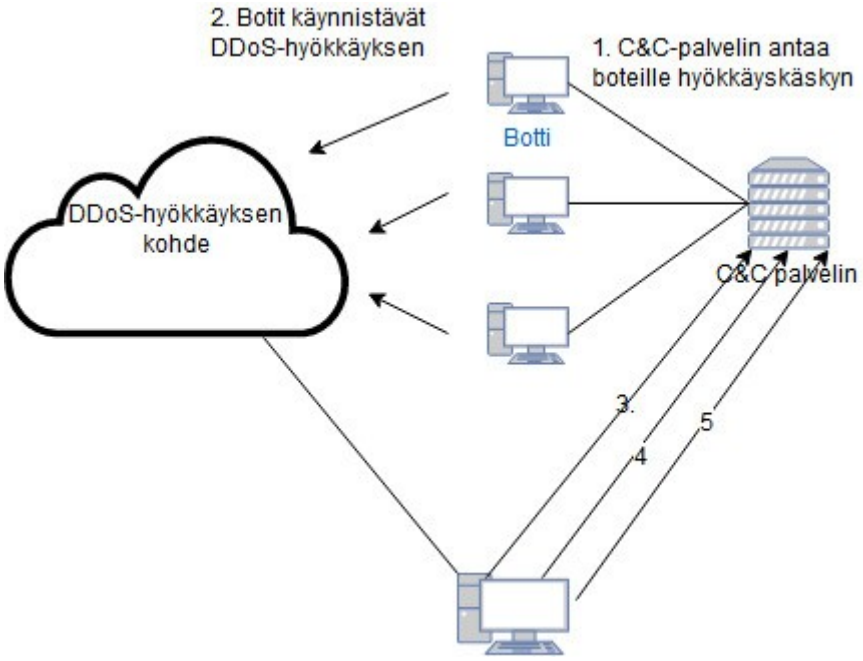
IPS-järjestelmät tarkkailevat verkkoa jatkuvasti ja etsivät vihamielistä liikennettä. Näin ne takaavat liiketoimintakriittisille järjestelmille niiden tarvitseman kapasiteetin. Ne vaikuttavat myös tietoturvan tilannekuvan muotoutumiseen.

Lähtessä [CK+14] käsitellään erilaista tapaa puolustautua pankkihyökkäysiä tekeviä bottiverkkoja vastaan. Tässä tavassa hyödynnetään vuodetun (leaked) Zeus 2.0.8.9 työkalupakin C&C-palvelimien haavoittuvuuksia.

Testit toteutettiin analysoimalla haittaohjelman tietoliikennettä ja testaamalla kaikkia HTTP GET -pyyntöihin perustuvia haavoittuvuuksia tunnettuihin Zeus C&C-palvelimiin, joiden IP-osoitteet oli saatu Zeus Tracker -sivustolta. HTTP POST -pyyntöihin perustuvia haavoittuvuuksia ei testattu sen vuoksi, että sellaiset palvelimet, joilla C&C pyörii omistajan tietämättä, eivät joutuisi ongelmiin.

Testauksessa havaittiin, että tämän Zeus-haittaohjelman versiossa oli haavoittuvuuksia. Testauksessa havaittiinkin, että C&C-palvelimista löytyi ainakin kolme haavoittuvuutta: Bufferien ylivuoto, riittämätön käyttöoikeuksien valvonta sekä haavoittuvat autentikointimekanismit. [CK+14]

Samat tutkijat käsittelevät samankaltaisen puolustautumisen laajentamista DDOS-



Kuva 12: Palvelunestohyökkäyksen torjuminen hyökkäämällä C&C-palvelimia vastaan [MS+15]

hyökkäyksiä tekeviin bottiverkkoihin. Tässä artikkelissa perehdytään Dirt Jumper -DDOS työkaluperheen analysointiin ja haavoittuvuuksien etsintään. Dirt Jumper-bottiverkot ovat tehokkaita työkaluja DDOS-hyökkäyksiin. [MS+15]

5.3.2 Bottiverkkojen lähettämän roskapostin torjuminen

Bottiverkot ovat niin suuri ongelma, että niiden torjumiseen nähdään valtio- ja yritystasoilla paljon vaivaa ja käytetään paljon resursseja. Tämän vuoksi nykyään bottiverkoilta lähtevää roskapostia pystytään torjumaan paremmin.

Ongelmana bottiverkkojen lähettämässä roskapostissa on se, että sähköpostit eivät lähde oikeista sähköpostipalvelimista, vaan lähetystä varten generoiduista osoitteista. Tällöin lähettäjien IP-osoitteet eivät ole oikeiden sähköpostipalvelimien osoitteita. Tämän takia ne voidaa tunnistaa IP:n tarkistustekniikoilla (IP check) ja sähköpostipalvelimien sormenjälkitekniikoilla. [Sal09]

Yksi yleisistä tekniikoista roskapostin torjumisessa ovat mustat listat (blacklists), joissa on listattu tunnettuja IP-osoitteita, joista lähtee roskapostia. Yksi tapa torjua roskapostia

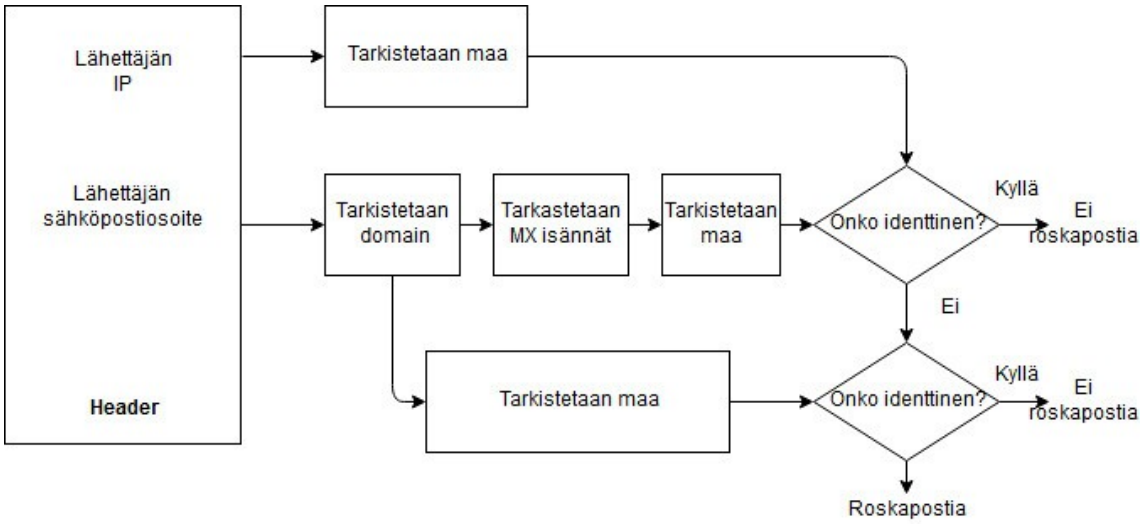
on blokata listalla olevista IP-osoitteista lähtevät postit. Tämän kuitenkin tekee ongelmalliseksi se seikka, että eri palvelut voivat käyttää jaettua hosting-palvelua. Tällöin jos tämän IP-osoiteelta lähtevä posti torjutaan, voi filtteriin jäädä myös normaalia postia.[Sal09]

Roskapostifiltterit pyrkivät tunnistamaan erilaisilla tavoilla viestin roskapostiksi. Yksi tapa niiden huijaamiseen on sellainen, että muokataan epäilyttäviä usein roskaposteissa esiintyviä sanoja eri näköiseksi (esimerkiksi sana ”Viagra” muokataan merkkijonoksi ”V I agra”). Kuitenkin muunnokset on tehty siten, että sanat ovat vielä ymmärrettävissä. Viestit voidaan myös koostaa templaateista, joissa on samankaltainen rakenne, mutta osa sanoista vaihtuu.[Sal09]

Toinen tapa huijata roskapostisuodattimia on kuviin perustuva roskapostitus. Siinä tekstit piilotetaan kuviin. Biggio ja muut esittelivät tekniikoita, millä tällaiset kuviin perustuvan roskapostituksen voi suodattaa. [BF+08] Roskapostittajien vastaus tähän ollut se, että kuviin on lisätty taustakuvaa ja taustameteliä (noise), jolloin tekstiä on vaikea tunnistaa kuvan seasta.

Yksi tapa erottaa roskapostia oikeasta postista on analysoida viestien header-ositoita. Headerista luetaan lähettäjän IP-osoite, sekä lähettäjän sähköpostiosoitte. Lisäksi tarkastellaan DNS tallennusta (DNS record), joka sisältää MX-tallennuksia (MX records), joissa määritellään MX isännät, eli se on lista postipalvelimista, joita jollain domain palveluntarjoajalla on vain yksi, jollain (esim. Hotmail, Yahoo, Gmail) on niitä useampia. [Sal09]

Kuva 13: Roskapostin tunnistaminen sen perusteella, onko lähettäjän IP-osoite ja sähköpostin domain samasta maasta [SaL09]



Kuvassa 13 kuvaataan järjestelmän toiminta [SaL09]

1. Puretaan header-osiosta lähettäjän IP-osoite ja lähettäjän sähköpostiosoite

2. Tarkistetaan IP-osoitteen perusteella maa, jossa kyseinen IP-osoite sijaitsee

3. Analysoidaan lähettäjän sähköpostiosoite, tutkitaan lähettäjän (Sender) domain, sekä lähettäjän MX isännät (MX Host)

4. Tarkistetaan, mistä maassa lähettäjän domain on

5. Tutkitaan MX -isäntiä, ja katsotaan, mikä maa niissä on määritelty

6. Mikäli kohdan 2 ja kohdan 4 ovat samat, viesti ei ole roskapostia - jos ovat eri, verrataan kohdan 2 tulosta kohdan 5 tulokseen, mikäli samat, ei ole roskapostia.

Yksi tapa erotella roskaposti tavallisesta postista on analysoida viestien muotoja (shape). Paul Stroufe ym. esittelevät sähköpostien muotoon perustuvan bottiverkon havaisijan (Email Shape based Botnet Detector, EsBod). Saman templaatin kautta luotujen templaattien voi havaita muistuttavan toisiaan muodoltaan, ilman että viestin sisältöä tarvitsee erikseen lukea. [CD+09]

6 Yhteenveto

Bottiverkot-ohjelmistot ovat haittaohjelmia. Ne erottaa muista haittaohjelmista se, saastuneet koneet kootaan verkoksi, jota verkon hallitsija voi hallita ja antaa niille komentoja. Verkossa on komentokeskuksia, jotka ohjaavat komennot verkon koneille.

Bottiverkot ovat suuri bisnes. Bottiverkkoja pitää myös suojata paitsi uhreilta, niin myös muilta kyberkriminaaleilta. Kilpailu on kovaa ja esimerkiksi TDL4 sisältää oman virustorjuntaohjelmiston, joka pitää koneen siistinä muilta haittaohjelmilta.

Bottiverkoilla voi toteuttaa erilaisia hyökkäyksiä. Erityisen hyvin suuret bottiverkot sopivat roskapostitukseen ja hajautettuihin palvelunestohyökkäyksiin. Näissä hyökkäyksissä on paljon hyötyä siitä, että hyökkäys toteutetaan monesta lähteestä. Muita hyökkäystapoja ovat

Alunperin bottiverkko-ohjelmistot käyttivät viestimiseen IRC-protokollaa, mutta se voidaan torjua sulkemalla palomuurista liikenteen käyttämä portti, jolloin liikenne ei pääse kohteeseen. Toinen tapa on käyttää HTTP-protokollaa, jolloin viestit kulkevat tavallisen liikenteen seassa, jolloin sitä ei voi blokata samalla tavalla, ja bottiverkon liikennettä on vaikeampi havaita. Jotkin botnet-ohjelmistot käyttävät muita protokollia tärkeässä tietoliikenteessä, tästä esimerkkinä luvussa 4.4 käsitelty TDL4, joka suoritti osan verkon liikeenteestä Kad-protokollaa hyödyntäen siten, että osa verkon koneista kuuluio julkiseen Kad-verkkoon. Storm worm taas käytti viestittelyyn Overnet-protokollaa.

Bottiverkkojen topologia voi olla keskitetyn mallin mukainen, jossa on yksi tai useampi komentokeskus, jotka välittävät käskyt verkon koneille. Tämän mallin hyvä puoli on se, että komennot saadaan välitettyä yksinkertaisesti verkon koneille. Malli on haavoittuvainen sille, että yksittäinen komentokeskus saadaan pois pelistä. Tällöin viestit eivät välity verkon koneille. Varhaisemmat bottiverkot oli aina järjestetty keskitetyn mallin mukaisesti.

Toinen tapa verkon toteutukseen on tehdä verkosta P2P-verkko, jolloin sillä ei ole erillisiä C&C -palvelimia vaan verkon koneet toimivat sellaisina. Tällöin verkko ei ole haavoittuva yksittäisten koneiden poistamisen vuoksi. Vaikka P2P-verkon ylläpito on raskaampaa, suurin osa uudemmista bottiverkoista on sellaisia. Verkot voivat olla joko

kokonaan, tai osittain hajautettuja. On myös olemassa erilaisia hybridiratkaisuja.

Tässä opinnäytetyössä on vertailtu kolmea bottiverkko-ohjelmistoa, joilla kullakin on omat erityispiirteensä. Storm Worm -ohjelmisto oli pahimmillaan suurimpia roskapostittajia. Se on UDP-pohjainen P2P-verkko ja se käyttää tietoliikenteeseen muokattua versiota Overnet P2P -protokollasta.

TDSS-sarjan neljäs versio TDL4-ohjelmistosta on erittäin vaikeasti havaittava ohjelmisto. Kasperskyn asiantutuntijat kuvailivat sitä “tuhoutumattomaksi bottiverkoksi”. TDL4 on bootkit-ohjelma, jolloin se ladataan aina koneen käynnistyessä ennen käyttöjärjestelmää. Tietoliikenteeseen käytetään Kad-protokollaa, mikä tekee siitä erittäin vaikeasti havaittavan. Osa verkon koneista kuuluu julkiseen kad-verkkoon, jonne komentojen jakaminen tapahtuu. Suurin osa verkon koneista kuuluu kuitenkin kad-sisäverkkoon, joille julkisessa verkossa olevat koneet nämä komennot ohjaavat.

Zeus ei ole varsinaisesti tarkoitettu suuriin roskapostituksiin tai palvelunestohyökkäyksiin, vaan se on tullut kuuluisaksi siitä että sen avulla on tehty hyökkäyksiä verkkopankkeja vastaan. Tähän tarkoitukseen se aluperin tehtiinkin. Zeus sisältää työkaluja, joilla voi varastaa salasanoja ja pankkitunnuksia. ZEUS-ohjelmistoja on liikkeellä lukuisia erilaisia, joissa on erilaisia ominaisuuksia, ja joilla on erilaisia tapoja tietoliikenteeseen.

Tässä opinnäytetyössä on keskitytty P2P-protokollaa käyttävään Gameover Zeus ohjelmistoon. Gameover Zeus suorittaa käskyjen viemisen boteille kolmella tasolla.Ylin taso on C2 taso, jonne komennot tallennetaan. Keskitaso on C2 proxytaso, jonka kautta alimmalla tasolla, eli varsinaisessa bottiverkossa sijaitsevat proxy botit hakevat ne itseleen ja levittävät ne verkon muille koneille HTTP-protokollan avulla.

Tässä opinnäytetyössä on jaettu bottiverkoilta suojautuminen kolmeen tasoon: saastumisen välttämiseen, bottiverkkojen havaitsemiseen ja hyökkäyksiltä puolustautumiseen.

Bottiverkkoon joutumisen ehkäisemisessä on tärkeää, että tietokoneen tietoturva on kunnossa, eli selaimen turvataso on riittävällä tasolla, skriptikielten (esim. ActiveX) oikeuksien kontrollointi tai tuki niille kokonaan pois päältä ja käyttöjärjestelmän ja ohjelmien säännöllinen päivittäminen. Lisäksi pitää olla palomuuri ja virustorjuntaohjelma. Tärkeää on myös käyttäjän oma toiminta, eli esimerkiksi

epämääräisten sähköpostien liitteitä ei pidä avata.

Toinen taso on bottiverkkojen toiminnan havaitseminen. Sen voi jakaa kahteen osaan, passiiviseen ja aktiiviseen. Hunajapurkit tekevät aktiivista havaitsemista houkuttelemalla haitallista liikennettä paikkoihin, joissa sitä ei pitäisi olla. Passiiviset menetelmät taas tyytyvät tekemään havaintoja liikenteestä.

Kolmas taso suojautumisessa on hyökkäyksiltä puolustautuminen. Tässä opinnäytetyössä on esitelty malleja, miten voidaan toteuttaa turvallinen järjestelmä, esimerkiksi yhdistämällä palomuuri, hunajapurkki ja monitorointi.

Bottiverkkojen kautta lähtevä roskapostitus on ongelmallista siinä, että siltä suojautuakseen ei voi vain blokata yhdestä lähteestä tulevia viestejä. Kuitenkaan bottiverkkojen lähettämä sähköposti ei lähde oikeista sähköpostipalvelimista vaan osoitteet on luotu lähettämistä varten. Täten ne voidaan tunnistaa IP-osoitteiden tarkistustekniikoilla sekä sähköpostipalvelimien sormenjälkitekniikoilla.

Pahimmista roskapostittajista pidetään musita listoja, joiden kautta lähtevät viestit voidaan blokata. Muita tapoja bottiverkkojen roskapostitukselta suojautumiseen on muotoihin perustuva tunnistaminen, joilla viestin muotoa analysoimalla se voidaan tunnistaa jollakin templaatilla tehdyksi. Toinen on viestien header-osioiden tutkiminen, jolloin havaitaan että vieti ei ole lähtenyt sieltä mistä se väittää lähteneensä.

7 Lähdeluettelo

BB+07 Bradley Tony, Binkley Jim, Evron Gadi, Harley David, Schiller Craig A.,Willems Carsten, Cross Michael 2007:Botnets: The Killer Web Applications, Published February 15th 2007 by Syngress Publishing http://books.google.fi/books?id=4MAuVjOx6kIC&pg=PA37&lpg=PA37&dq=botnet+rallying+process&source=bl&ots=vdqwO2aAkX&sig=1m4DgX3g3v3bzSmNBgk5I3QGwvw&hl=en&sa=X&ei=-kSnUdXdEpS10QWA4oD4AQ&redir_esc=y#v=onepage&q=botnet%20rallying%20process&f=false

JLZ09 Jiang Wei, Li Chao, Zou Xin 2009: Botnet: Survey and Case Study, *2009 Fourth International Conference on Innovative Computing, Information and Control*
https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=5412718

Pur03 Puri Ramneek 2003: Bots & Botnet: An Overview, SANS Institute, GSEC Practical Assignment Version 1.4b
http://www.sans.org/reading_room/whitepapers/malicious/bots-botnet-overview_1299 [28.4.2018]

StS09 Stankovic Srdjan, Simic Dejan 2009: Defense Strategies Against Modern Botnets, *(IJCSIS) International Journal of Computer Science and Information Security, Vol. 2, No. 1*, *2009* http://arxiv.org/pdf/0906.3768v2 [30.4.2018]

ThR16 Thangapandiyan M. Rubesh Anand P.M. 2016: An efficient botnet detection system for P2P botnet, *Wireless Communications, Signal Processing and Networking (WiSPNET), International Conference on, 23-25.5.2016*, https://ieeexplore.ieee.org/document/7566330/ ????

SWZ08 Sparks Sherri, Wang Ping, Zou Cliff C. 2008: An Advenced hybrid Peer-to-Peer Botnet, *IEEE Transactions on Dependable and Secure Computing (Volume: 7, Issue: 2, April-June 2010), 18.6.2008* https://ieeexplore.ieee.org/document/4569852/

CLY17 Cui Xiang, Li Ke, Yin Jie 2017: A Reputation-Based Resilient and Recoverable P2P Botnet, *Data Science in Cyberspace (DSC), 2017 IEEE Second International Conference on 26-29.6.2017*, https://ieeexplore.ieee.org/document/8005486/

Oll09A Ollmann Gunter, Botnet 2009: Communication Topologies, 4.6.2009, *http://www.technicalinfo.net/papers/PDF/WP_Botnet_Communications_Primer_(2009-06-04).pdf* [28.4.2018]

MPZ04 Masikos Michalis, Patrikakis Charalampos, Zouraraki Olga 2004: Distributed Denial of Service Attacks, *The Internet Protocol Journal vol 7 num 4* http://www.cisco.com/web/about/ac123/ac147/archived_issues/ipj_7-4/dos_attacks.html [28.4.2018]

Vau10 Vaugham-Nichols Steven J. 2010: The botnet business, *IT World 6.1.2010,* http://www.itworld.com/security/106428/the-botnet-business?source=inline

Kam08 Kamluk Vitaly 2008: The botnet business. *Kaspersky Securelist* 13.5.2008 http://www.securelist.com/en/analysis/204792003/The_botnet_business [28.4.2018]

DDM11 DeSantis Matthew, Dougherty Chad, McDowell Mindi 2011: Understanding and Protecting Yourself Against Money Mule Schemes, *2011 Carnegie Mellon University. Produced for US-CERT, a government organization, https://www.us-cert.gov/sites/default/files/publications/money_mules.pdf*

GeP12 Gerhards-Padilla Elmar, Plohmann Daniel 2012: Case Study of the Miner Botnet, 2012 4th International Conference on Cyber Conflict,

https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=6243985

FRS09 Feily Maryam, Ramadass Sureswaran, Shahrestani Alireza 2009: A Survey of Botnet and Botnet Detection, *2009 Third International Conference on Emerging Security Information, Systems and Technologies* http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=5210988

Dan09 Danchdev Dancho 2009: Research: Small DIY botnets prevalent in enterprise networks, *ZDNet September 29, 2009*, https://www.zdnet.com/article/research-small-diy-botnets-prevalent-in-enterprise-networks/

Wor Worms, *F-Secure artikkeli,* https://www.f-secure.com/en/web/labs_global/worms

Vir Viruses, *F-Secure artikkeli,* *https://www.f-secure.com/en/web/labs_global/viruses*

Tro Trojans, *F-Secure artikkeli,* https://www.f-secure.com/en/web/labs_global/trojans

EnW07 Eng Chris, Wysopal Chris 2007*:* Static Detection of Application Backdoors, *Datenschutz und Datensicherheit 34(3)T 2007 ,* https://www.researchgate.net/publication/251850474_Static_Detection_of_Application_Backdoors

MeP04 Meyer L., Penzhorn W.T. 2004: Denial of service and distributed denial of service-today and tomorrow, *IEEE Africon 2004* http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=1406828

LPR07 Leckie Christopher, Peng Tao, Ramamohanarao Kotagiri 2007: Survey of network-based defense mechanisms countering the DOS and DDOS problems, *Journal ACM Computing Surveys (CSUR) Volume 39 Issue 1,*

2007 Article No. 3, http://dl.acm.org/citation.cfm?id=1216370.1216373&coll=DL&dl=ACM&CFID=125223524&CFTOKEN=11965950

CC+13 Shih-Jen Chen, Yao-Hsin Chen, Chia-Heng Li, Fu-Hau Hsu, Yan-Ling Hwan, Chuan-Sheng Wangg 2014: Hawkeye: Finding Spamming Accounts, *Network Operations and Management Symposium (APNOMS), 2014 16th Asia-Pacific* http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=6996106

GI+08 Giffin Jonathon, Irani Danesh, Pu Calton, Webb Steve 2008: Evolutionary study of phishing, *eCrime Researchers Summit, 2008, 15-16 Oct. 2008* http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=4696967

Smi08 Smith Brad 2008: A Storm (Worm) Is Brewing, *Computer (Volume: 41, Issue: 2, Feb. 2008)* http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=4454392

Ste07 Stewart Joe 2007: Storm Worm DDoS Attack, *Secureworks threat analysis 8.2.2007* https://www.secureworks.com/research/storm-worm [30.4.2018]

BZ+09 Binbin Wang, Hao Tu, Jun Hu, Zhengbing Hu, Zhitang Li 2009: Actively Measuring Bots in Peer-to-Peer Networks, *Networks Security, Wireless Communications and Trusted Computing, 2009. NSWCTC '09. International Conference on 25-26 April* 2009, https://ieeexplore.ieee.org/document/4908338/ ???

Tar10 Tarakanov Dmitry 2010: ZeuS on the Hunt, Kaspersky *Securelist 12.4.2010*, http://www.securelist.com/en/analysis/204792107/ZeuS_on_the_Hunt [30.4.2018]

AEW15 Alazab Mamoun, Etaher Najla, Weir George R.S. 2015: From ZeuS to

Zitmo: Trends in Banking Malware, *Trustcom/BigDataSE/ISPA, 2015 IEEE, 20-22.8.2015,* http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=7345443

AB+13 Andriesse Dennis, Bos Herbert, Plohmann Daniel, Rossow Christian, Stone-Gross Brett 2013: Highly Resilient Peer-to-Peer Botnets Are Here: An Analysis of Gameover Zeus, *Malicious and Unwanted Software: "The Americas" (MALWARE), 2013 8th International Conference on 22-24.10.2013* http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=6703693

Her12 Herkanaidu Ram 2011: TDL-4 Indestructible or not???, *Kaspersky Securelist 4.6.2011* http://www.securelist.com/en/blog/516/TDL_4_Indestructible_or_not [30.4.2018]

Hug11 Hughes Jeff 2011: TDL-4 creates 4.5 million PC 'indestructible' botnet, *Digital Trends 29.6.2011*, https://www.digitaltrends.com/computing/tdl-4-enslaves-over-4-5-million-pcs-into-indestructible-botnet/

GoS11 Golovanov Sergey, Soumenkov Igor 2011: TDL4 – Top Bot, *Kaspersky Securelist 27.6.2011* http://www.securelist.com/en/analysis/204792180/TDL4_Top_Bot

AM+10 Vahdani Amoli, Payam Safari Majid, Mohammad Jorjor Zadeh Shooshtari, Mazdak Zamani, Hossein Rouhani Zeidanloo 2010: A Taxonomy of Botnet Detection Techniques, *Computer Science and Information Technology (ICCSIT), 2010 3rd IEEE International Conference on 9-11.6.2010* http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=5563555

CK+07 Hyunsang Choi, Hyogon Kim, Hanwoo Lee, Heejo Lee 2007: Botnet Detection by Monitoring Group Activities in DNS Traffic, *Computer and Information Technology, 2007. CIT 2007. 7th IEEE International*

Conference on 16-19.10.2007 http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=4385169&tag=1

CJM10 Chang-peng Ji, Jian Bao, Mo Gao 2010: Research on network security of defense based on Honeypot, *Computer Application and System Modeling (ICCASM), 2010 International Conference on 22-24.10.2010* http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=5622780

CK+14 Corbett Cherita, Kawka Christina, Watkins Lanier, Robinson William H. 2014: Fighting Banking Botnets By Exploiting Inherit Command and Control Vulnerabilities, *Malicious and Unwanted Software: The Americas (MALWARE), 2014 9th International Conference on 28-30.10.2014* https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=6999411

MS+15 Morales Jose Andre, Silberberg Kurt, Watkins Lanier, Robinson William H. 2015: Using Inherent Command and Control Vulnerabilities To Halt DDoS Attacks, *Malicious and Unwanted Software (MALWARE), 2015 10th International Conference on 20-22.10 2015* http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=7413679&tag=1

BF+08 Biggio, B, Fumera, G, Pillai, I, Roli, F 2008: Improving Image Spam Filtering Using Image Text Features, *Fifth Conference on Email and Anti-Spam (CEAS 2008) 21.8.2008* http://pralab.diee.unica.it/en/node/777

SaL09 Saraubon Kobkiat, Limthanmaphon Benchaphon 2009: SauFast Effective Botnet Spam Detection, *Computer Sciences and Convergence Information Technology, 2009. ICCIT '09. Fourth International Conference on 24-26.11.2009* http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=5370890

CD+09 Cangussu Joao, Dantu Ram, Phithakkitnukoon Santi, Sroufe Paul 2009: Email Shape Analysis for Spam Botnet Detection, *Consumer Communications and Networking Conference, 2009. CCNC 2009. 6th IEEE*

10-13.1.2009 http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=4784781