Botnet Tracking

Álvaro Esteban Gutiérrez

Resum—Una Botnet és una xarxa d'ordinadors anomenats Bots, controlats sense autorització del propietari i de forma remota per un agent extern maliciós anomenat Command & Control. Aquestes xarxes son conegudes principalment pels seus atacs de denegació de servei distribuït (DDoS) essent aquesta la motivació per trobar nous mecanismes de detecció d'aquestes xarxes i així ser capaç de minimitzar l'impacte d'un atac.

Durant aquest treball de final de grau, es volen identificar mètodes que puguin detectar la infecció d'una màquina a nivell de xarxa a través d'un estudi previ de la estructura de comunicació establerta amb el Command & Control, els missatges emprats i les dades extretes d'una mostra. Aquests mètodes, finalment han estat definits i gràcies a la implementació d'un d'ells a l'IDS Snort s'ha pogut demostrar la seva validesa.

Paraules clau—Botnet, Bot, C&C, ZeuS, Tracking, Wireshark, Snort, Honeypot, Malware.

Abstract—A Botnet is a network of computers called Bots, controlled without authorization of the owner and remotely by an external and malicius agent called Command & Control. These networks are known primarily for their distributed denial of service atack (DDoS), being this the motivatio to find new mechanisms to detect these networks and then be able to minime the impact of an atack.

During this final project work, I want to identify methodes which can detect the infeccion of a machine at a network level throught a prior study of the communication structure established with the Command & Control, the messages used and the data extracted from a sample. These methods have finally been defined and with the implementation of one of them in the Snort IDS I have been able to demonstrate its validity.

Index Terms— Botnet, Bot, C&C, ZeuS, Tracking, Wireshark, Snort, Honeypot, Malware .

– – – – – – – – – –

- *E-mail de contacte: alvaro.estebang@e-campus.uab.cat*
- *Menció realitzada: Enginyeria de Tecnologies de la Informació.*
- *Treball tutoritzat per: Guillermo Navarro (UAB).*
- *Curs 2014/2015*

– – – – – – – – – – ◆ – – – – – – – – – –

1 INTRODUCCIÓ

AQUEST treball ha estat una proposta per part del professor Guillermo Navarro per analitzar el comportament d'una Botnet i fer un seguiment d'aquesta.

1.1 Motivació del treball

El gran creixement d'Internet durant l'última dècada i mitja ha facilitat l'increment d'atacs on-line, essent l'atac de denegació de servei (DoS) un dels més potents i nocius. Dintre dels atacs de denegació de servei, trobem l'atac distribuït (DDoS) com el de més envergadura.

Un atac de denegació distribuït es llençat per Botnets, grans xarxes d'ordinados anomenats Bots interconectats i controlats per un usuari maliciós. Aquests atacs normalment estan dirigits cap a aplicacions Web, amb l'objectiu de bloquejar recursos o degradar el rendiment del servei utilitzat per la màquina objectiu. Altres objectius poden ser aconseguir popularitat dintre del mon hacker, trencar la confidencialitat d'un servei, etc.

És una tendència creixent el número d'atacs de denegació de servei durant els últims anys i per tant l'ús de Botnets per fer aquests atacs. La motivació pel desenvolupament del treball és veure com es realitza la comunicació entre els Bot i el C&C per poder trobat mètodes de descobriment d'infecció i ajudar a la securització d'Internet.

1.2 Motivació personal

Personalment, tot i que la investigació de Botnets està a l'ordre del dia, no s'ha publicat un volum ampli d'informació a baix nivell d'aquestes i es per això que trobo molt interessant el fet de poder veure la comunicació i el seu modus operandi i ser capaç de trobat mètodes de detecció.

Un altre de les meves motivacions principals és que aquest projecte pot tractar dades o enfocar-les des d'un punt de vista que no hagi estat analitzat abans i per tant, poder arribar a ser l'inici d'un nou mètode real de detecció d'infecció.

Finalment, l'última motivació a esmentar és l'interés que sempre he mostrat per la seguretat informàtica i les possibilitats que hem dóna aquest treball per poder ampliar els meus coneixements.

1.3 Objectius del projecte

Aquest projecte es tracta d'un treball d'anàlisis i investigació, on es portarà a terme la infecció de forma involuntària o provocada i seguidament la investigació de les comunicacions.

Els objectius fixats pel treball es troben detallats a continuació:

1. El primer objectiu comporta la creació d'un escenari, format per un monitor de xarxa i una màquina capaç de ser infectada de forma externa per una Botnet. Aquest escenari ha de permetre una fàcil monitorització, atès que es farà un anàlisi exhaustiu de tot el tràfic i per tant ens facilitarà la feina a l'hora d'analitzar aquestes dades un alt percentatge de tràfic legítim i de valor.

2. El segon objectiu, en cas de no aconseguir ser infectat per un agent extern, implica la instal·lació del Command & Control d'una xarxa Botnet i la posterior infecció voluntària d'una màquina essent aquesta monitoritzada.

3. El tercer i últim objectiu serà el més extens i on s'ubica el contingut més important del projecte. En aquest apartat farem una investigació de la comunicació que s'estableix entre el Command & Control i el Bot, quan aquest està en estat passiu. Seguidament a partir del mostreig realitzat, s'analitzaran les dades extretes donant com a resultat una sèrie de conclusions. Finalment, es definiran mètodes vàlids per la detecció d'una infecció a temps real a nivell de xarxa.

Aquests seran els tres objectius principals del projecte, essent el tercer el més rellevant i important.

1.4 Treballs relacionats

Donat que el camp de les Botnets té un recorregut intermedi, existeixen treballs relacionats amb el tracking de Botnets havent-n'hi informació relacionada.

Existeixen organitzacions com "The Honeynet Project" , fundada a l'any 1999, dedicada a la investigació i seguiment dels últims atacs i al desenvolupament d'eines open source amb l'objectiu de millorar la seguretat global d'Internet. Aquesta organització a contribuït amb la lluita contra malware i atacs maliciosos de hacking, així com la conscienciació i educació del públic de les amenaces i vulnerabilitats que existeixen a Internet avui en dia.

Es poden trobar també pàgines com zeustracker.abuse.ch on es pot, a temps real, fer un seguiment dels servidors Command&Control de ZeuS i hosts maliciosos que contenen arxius d'aquest tipus de Bot. Així mateix, també proveeixen d'arxius amb llistes negres amb les adreces on es troben aquests agents maliciosos per poder detectar infeccions o preveure-les.

Per una altra part, existeix documentació que aprofundeix una mica més en la comunicació d'aquestes Botnets com pot ser l'informe que va presentar el laboratori de Kaspersky "Botnet Traffic Detection", on entra més en detall en el tipus de comunicació que n'hi ha entre el C&C i els Bots en una estructura jeràrquica, no P2P, com es tractarà al llarg d'aquest projecte. A més, proposa un mètode basat en HMM (Hidden Markov Model) per fer un càlcul de probabilitats i determinar si existeix una

infecció.

Finalment n'hi ha treballs de recerca com el presentat per ChulWoo Park, HyoSung Park i KiChang Kim, on s'investiga, a través de les debilitats dels algoritmes de xifratge de les comunicacions, com detectar en temps real i fent un mostreig de paquets, si existeix una infecció o per contra es tràfic legítim.

Podem veure que el camp de detecció i seguiment de Botnets ha estat i és un tema candent i per tant existeix informació d'on podré ajudar-me pel desenvolupament del meu treball.

1.5 Continguts

Els continguts que es veuran de forma detallada al llarg del paper expliquen primerament que és una Botnet i de quines parts consta. Seguidament es parlarà de l'inici del projecte i quins seran els passos a seguir pel correcte desenvolupament d'aquest.

Un cop hem detallat la metodologia, començarem amb la consecució del primer objectiu, establint un escenari que hem permeti la creació d'un Honeypot capaç d'ésser infectat i que pugui ser monitoritzat alhora.

Com el projecte contempla la alterativa d'infectar de forma voluntària el honeypot en cas de no aconseguir-ho de forma externa, s'explicarà com ha estat el procés que ha portat a fer-ho i com s'afronta.

Per començar la via alternativa s'explicarà com ha estat la instal·lació del servidor del Command & Control de ZeuS i la infecció voluntària a la màquina objectiu.

En el moment que s'estableix l'escenari final i s'instal·la ZeuS en ambdues parts, al servidor i al Bot, es passarà a la fase d'investigació on es realitzarà un anàlisi de com és la comunicació entre el C&C i el Bot i es farà una extracció de dades que serviran a la postra per trobar mètodes de detecció.

A continuació de l'extracció de dades i de l'anàlisi fet, es detallaran els tres mètodes trobats per detectar una infecció d'una Botnet a nivell de xarxa i les dificultats que es poden trobar a l'hora d'implementar-les en un cas real.

Per finalitzar i corroborar els mètodes extrets, es farà la prova d'un d'ells amb l'ajuda de l'IDS Snort, prèviament instal·lat, i s'explicaran els resultats obtinguts.

1.6 Planificació i metodologia

La metodologia que s'ha seguit pel correcte desenvolupament del projecte i la consecució dels objectius plantejats, ha dividit el treball en tres grans parts:

1. Recerca inicial d'informació de Botnets, els tipus existents, atacs comuns i tècniques de detecció
2. Establiment de l'escenàri que permetrà l'establiment d'un Honeypot, la seva monitorització i la posterior infecció.
3. En cas de no ser infectat de forma externa, establiment d'un Command & Control i infecció de forma voluntària.
4. Monitoreig de l'activitat que n'hi ha entre el Bot i el C&C mitjançant un sniffer de xarxa. En aquesta fase s'utilitzarà Wireshark.
5. Anàlisis de l'estructura de la comunicació i de les dades extretes. Establiment de mètodes de detecció d'infecció i corroboració mitjançant la implementació d'un d'ells.

Pel que fa a la planificació seguida ha estat establerta en base la metodologia emprada.

S'ha definit un període d'investigació, on s'ha entès que era una Botnet i adquirir els coneixements bàsics i seguidament s'han establert els tres grans períodes de desenvolupament.

El primer període contempla la creació d'un Honeypot establert en una xarxa monitoritzada i amb la configuració adient per ser infectat.

El segon període, contempla una via alternativa a la infecció per una Botnet externa, en cas de no aconseguir el primer objectiu de forma total i el Honeypot no aconsegueixi ser infectat. Durant aquest temps, s'establirà un Command & Control i es farà la infecció de forma voluntària a l'ordinador objectiu.

Finalment, el tercer i últim període es tracta d'una investigació, on a través de les dades extretes es detallaran una sèrie de mètodes vàlids per la detecció d'una infecció de Botnet, a nivell de xarxa.

1.7 Estat de l'art

L'estat de l'art en el camp de les Botnets està supeditat a les dos grans necessitats que tenen aquestes per viure.

La primer aspecte més important es troba al manteniment d'un accés persistent als Bots, actualitzant-los regularment amb nou codi maliciós, així com permetent l'accés al sistema quan es necessita realitzar un control remot. Han aparegut així nous atacs com són el "Siesta Campaign" o el "Antifulai Campaign", que permeten la comunicació amb el Command & Control a través de aplicacions al nuvol, serveis web, etc.

El segon aspecte més important es tracta de la seguretat del Command & Control a no ser identificat. Aquesta premissa fa que l'estructura de les Botnets estigui canviant d'una infraestructura jeràrquica, a la creació de xarxes de Bots P2P. Aquestes xarxes permeten al Command & Control estar amagat entra la resta de Bots, formant ell

part de la xarxa.

No obstant, la realització d'aquest projecte es centrarà en l'estudi de la comunicació d'una Botnet jeràrquica i en estat d'inactivitat.

2 DESCRIPCIÓ BÀSICA D'UNA BOTNET

2.1 Que és una Botnet?

Una Botnet és una xarxa d'ordinadors anomenats Bots, controlats de forma remota per un ent maliciós anomenat Command & Control (C&C). Des d'aquest punt de control es pot demanar a la xarxa d'ordinadors controlats que facin tasques malicioses d'entre les quals els atacs de denegació de servei distribuït és el més important i alhora més potent. Altres tasques son l'enviament d'Spam, la mineria de Bitcoins o fraus de clicks.

2.2 Parts de les que es composa una Botnet

Les Botnets estan composades per dos parts:

Bots: Es tracta de l'eina principal d'una Botnet. Representa cadascun de les maquines que n'hi ha sota el control del servidor central C&C i que han estat infectades per un arxiu maliciós sense el consentiment o autorització del propietari de la màquina. Cadascun d'aquests Bots estan en contacte directe amb el C&C i operen en base a les directrius que arriben des d'aquest.

Command & Control: També anomenat C&C. És l'agent maliciós que infecta el Bot d'una de les següents maneres:

- De forma directe: a través d'una vulnerabilitat a la màquina objectiu.
- Indirecte de forma activa: enviant correus que contenen l'arxiu maliciós i essent executat posteriorment pel propietari del Bot.
- Indirecte de forma passiva: deixant l'arxiu emmagatzemat en un servidor i permetent que la gent s'ho descarregui de forma voluntària i l'executi.

El C&C és l'encarregat d'administrat i gestionar els Bots de la xarxa i per tant, l'autor dels atacs donat que serà aquest qui enviï la directriu de llençar un atac de denegació de servei distribuït (DDoS), un enviament massiu d'Spam, etc.

2.3 Tipus de Botnets

Existeixen dos tipus de Botnets, les xarxes tradicionals i les P2P:

Les xarxes tradicionals estan basades en una jerarquia, on trobem el nivell superior amb el Command & Control i el nivell inferior on trobem la xarxa de Bots. Aquest tipus de Botnets estan formades per una sèrie de C&C que son els encarregats de enviar les directrius a la resta de bots amb una comunicació un a un o un a molts. Aquestes xarxes normalment utilitzen diversos C&C per no dependre únicament d'un, de forma que si un d'ells estigués inhabilitat, la Botnet no quedés sota el control de ningú. D'aquesta manera asseguren que a menys que tots els servidors de control caiguessin, sempre podran gestionar la Botnet i modificar-la de la forma que disposin.

Les xarxes P2P son un nou tipus de Botnets on tots els bots estan al mateix nivell, amb la diferència que el C&C és l'únic que pot enviar directrius. Aquests s'envien de forma peer to peer, és a dir, la directriu és enviada inicialment pel C&C, però després s'estén entre els Bots essent aquests els encarregats de passar-la fer que arribi a tota la xarxa.

Tot i que aquest últim tipus és cap tendeix a anar, aquest projecte està enfocat a l'anàlisi de xarxes de Botnets tradicionals.

2.4 Canals de comunicació

Existeixen diversos canals per la comunicació de les Botnets. Els principals són els següents:

1. A través del protocol HTTP fent una comunicació xifrada amb una aplicació Web de C&C. Aquest no es tracta d'un protocol eficient a l'hora de transmetre una directriu a un gran grup de Bots.

2. A través del protocol IRC. Es tracta del mètode de comunicació més empleat, donat que és un protocol molt efectiu a l'hora de comunicar-se amb un gran grup de màquines, com hem pogut veure durant la carrera.

3. A través del protocol P2P, el canal emprat per la comunicació en les Botnets d'aquest mateix tipus. Es tracta del mètode de comunicació que permet a la xarxa propagar la directriu de forma eficaç i fent que arribi a tots els Bots disponibles en aquell moment.

4. Finalment trobem nous tipus de comunicacions basats en l'emmascarament del missatge com pot ser a través de l'enviament de imatges amb informació continguda a la metadata, a través d'aplicacions com Tor o senzillament canviant l'estat a Linkedin.com posant la directriu i fent que els bots es fixin en aquest camp.

3 ESCENARIS

3.1 Avaluació dels possibles escenaris

Primerament, la via principal del projecte pretén establir un escenari on hi hagi una màquina monitoritzada i alhora vulnerable per poder ser infectada. La idea principal és crear una màquina virtual amb aquests paràmetres, però finalment, es vol realitzar un treball en un escenàri de desenvolupament i s'opta per l'establiment d'un Ho-

neypot en una màquina real.

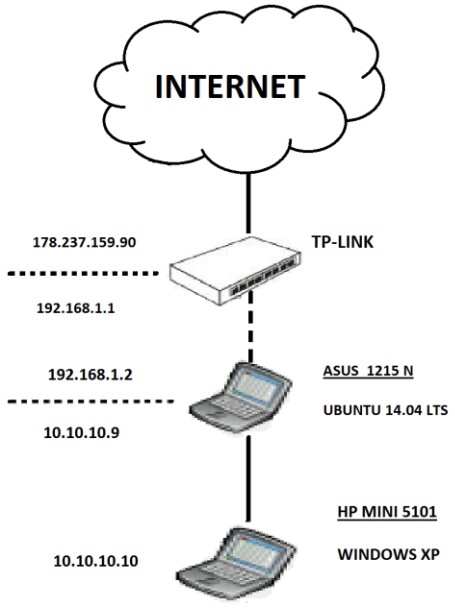
Seguidament es va haver d'escollir un sistema operatiu capaç de ser infectat. Per l'elecció d'aquest es va haver d'escollir entre el sistema operatiu windows XP sense cap Service Pack i amb el SP2. Finalment es va decantar pel que no incorporava cap tipus de seguretat afegida, el windows XP sense service pack ja que aquest donava un ventall de vulnerabilitats encara més gran respecte de l'altre, amb l'inconvenient de ser més difícil aconseguir-ho.

Un cop establert, es va haver de escollir com s'anava a monitoritzar i com fer les conexions des del router fins a la màquina monitora i d'aquesta fins al Honeypot.

La primera elecció va ser la instal·lació de Windows 7 al monitor i amb l'ajut d'una aplicació, fer de Wifi Acces Point. L'inconvenient llavors residia en que el sistema Windows instal·lat al Honeypot no es podia conectar atès que era necessària una clau WPA2 i la versió establerta només coneixia les claus WEP.

La següent opció llavors consistia en realitzar la instal·lació anterior però amb un sistema Linux com equip monitor. El resultat va ser semblant, la connexió per Wifi entre el monitor Linux i el Honeypot no era tabú.

Finalment, es va decidir a fer una connexió Wifi entre el router i l'equip Linux i la creació d'una LAN entre aquest i l'ordinador que seria infectat. Per tant l'escenari final va quedar de la següent forma:



3.2 Obrint la connexió a l'exterior

Un cop establert l'escenari, s'ha de permetre la infecció del Honeypot. Per fer aquesta infecció, primerament des del equip de monitoreig que farà a l'hora de bridge, s'utilitzarà Iptables per donar accés tant del Honeypot a Internet com d'Internet al Honeypot.

Un cop s'estableix comunicació cap a l'exterior, s'ha de permetre que un agent extern sigui capaç de infectar el Honeypot de forma voluntària. Per fer això, iniciaré el servidor web de Windows IIS.

Seguidament s'obren els ports del servei HTTP al router i es redirigeix tot el tràfic cap a l'equip de monitoreig. Es repeteix l'acció al monitor i es redirigeix tot el tràfic que entra pel port 80 cap al Honeypot. Un cop fet això, ja n'hi ha accés des de l'exterior fins al Honeypot, donant la possibilitat així que un agent extern pugui infectar el Honeypot.

Arribat aquest punt, s'instal·la Wireshark al equip de monitoreig i es realitzen connexions amb el Honeypot a pàgines infectades conegudes per fer pública la ip del Honeypot i aconseguir ser infectat.

Després de dos setmanes de monitorització del honeypot, el resultat ha estat negatiu i no s'ha estat capaç de ser infectat. Aquesta afirmació es pot fer atès que l'equip es troba en una xarxa a part, on s'ha monitoritzat tot el tràfic, essent aquest gairebé inexistent.

4 Alternativa (infectar jo mateix)

4.1 Alternativa a la infecció externa

Degut a que existia la possibilitat de no ser infectat per un agent extern, estava establerta la via alternativa de fer la infecció de forma voluntària. Després de dos setmanes i al començament del tercer període comença el *deadline* de l'anàlisi de dades i al no estar infectat, s'inicia el procés d'establiment d'un Command & Control i la posterior infecció voluntària.

Per fer això, s'ha decidit descarregar una Botnet pública, ZeuS versió 1.2.7.19. Aquesta aplicació es tracta d'una Botnet tradicional, a través de la qual s'instal·larà un servidor de Command and Control, es crearà l'arxiu maliciós i s'executarà aquest últim al Honeypot.

Tot i que hagués estat un projecte més profund essent infectat de forma involuntària, era un factor que no depenia de les meves accions i per tant, es seguirà el mateix mètode amb aquest, intentant simular un cas real d'infecció.

5 Instal·lació ZeuS

El procés d'instal·lació d'una Botnet s'inicia amb l'avaluació de quina serà l'aplicació a instal·lar. Finalment, es decideix l'utilització d'una versió de ZeuS Open Source 1.2.7.19 que conté el servidor a instal·lar amb el Command & Control i l'aplicació la qual a través de la configuració facilitada, crearà l'arxiu maliciós que a posteriori serà l'origen de la infecció.

Pel que fa a l'escenari, es canviarà la plataforma de Honeypot respecte l'escenari inicial. En aquest cas es

decanta pel sistema operatiu Windows XP SP2, ja que conté llibreries necessàries per poder executar l'arxiu maliciós.

Per fer la instal·lació de ZeuS es divideix en dos seccions, per una part es fa la instal·lació del Command & Control i per una altra, s'utilitza l'aplicació per la creació de l'arxiu maliciós i s'executa posteriorment al Honeypot.

5.1 Server side

La instal·lació del Command & Control s'estableix a la plataforma XAMPP del equip Linux, la màquina dedicada a fer la monitorització.

Per començar, es descarrega l'aplicació de XAMPP i s'instal·la el servidor Web com a requisit previ a la instal·lació del C&C. Seguidament, s'estableixen les taules a la base de dades que utilitzarà ZeuS i a continuació s'executa des del servidor Web l'arxiu d'instal·lació. Un cop acabat la instal·lació, el servidor de control de la xarxa de Bots quedarà establert. Serà des d'aquí des d'on seran gestionats els Bots, on es rebran els paquets amb informació dels Bots o des d'on s'enviaran les accions a realitzar tals com els anteriorment anomenats atacs DDoS.

5.2 Client side

La infecció a la part del client, es realitzarà primerament amb l'execució de l'aplicació ZeuS Builder i introduint les dades de configuració del Command & Control i els temps de demanda de l'arxiu de configuració i informe de dades. Les dades escollides son les següents:

IP	192.168.1.2
Temps arxiu de configuració	60
Temps d'informe	20

Seguidament, es crearà l'arxiu maliciós i serà executat al Honeypot. Des d'aquest moment s'iniciarà la comunicació i l'enviament dels missatges detallats a l'apartat de comunicació.

5.3 Monitorització

Un cop s'ha establert l'escenari real d'infecció es procedirà al monitoreig per veure com es realitza la comunicació a baix nivell i poder extreure dades per , a posteriori, fer un análisi i definir mètodes de detecció.

Per fer aquesta monitorització s'utilitzarà l'sniffer de xarxa instal·lat amb anterioritat. S'executarà Wireshark com superusuari i s'esnifarà el tràfic de la xarxa local creada entre l'equip monitor, ara també C&C, i el Honeypot.

6 INVESTIGACIÓ

6.1 Introducció a la investigació

Finalment s'ha establert l'escenari que permetrà fer un estudi de com és la comunicació entre un Bot, i el seu C&C. És ara quan s'executa l'sniffer de xarxa Wireshark i es pot veure quin esquema de comunicació és utilitzat entre la màquina infectada i el seu centre de control.

Per començar es mostraran com són els missatges que s'envien a través de la xarxa i seguidament s'exposaran els resultats d'un mostreig realitzat durant un període de temps d'aquesta comunicació. Per finalitzar, utilitzant l'esquema de comunicació de Zeus i analitzant les dades extretes s'establiran una sèrie de conclusions.

6.2 Comunicació

La comunicació que s'ha estudiat ha estat la d'un Bot en estat inactiu essent monitoritzat pel Command & Control. S'ha escollit analitzar aquesta comunicació en comptes de l'enviament d'accions per part del C&C donat que més del 99% del temps que un Bot està encès, únicament es troba en estat de repòs i monitoritzat per part del servidor de control.

La comunicació es composa de quatre missatges, dos per la configuració del Bot i els altre dos per l'enviament d'informació d'aquest:

Missatges de configuració del Bot:

1. Missatge que envia el Bot al Command & Control demanant l'últim arxiu de configuració. Aquest conté informació com l'adreça o conjunt d'adreces on pot allotjar-se el Command & Control, temps de resposta entre paquets, etc.. Els canvis de configuració en aquest aconsegueixen fer encara més difícil si cap, tant la detecció de la infecció, com la localització del Command & Control.

2. Missatge de resposta per part del C&C enviant l'arxiu demanat de forma xifrada. El Command & Control envia de forma xifrada l'últim arxiu de configuració actualitzat amb les dades necessàries.

Missatges d'enviament d'informació del Bot:

3. Per l'enviament d'informació per part del Bot, s'utilitza un missatge de tipus POST amb la informació sensible que s'extreu de la màquina infectada. Aquesta pot ser informació emmagatzemada en cookies, informació que està sent enviada, screenshots en temps real, etc.. tots ells xifrats.

4. Missatge d'acceptació d'arribada de la informació. El C&C indica al Bot que la informació ha estat rebuda correctament.

Un cop detallada la comunicació d'un client ZeuS amb el C&C, es mostrarà com és la successió i intercanvi de missatges:

Comunicació per la configuració del Bot:

1. Primerament, el client Zeus envia un missatge GET al C&C per a que aquest li enviï l'últim arxiu de configu-

ració actualitzat.

2. A continuació, C&C li envia un missatge HTTP/1.1 200 OK (application/octet-stream) amb l'arxiu de configuració xifrat.

A partir d'aquest moment, aquest intercanvi es realitzarà cada període fixat per el servidor de control i subjecte a canvis.

Comunicació per l'enviament d'informació del Bot:

1. El Bot envia un missatge POST al C&C amb la informació sensible que s'extrau de la màquina.

2. Finalment per concloure, el C&C respon amb un missatge HTTP/1.1 200 OK (text/html) conforme la informació ha arribat correctament.

A partir d'aquest moment, aquesta successió es repetirà de forma immediata fins que l'arxiu amb les dades s'envia sencer. En cas d'estar totalment inactiu i no crear noves dades s'enviarà un únic paquet amb el fitxer buit. El període entre una ràfega i una altra estarà detallada a la configuració definida pel Command & Control.

6.3 Dades del mostreig

Un cop s'ha mostrat com és la comunicació, s'ha realitzat una extracció de dades a partir de la mostra de Wireshark. Les dades mostren quins són les mitjanes de temps que n'hi ha entre missatges i la mitjana de tamany de cadascun.

Cal detallar, que pel camp "Tamany de missatges POST" s'han trobat tres tipus: Petits, Mitjans i Grans. Això és degut a que s'han trobat aquests tres grups ben diferenciats i era important remarcar-ho.

Temps mig entre el missatge nº1 i missatge nº2	0.88 segons
Temps entre missatge nº 3 i missatge nº 4	0,128 segons
Temps predeterminat entre dos missatges GET	60 minuts
Temps predeterminat entre dos missatges POST	20 minuts
Temps entre dos missatges GET	3600.065 segons
Temps entre dos missatges POST enviant el mateix informe	0.167 segons
Temps entre dos missatges POST	1200.814 segons

Mitjana Temps

Tamany de missatge GET	206 bytes
Tamany de missatge de resposta al missatge GET	412 bytes
Tamany de missatges POST	*Petits* : 195 bytes *Mitjans:* 576 bytes *Majors:* 1370 bytes
Tamany de missatge de resposta al missatge POST	Entre 381 i 382 bytes.

Mitjana Tamany

6.4 Conclusions del mostreig

Per finalitzar aquesta segona divisió del document, es detallaran les conclusions finals que s'han extret a través de l'anàlisi dels paquets del mostreig i de les dades que s'han mostrat a les dos taules anteriors:

1- El tamany dels missatges GET és sempre de 206 bytes a la meva mostra en el 100% dels casos, pel que es pot pràcticament assegurar que encara que no sigui d'aquesta mateixa mesura (206 bytes), serà sempre d'un tamany igual, per exemple essent tots de 220 bytes.

2- El tamany de les respostes al missatge GET és sempre de 412 bytes a la meva mostra en el 100% dels casos, pel que es pot pràcticament assegurar que encara que no sigui d'aquesta mesura (412 bytes), seran tots sempre d'un mateix tamany.

3- Un 80% dels missatges POST analitzats durant el mostreig realitzat mentre s'utilitza la màquina infectada i per tant es creen noves dades, tenen un tamany que oscil·la entre els 510 i 640 bytes, és a dir, pertanyen al grup mitjà.

4- El tamany dels missatges POST, quan no existeixen noves dades a enviar, és sempre del mateix tamany el 100% de vegades, 509 bytes a el mostra analitzada. D'aquesta forma, es pot gairebé assegurar que sempre serà d'un tamany estàtic.

5- El tamany de les respostes als missatges POST, oscil·la sempre entre els 381 i 382 bytes (100% de vegades). D'aquesta forma es pot assegurar que sempre serà d'un tamany gairebé estàtic, amb una variació d'un byte.

6- Els temps estipulats a l'arxiu de configuració, es segueixen estrictament, amb una variació menor al 0,07%.

7- Durant la successió de missatges POST per enviar un mateix informe, el temps que transcorre entre dos missatges consecutius varia entre els 0.130 i els 0.190 segons.

8- El temps entre el missatge POST i la seva resposta oscil·la entre els 0.124 i els 0.135 segons pràcticament sempre, tenint en compte que l'experiment és en una LAN i no viatja a través d'Internet.

9- El canal pel que es comunica amb el C&C és HTTP, per tant, tots els missatges utilitzen únicament aquest protocol.

10- En cas que el missatge GET no rebi un nou arxiu de configuració, tant aquests missatges com els POST s'enviaran sempre a la mateixa direcció, en el cas de la mostra les direccions han estat /xampp/1/cfg.bin pel missatge GET i /xampp/1/gate.php pel missatge POST.

11- Els missatges de resposta al POST son tipus text/html, però no contenen realment dades en aquest format, sinó possiblement una resposta xifrada de confirmació.

12- Tant pels missatges POST com pels missatges GET, el camp User-Agent del protocol HTTP és el mateix, essent aquest: Mozilla/4.0 (Compatible; MSIE 6.0; Windows NT 5.1; SV1)\r\n.

Aquestes són les conclusions més importants i característiques de ZeuS que he trobat i per tant, les que prendran més rellevància a l'hora de identificar mètodes capaços de detectar infeccions.

7 MÈTODES

Amb les dades i conclusions a la mà, s'explicaran a continuació una sèrie de mètodes de com es podria, en temps real i a nivell de xarxa, detectar la infecció de ZeuS sense fer cap anàlisis d'arxius de la màquina en qüestió.

7.1 Mètode 1

Aquest primer mètode està basat en la detecció d'una connexió realitzada a una adreça coneguda.

Aquest mètode es tracta de la forma més precisa de detectar la infecció, però també la més estàtica: El mètode consisteix en esnifar tots els paquets HTTP i veure l'origen i destinació. Si un d'aquests camps conté una adreça coneguda, es pot assegurar que el sistema està infectat. Pel coneixement d'aquestes adreces malicioses, s'utilitzarà un servei web molt conegut, que ens proveirà d'una llista actualitzada de adreces conegudes, es tracta de https://zeustracker.abuse.ch/.

Amb un mètode semblant però d'una forma menys precisa, es pot comprovar si la connexió dels missatges HTTP GET o POST es fa a una extensió coneguda pels arxius de configuració i d'enviament d'informes. En el meu cas les extensions eren /xampp/1/cfg.bin pel GET i /xampp/1/gate.php pel POST. Aquest segon sub-mètode pot donar falsos positius, però juntament amb un anàlisi estructural i/o utilitzant el segon mètode es podria arribar a assegurar la infecció amb un número de mostres suficient.

7.2 Mètode 2

El segon mètode està basat en la diferència que n'hi ha en el missatge de resposta al POST entre el tipus de missatge que posa que és i el seu contingut. Aquesta resposta posa que és de tipus text/html tot i que realment no ho és, sinó que es tracta d'una resposta de confirmació xifrada. El següent mètode consta de dues parts.

Primerament, s'han de rastrejar paquets del tipus HTTP 1.1 200 OK, text/html. El primer filtre seria la cerca dels missatges HTTP que són l'únic tipus que empra Zeus i el segon filtre seria buscar de tipus 1.1 200 OK (text/html) que es tracta dels paquets de resposta per part del C&C als missatges POST d'informes que fa el BOT.

Seguidament, com s'ha comentat a l'anàlisi anterior, pels missatges de resposta HTTP 1.1 200 OK (text/html) el camp de dades no és un camp de tipus text html pla sinó una resposta xifrada. Per tant, s'ha de comprovar que el camp del cos del missatge no sigui text html pla identificable. Si s'identifica un missatge d'aquest tipus i el cos de les dades està xifrat, es pot afirmar que existeix la possibilitat de estar infectat per ZeuS. El problema d'aquest mètode recau en que no es pot assegurar al 100% que tots els missatges d'aquest tipus siguin de ZeuS, ja que poden haver-hi altres aplicacions que també ho utilitzin de manera voluntària o involuntària i per tant poden haver-hi casos de falsos positius.

De la mateixa manera que amb el sub-mètode anterior, es pot assegurar la infecció aplicant adicionalment el tercer mètode i fent un anàlisi estructural per afirmar de forma precisa si s'està realment infectat.

7.3 Mètode 3

El següent i últim mètode està basat en la recerca de l'estructura de comunicació de ZeuS.

Aquest mètode consisteix en fer un anàlisi dels missatges que s'envien per poder identificar el patró estructural que segueix ZeuS i així afirmar que s'està infectat. La precisió d'assegurar la infecció augmenta a mesura que la mostra creix. L'estructura d'aquest mètode és la següent:

Primerament s'agafaran únicament els missatges HTTP POST i es compararan amb els de la mostra. S'ha escollit fer la cerca de missatges POST atès que l'enviament d'una ràfega d'aquests missatges per l'expedició d'un informe, crearà diversos paquets del mateix tipus. Quan es troben dos missatges amb els camps del protocol HTTP iguals en un mateix interval de

temps, a excepció del camp de longitud i informació -data amb l'informe xifrat- es buscarà a l'interval que va des del primer post fins al segon, els missatges de HTTP/1.1 200 OK (text/html). Finalment, s'aplicarà el segon mètode definit per comprovar si realment és un arxiu de text. A partir d'aquí, en cas de detectar que no sigui un arxiu de text, es pot continuar la cerca de missatges amb el camp de protocol idèntics als POST trobats i aconseguir més mostres per poder declarar, cada cop amb més probabilitat i precisió, que realment s'està infectat per una Botnet.

D'una forma més acurada, s'aplicaran adicionalment solucions com la cerca de missatges POST i GET amb el camp User-Agent com està definit al punt dotze de les conclusions, o la cerca d'estructures de missatges GET i un missatge immediat HTTP/1.1 200 OK (application/octet-stream).

Finalment, altres factors a tenir en compte poden ser la cerca només de missatges POST i la seva resposta amb longitud basada en l'anàlisi, la realització d'un anàlisis dels missatges GET enviats per comprovar si són del mateix tamany, etc...

8 Dificultats

Les dificultats que es poden trobar a l'hora d'aplicar els mètodes definits amb anterioritat son els següents:

1- L'adreça del Command & Control es tracta d'una URL no coneguda. D'aquesta manera seria gairebé impossible detectar la infecció.

La forma de resoldre-ho implicaria l'anàlisi de l'extensió de la URL, donat que molts d'aquests utilitzen adreces acabades amb gate.php o secure.php.

2- La dificultat que té la cerca de pautes de comunicació recau en que es tracta de buscar una estructura dinàmica donat que l'arxiu de configuració pot anar canviant cada cop i així els seus temps entre missatges POST i missatges GET o les seves adreces d'enviament.

La forma de resoldre aquesta dificultat es troba en que normalment l'interval en que es realitza l'acció d'enviament de l'arxiu de configuració tindrà un temps major al dels informes. D'aquesta manera pots aconseguir com a mínim dos missatges POST iguales als que es pot aplicar el mètode tres.

En cas que l'arxiu de configuració tingui un temps definit menor al d'enviament d'informes, es pot adaptar el mètode tres canviant els missatges POST pels missatges GET i fent una cerca dels missatges de resposta a aquest, donat que sempre són d'un mateix tamany o tenen una variació molt petita. Per una altra part, també es pot seguir amb el mètode anterior, amb l'inconvenient que serà necessari un temps major per l'anàlisi de la mostra en cas de que les adreces canviïn al llarg del temps.

9 Comprobació de la metodologia amb Snort

9.1 Introducció

Per finalitzar el treball, s'ha corroborat un dels mètodes establerts i s'ha realitzat la comprovació mitjançant l'IDS Snort.

Aquesta comprovació ha donat un resultat satisfactori, assumint així que l'objectiu del projecte basat en la identificació d'un mètode capaç de detectar una infecció a temps real i a nivell de xarxa ha estat totalment assolit.

El mètode escollit ha estat el primer, donat que per una part és el més viable a l'hora d'implementar-ho, dins del marc d'aquest projecte de final de grau i seguidament per que es tracta de la metodologia que hem donarà un nivell de precisió per la detecció major.

9.2 Comprobació

El IDS Snort ha estat executat amb regles bàsiques basades en el primer mètode de cerca de pàgines conegudes i el resultat ha estat positiu.

Primerament, s'ha instal·lat l'IDS Snort, encarregat de aplicar les regles als paquets filtrats, al sistema operatiu Ubuntu. Seguidament s'ha creat l'arxiu de configuració "snort.conf" i s'ha inclòs tots els arxius necessaris per el correcte filtratge.

Un cop omplert, s'ha descarregat la llista de direccions conegudes des de la pàgina detallada al segon mètode: https://zeustracker.abuse.ch/blocklist.php. Seguidament, s'ha inclòs l'arxiu descarregat a l'arxiu de configuració d'Snort.

Tot i que en un cas real, la configuració d'Snort hagués acabat aquí, en el meu cas, s'ha creat una nova regla d'alerta, posant el host 192.168.1.2, on està ubicat el servidor de ZeuS i l'extensió ha estat /gate.php.

Un cop Snort es troba correctament configurat, s'ha executat el servidor ZeuS, i seguidament s'ha fet el mateix amb Snort i finalment s'ha encès el Bot. Un cop aquest ha començat a enviar paquets, Snort ha enregistrat al logs del sistema el missatge d'alerta d'infecció.

10 Conclusió

Com s'ha pogut veure al llarg de l'informe, les Botnets son un malware que s'està estenent cada cop més i estan prenent cada cop més força els seus atacs DDoS, no obstant els seus mètodes de detecció a temps real són encara escassos i la seva capacitat de detecció de Botnets desconegudes, molt reduïda.

Els objectius plantejats a l'inici del projecte han estat complerts amb la incidència de no haver estat infectat per una Botnet real el Honeypot inicialment establert. Pel que fa a l'escenari de desenvolupament, s'ha configurat per

poder albergar un Honeypot i monitoritzar-ho, tot i que finalment no s'ha arribat a infectar. El segon objectiu ha estat aconseguit atès que s'ha establert un Command & Control de la Botnet ZeuS i s'ha infectat voluntàriament el Honeypot. Finalment A través de l'anàlisi de la comunicació, de les dades extretes i de les observacions s'han establert tres mètodes vàlids per la detecció d'una infecció d'una Botnet, essent un d'ells demostrat i corroborat. Aquests mètodes han estat plantejats amb l'objectiu de poder fer una detecció no només a nivell de xarxa sinó també amb la capacitat de trobar nous tipus de Botnets basats en la comunicació de ZeuS.

El primer dels mètodes ha estat no només plantejat de forma teòrica sinó implementat i provat en un entorn de desenvolupament amb el servei d'Snort. Els dos mètodes restants han estat definits de forma teòrica i tot i que a priori han de ser vàlids, no han estat corroborats per falta de la seva implementació. S'han detallat quines poden ser les dificultats trobades i com es poden superar per donar un caire més real i tenir una visió de com pot ser el comportament d'un agent maliciós.

Un cop finalitzat el treball puc dir que tant la planificació com la consecució dels objectius plantejats a l'inici han esta totalment un èxit, atès que aquests han estat assolits en el període que s'esperava.

Un cop finalitzat el treball, una de les principals línies d'investigació, seria la implementació dels mètodes dos i tres i la seva corroboració en un entorn real. Ha estat impossible dins del marc d'aquest treball fer la comprovació de qualsevol d'aquests dos mètodes donat que el temps a la planificació ha estat complert. Seguidament, un altre projecte a curt termini seria l'estudi de com es llencen els atacs DDoS i com es comporta el Bot. Aquesta investigació i el seu resultat, podrien completar els mètodes definits en aquest projecte i aportar un gran valor.

Per una altra part, existeixen altres línies d'investigació com seria la cerca de vulnerabilitats a l'algoritme de xifrat de la comunicació, donat que es tracta d'un algoritme RC4, i és conegut com a dèbil per naturalesa o l'estudi de com s'envien aquests arxius maliciosos, essent aquest un camp de gran amplitud.

11 AGRAÏMENTS

Vull donar les gràcies primerament al meu tutor Guillermo Navarro per totes les facilitats que m'ha donat al llarg del treball, a la meva família i especialment a la meva parella Verónica Rodríguez Romero pel seu suport durant la realització del projecte.

12 BIBLIOGRAFIA

[1] *The honeypot project: "Know your Enemy: Tracking Botnets"* [en línia] Paul Bächer, Thorsten Holz, Markus Kötter, Georg Wicherski. Disponible en http://www.honeynet.org/papers/bots

[2] *Botnet tracking: Tools, Techniques, and Lessons learned* [en línia] Dr. Jose Nazario. *Disponible en:* https://www.blackhat.com/presentations/bh-dc-07/Nazario/Presentation/bh-dc-07-Nazario.pdf

[3] *Botnet tracking: Tools, Techniques, and Lessons learned* [en línia] Dr. Jose Nazario. *Disponible en:* https://www.blackhat.com/presentations/bh-dc-07/Nazario/Presentation/bh-dc-07-Nazario.pdf

[4] *Catching Malware: Detecting, Tracking and Mitigating Botnets* [en línea]. Holz & Wicherski, 2006. Disponible en: http://www.blackhat.com/presentations/bh-jp-06/BH-JP-06-Wicherski-Holz.pdf

[5] *Survival Time on the Internet* [en línea]. Lorna Hutcheson, 2008-07-14. Disponible en: https://isc.sans.edu/diary/Survival+Time+on+the+Internet/4721

[6] *Remote Administration Tool Zeus BotNet (RAT)* [en línia]. Vishnu Valentino. Disponible a: http://www.hacking-tutorial.com/hacking-tutorial/remote-administration-tool-zeus-botnet-rat/#sthash.pgTULCcP.dpbs

[7] *VRT Labs – Zeus Trojan Analysis* [en línia]. Sourcefire VRT Labs. Disponible a: https://labs.snort.org/papers/zeus.html

[8] *Realtime C&C Zeus Packet Detection Based on RC4 Decryption of Packet Length Field* [en línia]. ChulWoo Park, HyoSung Park, KiChang Kim, 2014. Disponible a: http://onlinepresent.org/proceedings/vol64_2014/14.pdf

[9] *Botnet Traffic Detection* [en línia]. Chen Lu, 9 de Novembre de 2011. Disponible a: http://www.kaspersky.com/images/Chen%20Lu-10-108171.pdf

[10] *What is Zeus?* [en línia]. James Wyke, Maig 2011. Disponible a: http://www.sophos.com/medialibrary/PDFs/technical%20papers/Sophos%20what%20is%20zeus%20tp.pdf

[11] *Sistemas de detección de intrusos y Snort (II). Creación de reglas (I)* [en línia]. Seguridad y Redes, 22 de Gener de 2008. Disponible a: https://seguridadyredes.wordpress.com/2008/01/22/sistemas-de-deteccion-de-intrusos-y-snort-ii-creacion-de-reglas-i/