POLITECNICO DI MILANO
Facoltà di Ingegneria dell'Informazione
Corso di Laurea Specialistica in Ingegneria Informatica

UNA METODOLOGIA DI ANALISI DI MALWARE UTILIZZATI PER LA CREAZIONE DI BOT

Relatore: Prof. Stefano ZANERO
Correlatore: Ing. Guido SALVANESCHI

Tesi di Laurea di:
Mauro PESSINA
Matr. n. 720409

Anno Accademico 2009–2010

Alla mia famiglia

Sommario

Contestualmente all'enorme crescita del mercato IT a cui si è potuto assistere negli ultimi anni, si è verificato un rilevante incremento del numero delle minacce presenti nella rete. L'evoluzione dei malware ha seguito una crescita analoga, rendendosi una delle principale sfide per gli esperti in sicurezza del settore.

In questo contesto si è reso necessario concentrare gli sforzi sulla comprensione del fenomeno, in modo da poter presentare una risposta efficace alla minaccia che esso rappresenta. Si è così sviluppata la disciplina dell'analisi del malware, mirata alla comprensione della struttura e del funzionamento dei codici malevoli.

Tale disciplina ha portato alla creazione di infrastrutture automatizzate per la comprensione del malware. Il progetto WOMBAT [1] ha l'obiettivo di sviluppare nuove tecnologie che possano facilitare lo studio delle minacce alla rete Internet ed ai servizi che essa fornisce. In questo ambito sono state sviluppate delle API che costituiscono un'ottima piattaforma per l'analisi dei codici malevoli e dei loro meccanismi di diffusione e controllo [2].

La tesi verte sullo sviluppo di una metodologia basata su tale strumento per l'osservazione del comportamento di dataset di malware. Nello specifico ci si propone di analizzare l'infrastruttura di controllo di botnet a partire dall'esame dei bot con cui sono state costruite. Il lavoro presenta lo sviluppo di una suite di tool destinati a differenti livelli di indagine del fenomeno.

Inizialmente la tecnica è sufficientemente generica da prestarsi ad analisi di campioni di malware qualsiasi. Procedendo nel lavoro si presenta invece un approccio orientato specificatamente al problema delle botnet, andando ad analizzarne infrastrutture e sistemi di controllo. Si cerca di individuare schemi comuni all'interno dei dataset esaminati, verificando l'eventuale esistenza di connessioni tra famiglie di malware apparentemente differenti.

La metodologia sviluppata è quindi applicata un dataset di bot IRC. Vengono individuate relazioni significative tra botnet apparentemente differenti e si scopre come in alcuni casi dietro a migliaia di infezioni possa celarsi lo stesso individuo od organizzazione.

Indice

Elenco delle figure

Elenco delle abbreviazioni

IRC	Internet Relay Chat
API	Application Programming Interface
AV	AntiVirus
DoS	Denial of Service
DDoS	Distribuited Denial of Service
C&C	Command and Control
TCP	Transmission Control Protocol
UDP	User Datagram Protocol
ICMP	Internet Control Message Protocol
IP	Internet Protocol
HTTP	Hypertext Transfer Protocol
AS	Autonomous System
DGA	Domain Generation Algorithm
CSV	Comma Separated Value

Capitolo 1

Introduzione

La sempre più profonda diffusione dell'informatica nella vita di tutti i giorni ha aperto la strada ad un grande numero di attività che fino a pochi anni fa sembravano impensabili. Ad oggi chiunque entra in contatto più volte nell'arco della giornata con dispositivi tecnologici sempre più complessi, ma spesso l'utilizzo di questi strumenti non è accompagnato da una adeguata sensibilità al tema della sicurezza informatica.

L'argomento è più che mai attuale, perchè se da una parte la crescente complessità dei sistemi informatici apre un grande ventaglio di opportunità per cittadini ed aziende, dall'altro tali sistemi sono sempre più difficili da tutelare nei confronti degli attacchi provenienti da entità interessate all'alterazione o al furto dei dati da essi conservati.

In questo contesto si inserisce il fenomeno del malware. Si definisce malware un qualsiasi software creato con il solo scopo di causare danni più o meno gravi al computer su cui viene eseguito [3]. Per combattere il fenomeno è necessario prima di tutto acquisirne una maggiore conoscenza, capendone funzionamento e meccanismi di diffusione.

Ad oggi la sfida è l'identificazione non solo di chi sia l'autore del codice malevolo, ma anche chi abbia pagato per tale sviluppo. In quest'ottica occorre ricordare che il malware è solo uno strumento, la vera minacca è l'individuo che lo controlla.

Una direzione di ricerca importante in quest'ottica è lo sviluppo di metodi che consentano una risposta tempestiva alle infezione, anche a quelle mai viste in precedenza. Individuare rapidamente le informazioni necessarie alla comprensione del comportamento del malware senza sovraccaricare l'utente di dati inutili è un'operazione estremamente difficile; oltre a ciò è necessario affrontare le contromisure che gli attaccanti pongono in essere per negare all'organizzazione attaccata di sfruttare i comuni tool di analisi.

1.1 Sviluppo della metodologia di analisi

1.1.1 Motivazioni

L'analisi del malware è l'isolamento di un malware ed il suo studio. Obiettivo dello studio è l'individuazione di risposte a domande sul codice malevolo in esame. Tali domande possono focalizzarsi sugli obiettivi del malware, sulle modalità dell'infezione, la sua durata ed i metodi per la sua rimozione; è necessario essere in grado di valutare le capacità di chi sta eseguendo l'attacco, individuare le giuste contromisure ed indagare i meccanismi di diffusione per impedirne la propagazione su altre macchine. Occorre inoltre essere in grado di valutare il danno causato, in modo da potervi porre rimedio per quanto possibile, e capire in che modo si possa evitare che il fenomeno si ripeta in futuro [4].

1.1.2 Metodologia

Il lavoro qui effettuato è orientato allo sviluppo di una metodologia fortemente automatizzata per lo studio di dataset di malware. Nello specifico il metodo utilizzato si presta in particolar modo all'analisi di malware orientato alla creazione di botnet, tuttavia la tecnica utilizzata è facilmente applicabile ad un qualsiasi tipo di codice malevolo che preveda la generazione di traffico di rete.

L'indagine viene condotta utilizzando lo strumento delle WAPI, cioè un insieme di API sviluppate nell'ambito del progetto Wombat orientate allo studio di codici malevoli. Si propone una metodologia che possa facilitare lo studio di un dataset di malware, partendo dal riconoscimento delle famiglie presenti, utilizzando la nomenclatura di un antivirus, fino ad arrivare allo studio dell'infrastruttura di controllo delle botnet create utilizzando i campioni in esame. Si approfondisce quindi lo studio di questa infrastruttura indagando quei domini e server contattati dai file in esame, verificando inoltre la presenza di eventuali individui od organizzazioni comuni dietro alla gestione di botnet apparentemente differenti.

Il primo passo consiste nell'individuazione delle famiglie di malware presenti nel dataset di nostro interesse. Per far ciò è utilizzata la nomenclatura dell'antivirus con il maggior rateo di riconoscimento per l'analisi in oggetto, in modo da garantire una copertura quanto più estesa possibile dei file da esaminare.

Si procedee poi alla determinazione del traffico di rete generato, alla ricerca di quella che potrebbe costituire l'infrastruttura di controllo dei malware studiati. I domini ed i server così individuati vengono indagati alla ricerca di informazioni utili per l'analisi.

A seguire si cerca di determinare un subset dei domini fin qui rilevati per approfondire l'analisi. L'applicazione del procedimento seguito in questa fase alla totalità dei domini individuati introdurrebbe infatti una enorme quantità

di rumore che sicuramente non gioverebbe alla chiarezza dei risultati del lavoro. Si è quindi optato per la selezione di quei domini caratterizzati da un elevato numero di richieste da parte dei malware contenuti nel dataset. Di tali domini il tool sviluppato ricerca i dati necessari a dimostrare evenutali relazioni reciproche od interazioni con una specifica famiglia di malware.

1.2 Case study

Al termine dello sviluppo della metodologia di analisi si procederà alla validazione della stessa applicandola ad caso reale.

Nello specifico si esaminerà un dataset di malware che utilizzano il protocollo IRC come mezzo di comunicazione tra la macchine infettate e gli attaccanti. L'applicazione dell'analisi su tale dataset si presenta interessante in quanto IRC risulta essere il protocollo di comunicazione più utilizzato per il controllo delle botnet [5].

Durante esecuzione del caso di studio si osserverà la notevole predominanza di alcune famiglie di malware, si eseguirà un'analisi live di quei domini e server che costituiscono l'infrastruttura di controllo dei bot e si indagheranno più in dettaglio alcuni server caratterizzati da un'elevato numero di contatti da parte dei sample nel dataset. Al termine di questo processo si disporrà di una notevole quantità di informazioni circa i server utilizzati per il controllo delle botnet. Sarà possibile osservare la distribuzione sullo spazio IP di tali server, così come si potrà visualizzare la distribuzione geografica degli IP individuati.

L'analisi procederà utilizzando gli strumenti sviluppati durante la preparazione della metodologia di indagine, tuttavia in alcuni casi sarà necessario l'intervento umano approfondire l'indagine di alcune caratteristiche peculiari del dataset. Si noterà che in alcuni casi è possibile dimostrare connessioni tra infra-

strutture di controllo apparentemente differenti, mentre in altri casi si potranno rilevare solamente degli indizi in tal senso.

1.3 Struttura della tesi

Nel capitolo 2 si presenterà il background del lavoro eseguito. Innanzitutto si affronterà l'argomento dell'analisi del malware, verrà introdotto il problema e si proporranno le tecniche più comunemente utilizzate per affrontarlo. Più specificatamente verranno descritti i 2 approcci, statico e dinamico, che è possibile seguire per condurre l'analisi su dei campioni di malware. A seguire si affronterà l'argomento delle botnet, presentandone le attività ed indicando quali siano i bersagli più appetibili per gli attaccanti. Ci si addentrerà quindi nella descrizione dei malware basati su IRC, analizzando protocollo, elementi distintivi di un attacco da parte di IRC bot, meccanismi di infezione e controllo e caratteristiche delle botnet di questo tipo. Verranno poi introdotto le WAPI, cioè le interfacce utilizzate per l'implementazione del lavoro di analisi presentato a seguire.

Il capitolo 3 costituisce il nucleo della tesi, affrontando effettivamente il problema dello sviluppo della metodologia di analisi per dataset di malware. Il lavoro qui presentato verrà suddiviso in 3 fasi successive, che consisteranno rispettivamente nell'identificazione dei campioni contenuti nel dataset, nell'analisi live sull'infrastuttura di controllo così rilevata e quindi nell'approfondimento dell'indagine per quei domini che si riveleranno più significativi.

Nel capitolo 4 il procedimento sviluppato troverà applicazione per l'analisi di un dataset di malware IRC- based. Verranno quindi sfruttati gli strumenti sviluppati in precedenza per verificarne l'efficacia in un'applicazione reale.

Il capitolo 5 presenterà poi i risultati di questo lavoro.

Capitolo 2

Background

2.1 Malware analysis

Malware è un termine generico che indica un tipo di software non desiderato. Tale software costituisce una minaccia molto seria per qualsiasi utente che interagisca con un computer. Secondo alcune stime, le perdite monetarie derivanti dall'infezione da parte di malware nel solo anno 2005 hanno raggiunto i 14 miliardi di dollari [6]. Il fenomeno è tuttavia in costante crescita, in quanto il business legato a tale tipo di software è estremamente remunerativo.

Gli autori di tali applicazioni hanno diverse fonti di rendita, che spaziano dalla vendita del programma da essi sviluppato, all'effettivo utilizzo di tali software per infettare un vasto numero di macchine da utilizzare poi come piattaforma per il lancio di attacchi informatici o per l'invio di messaggi di spam [7].

Una chiara indicazione della gravità del problema proviene anche dalla crescente attenzione della stampa generalista: il caso più recente riguarda il worm Stuxnet, salito alle cronache nonostante la maggioranza dei normali utenti di computer non fosse interessata dall'infezione [8].

Esistono molte forme di protezione dal malware [9]. Innanzitutto è conveniente utilizzare antivirus (AV) e firewall, accertandosi che il database di protezioni dell'antivirus sia aggiornato di frequente. Scansioni ad intervalli regolari sono consigliate per monitorare quei file che potrebbero aver superato i controlli real time dell'antivirus. Sistema operativo ed applicazioni installate dovrebbere essere sempre mantenute alla versione più recente, in modo da ridurre la possibilità che l'infezione possa avvenire sfruttando falle conosciute di vecchie versione del software utilizzato. E' inoltre preferibile acquisire una certa dimestichezza con le impostazioni di sicurezza della applicazioni in grado di accedere alla rete, al fine di individuare il giusto compromesso tra semplicità d'uso e sicurezza. L'utilizzo di password sicure è fondamentale: il sistema di controllo degli account utente presente in tutti i moderni sistemi operativi costituisce un meccanismo di protezione molto importante, la cui sicurezza dipende però da quella della password dell'account con diritti di amministratore. A monte di tutti questi meccanismi di protezione dovrebbe in ogni caso esserci l'utente, la cui sensibilità sul tema della sicurezza informatica dovrebbe costituire il primo meccanismo di protezione dal malware .

Vale ora la pena di approfondire il funzionamento degli antivirus. Questi software fanno generalmente affidamento su un database di signature in grado di riconoscere le istanze di un malware. Qualora i produttori di soluzioni AV dovessero individuare un nuovo malware, il sistema di protezione prevede in genere un aggiornamento delle informazioni contenute nel database e la conseguente trasmissione di esso alle singole macchine sui cui l'antivirus è installato. Ovviamente in un meccanismo di funzionamento come quello appena presentato il tempo è una variabile estremamente importante: prima il produttore dell'AV riesce ad analizzare il sample di un nuovo malware, studiandone comportamento ed effetti sul sistema, prima le macchine degli utenti posso essere messe in sicu-

rezza. L'analisi dello sviluppo dell'infezione su di una singola macchina si rende poi obbligatorio per la rimozione del malware che l'ha causata. Generalmente non è infatti assolutamente sufficiente la rimozione del binario dell'applicazione indesiderata, molto più spesso è necessario scovarne i residui nel sistema ed eliminare le alterazioni che esso ha causato. Tutte queste azioni richiedono una conoscenza dettagliata del codice maligno e del suo funzionamento.

Finora si è parlato di produttori di suite antivirus, in quanto appaiono come i maggiori interessati a queste tematiche, ma il problema non è tuttavia assolutamente limitato a queste compagnie. La sempre maggiore diffusione di applicazioni identificabili come malware rende possibile la mancata individuazione di uno di essi: in tal caso starà al responsabile della sicurezza studiare l'infezione avvenuta e cercare di porvi rimedio.

In definitiva l'obiettivo dell'analisi del malware è capire come uno specifico frammento di codice malevolo si comporti, in modo da rendere possibili l'implementazione di difese che pongano la rete al sicuro dall'infezione. Ad infezione avvenuta sono invece due le domande a cui è necessario rispondere per rimediare al problema: come la macchina sia stata infettata e che azioni intraprenda il malware che ha causato l'infezione.

L'approccio tradizionalmente usato per l'analisi comportamentale di un programma sconosciuto consiste nell'esecuzione del programma in un ambiente isolato e nell'osservazione delle sue attività [14]. L'ambiente è spesso costituito da un debugger, utilizzato da un analista umano per scorrere manualmente il codice ricercandone le funzionalità. La mole di nuovi sample che i vendor di antivirus ricevono ogni giorno rende però un procedimento manuale di questo tipo impossibile da applicare, si rende così necessario lo sviluppo di piattaforme automatizzate che siano in grado di raccogliere dati sufficienti senza la costante supervisione di un analista.

Un metodo per automatizzare questa analisi è eseguire il malware in una macchina virtuale isolata dal mondo esterno[10]. Durante l'esecuzione del programma le sue interazioni con il sistema operativo vengono memorizzate e successivamente riportate all'analista. In tal modo l'essere umano viene sgravato dall'oneroso compito di studiare l'intero codice, operazione che può in alcuni casi rendersi comunque necessaria anche utilizzando una tecnica di questo tipo. Il report presentato contiene infatti informazioni che aiutano nell'esame del codice malevolo ma che non possono tuttavia prescindere da uno studio attento da parte dell'analista.

Gli approcci generalmente utilizzati per l'automatizzazione dell'analisi di dataset di malware presentano tuttavia delle problematiche non trascurabili.

Un problema è costituito dal fatto che generalmente il malware è dotato di meccanismi di controllo dell'ambiente in cui viene eseguito, essendo così in grado di rilevare l'eventuale presenza di una macchina virtuale. In tal caso il malware potrebbe alterare il suo comportamento rendendo i risultati prodotti dall'analisi nella migliore delle ipotesi inaccurati. Alcuni codici maligni sono in grado di controllare la presenza di breakpoints indici della presenza di un debugger [12]. Questo richiede che l'ambiente di analisi sia invisibile al malware in fase di studio.

Un altro problema potrebbe presentarsi qualora tale ambiente dovesse rivelarsi inadeguato per l'osservazione del software in esame. In questo caso il malware potrebbe ingannare l'analisi e non essere riconosciuto come codice malevolo. Ciò rende necessario che l'ambiente sia il più completo possibile e che copra ogni interazione tra sistema operativo e malware.

2.1.1 Tecniche di analisi statica

L'analisi di eseguibili sconosiuti non è un problema che si è diffuso solo recentemente. Molte soluzioni esistono già e possono essere suddivise in due

catogorie: tecniche di analisi statica e tecniche di analisi dinamica. In questa sezione ci occuperemo dell'analisi statica.

L'analisi statica è il processo di analisi del codice di un programma senza che esso venga eseguito[13]. In questo processo il binario è generalmente disassemblato, cioè il codice binario viene tradotto in istruzioni assembler. Quindi possono essere applicate tecniche di control flow e data flow per tentare di stabilire il funzionamento del programma.

Tale tipo di analisi presenta il vantaggio di poter coprire l'intero codice del programma e risulta generalmente più veloce della propria controparte dinamica. Ciò nonostante un problema dell'analisi statica è che molte domande circa il funzionamento di un programma non sono in grado di trovare risposta senza la sua effettiva esecuzione. Esistono studi che dimostrano come, sfruttando la generale linearità dei comuni software, molti di questi problemi possano essere efficacemente approssimati nella pratica. Sfortunatamente la situazione è ben diversa quando si ha a che fare con dei malware[14]. Il codice maligno può infatti essre scritto con il chiaro intento di renderlo il meno facile possibile da esaminare. In particolare il creatore di software di questo tipo può utilizzare tecniche di offuscamente per ostacolare il disassembler e l'interpretazione del codice così ottenuto.

Il termine offuscamento fa riferimento a tecniche in grado di preservare le funzionalità del programma rendendo però difficile all'analista estrarne e comprenderne la struttura. Oltre a ciò il codice può essere deliberatamente studiato per renderne la comprensione il più difficile possibile. L'idea base per tali metodologie di offuscamento è che possano essere applicate automaticamente ma che le modifiche da esse apportate non possano essere facilmente annullate, anche nel caso l'analista dovesse conoscere le modalità di mascheramento del codice utilizzate.

Una possibile tecnica per realizzare l'offuscamento è l'utilizzo di particolari primitive dette "opaque constants", che altro non sono che un'estensione dell'idea degli "opaque predicates", definiti come "espressioni booleane i cui valori sono conosciuti all'offuscatore ma di difficile determinazione per metodi automatici di de-offuscamento" [14]. La differenza tra costanti e predicati di questo tipo risiede nel fatto che le "opaque constants" non assumono valori booleani ma interi. Nello specifico, le "opaque constants" sono meccanismi utilizzati per caricare nei registri del processore valori che non è possibile determinare staticamente. Utilizzando questo metodo è possibile costruire trasformazioni difficili da analizzare staticamente. Ad esempio un attaccante potrebbe sostituire le destinazioni di istruzioni di tipo jump o call con "opaque constants".

Oltre a ciò può capitare il caso in cui un analizzatore statico non sia in grado di esaminare il codice che effettivamente sarà eseguito. In particolare ciò è vero per programmi automodificanti che utilizzino tecniche polimorfiche [16] o metamorfiche e per software che utilizzino packer [15] per estrarsi solo in fase di esecuzione .

2.1.2 Tecniche di analisi dinamica

Contrariamente a quanto avviene durante un'analisi statica, le tecniche dinamiche prevedono l'analisi del codice in fase di run-time[17]. Malgrado questo tipo di tecniche non possa permettere un'esame completo del codice, presentano il vantaggio di analizzare solo quelle istruzioni effettivamente eseguite dal malware. In più l'analisi dinamica è immune dalle tecniche di offuscamento e non presenta i problemi relativi ai programmi automodificanti.

Quando si opta per l'esecuzione di un'analisi dinamica il primo problema che si presenta è la selezione dell'ambiente migliore per l'esecuzione del campione del malware. Ovviamente non è pensabile eseguire il programma direttamente sul computer dell'analista nè su qualsiasi altra macchina connessa alla rete.

Utilizzare una macchina indipendente dedicata potrebbe essere una soluzione, ma gli enormi tempi di ripristino della macchina dopo ogni test non la rendono conveniente.

La scelta maggiormente diffusa è l'utilizzo di una macchina virtuale[10]: in questa configurazione il codice maligno può infettare solo il PC virtuale non intaccando la funzionalità della macchina fisica. Dopo aver eseguito un'analisi dinamica l'immagine del disco rigido infetto viene semplicemente cancellata e sostituita da una versione "pulita" (snapshot [11]). Le soluzioni basate su macchine virtuali sono in questo caso sufficientemente veloci da non presentare differenze significative rispetto all'utilizzo di una macchina reale, mentre invece i tempi di reinstallazione sono drasticamente ridotti. Sfortunatamente il problema in cui si può incappare con una metodologia di questo tipo è il rilevamento da parte del malware della macchina virtuale in cui viene eseguito: in tal caso potrebbe modificare il proprio comportamente rendendo inutilizzabile l'analisi.

Un'alternativa percorribile è l'utilizzo di un PC emulator[14]. Tale software è in grado di emulare ogni componente di un personal computer, compresi hard disk, processore, scheda video e ogni altra periferica per simulare l'esecuzione del sistema operatico su una macchina reale. Si noti che tale soluzione è differente rispetto all'utilizzo di una virtual machine: le macchine virtuali sono in grado di gestire solo un subset delle istruzioni realmente eseguibili da un processore, i PC emulator non presentano invece questo limite. Visto che ogni istruzione viene emulata via software non è possibile per l'eseguibile analizzato rilevare la differenza tra questo ambiente e una macchina reale.

L'unica caratteristica differente tra un sistema reale ed uno emulato è la differente velocità di esecuzione. Tale differenza può essere rilevata da applicazioni che si basino su metriche temporali per la rilevazione di ambienti simulati. Una possibile contromisura sarebbe la falsificazione delle informazioni sul clock per

simulare una maggiore velocità di esecuzione.

Una volta selezionato lo strumento software migliore per l'osservazione del funzionamento occorre stabilire che tipo di informazioni si vogliano ottenere dall'ambiente di test. Esistono sistemi che concentrano le proprie attenzioni sull'interazione tra l'applicazione ed il sistema operativo controllando le chiamate di sistema. Sono diffusi ad esempio strumenti in grado di monitorare tutti i processi attivi nell'ambiente di test, oppure tool dedicati all'analisi delle attività del registro di Windows o del file system. Queste applicazioni sono implementate come driver che intercettano le chiamate al sistema operativo, risultando così invisibili al software sotto esame.

2.2 Botnet

La rete internet è, per sua natura, caratterizzata da un numero finito di risorse. Gli attaccanti hanno imparato a sfruttare questa caratteristica congestionando computer e servizi di rete tramite richieste fittizie, al fine di impedirne la fruizione agli altri utilizzatori.

Tali attacchi, denominati Denials of Service (DoS), si sono evoluti passando da modelli singolo attaccante contro singolo bersaglio a meccanismi più sofisticati, caratterizzati da molteplici attaccanti verso un solo bersaglio, fino ad arrivare a quello che oggi è conosciuto come Distribuited Denial of Service (DDoS). Tale tipo di attacco prevede la gestione di un gran numero di host attaccanti verso una vittima[18].

Gli strumenti e la tecnologia per attacchi di tipo DoS è a mano a mano diventata più semplice da reperire ed utilizzare. Il trend che si è andato affermando negli ultimi anni prevede un modello costituito da pochi attaccanti che, infettando un gran numero di host collegati ad internet, li sfutta per compiere attacchi di tipo DDoS.

Malware caratterizzati da un comportamento di questo tipo sono detti bot; una rete composta da numerosi bot è detta botnet.

I criteri utilizzati per la selezione di vittime da trasformare in bot sono high bandwidth ed high availability. I bersagli potenziali sono non solo server universitari, ma anche ISP e home users dotati di connessioni a banda larga.

Su internet sono liberamente scaricabili migliaia di pacchetti preconfezionati altamente configurabili e personalizzabili per la creazione di botnet. Tali software spaziano da codici altamente specializzati "fatti in casa" a soluzioni più complete ed estese adatte ad una gran quantità di applicazioni. L'usuale meccanismo di protezione da infezioni di bot consiste nell'utilizzo di pacchetti antivirus, le cui capacità di riconoscimento sono però limitate alle varianti di malware conosciute.

2.2.1 Attività delle Botnet

A seguire si presentano alcune delle attività tipicamente compiute dagli attaccanti usando le botnet.

- *Attacchi di Distributed Denial of Service*[18]: questo è il tipo di attacco più frequente che viene effettuato tramite le botnet. L'attaccante può utilizzare il proprio esercito di macchine infette per inviare enormi quantità di pacchetti UDP, richieste ICMP o TCP sync verso i server obiettivo dell'attacco. Con migliaia di zombie (così sono dette le macchine infettate da un bot) al proprio comando l'attaccante è così in grado di occupare l'intera banda del server impedendogli di rispondere alle richieste legittime.

- *Infezioni locali*: tramite l'installazione del bot l'attaccante prende pieno possesso della macchina della vittima, su cui potrebbe installare ad esempio trojan o key logger [19] al fine di rubare informazioni preziose dalla

risorsa infettata, quali ad esempio possono essere numeri di carta di credito o altre informazioni personali.

- *Commercio di banda*: un'altro interessante utilizzo del PC infetto è lo scambio di banda degli high speed bots (macchine con connessione a banda larga) tra diverse comunità hacker.

- *Backdoor*: i bot vengono installati sulle macchine compromesse come backdoor per mantenere l'accesso anche dopo l'exploit, specialmente nel caso in cui nella rete sia già presente del traffico di rete legittimo. L'attaccante potrebbe configurare i bot per utilizzare la stesse porte del traffico legittimo, riducendo la possibilità di essere rilevato dagli amministratori di sistema.

- *Hosting di materiale illegale*: è un fenomeno molto diffuso che prevede l'utilizzo dei bot per rendere gli host infettati parte di reti di file sharing ed utilizzare le loro risorse come hosting di file illegali.

Il tracciamento degli attaccanti responsabili dell'installazione di bot e del loro utilizzo con fini illeciti è molto difficile e raramente perseguito da parte degli ISP o degli amministratori di sistema. Appare però chiaro che il tipo di utilizzo chè può essere effettuato di una botnet dipende solo dalla fantasia dell'attaccante.

2.2.2 Bersagli principali e vittime

Gli attaccanti alla ricerca di host da infettare sono principalmente guidati da alcune precise caratteristiche, che qui di seguito presentiamo.

- *Elevata larghezza di banda*: una delle principali caratteristiche che rendono un host appetibile è la disponibilità di banda della connessione internet, la cui ampiezza consente di potenziare un attacco di tipo DDoS o di scaricare software non legali ad alte velocità.

- *Availability*: l'attaccante predilige macchine sempre attive e connesse alla rete, in modo da poter gestire secondo le proprie preferenze la botnet in qualsiasi momento.

- *Bassa consapevolezza del rischio e capacità di individuazione dell'infezione*: utenti con scarsa sensibilità alle tematiche della sicurezza su internet o con scarse risorse a disposizione per un monitoraggio efficace sono ovviamente bersagli appetibili per l'infezione. Sistemi operativi ed applicazioni non aggiornati o la mancanza di una adeguata protezione tramite firewall offrono all'attaccante la possibilità di fare breccia nel sistema e mantenere il controllo della macchina per lunghi periodo di tempo senza essere individuato.

- *Localizzazione geografica*: l'attaccante bersaglia più facilmente macchine geograficamente lontane da lui e con minima probabilità di tracciamento da parte delle forze dell'ordine.

Il profilo tipico che soddisfa tutti i criteri sopra presentati è quello di un utente domestico dotato di connessione broadband o di server universitari sempre attivi connessi a banda larga.

2.2.3 I bot basati su IRC

La struttura di comunicazione ad oggi più utilizzata [5] per la gestione di botnet è basata sulle reti di Internet Relay Chat (IRC)[20], utilizzate come centri virtuali di comunicazione e controllo (C&C) verso le macchine infette. La larga diffusione di reti IRC aiuta i malintenzionati a nascondere le proprie attività dietro la facciata di traffico IRC legittimo.

Tipicamente un bot, una volta installato su una macchina vittima, stabilisce una connessione in uscita verso la porta IRC di un server e si unisce ad un canale privato. La presenza di numerose reti IRC pubbliche consente agli attaccanti di gestire un'infrastruttura stabile e scalabile per il mantenimento, l'espasione ed il controllo della botnet formata.

2.2.3.1 La rete IRC

IRC è un protocollo internet sviluppato in Finlandia nel 1988, con funzioni base che consentono a qualsiasi utente connesso ad internet di unirsi a discussioni testuali in real time[20]. Ogni discussione risiede su un "canale" e molte persone possono accedervi contemporaneamente. Il protocollo IRC è basato su un'architettura client-server e ben si adatta all'esecuzione su svariate macchine in maniera distribuita.

Una configurazione tipica include un singolo processo (il server) che costituisce il punto centrale a cui i client si connettono, eseguendo la distribuzione dei messaggi e le altre funzioni ad essa correlate.

Per accedere ad una chat IRC l'utente esegue un programma detto "IRC client" che si connette ad un "server" della rete IRC . Tutti i server sono interconnessi e distribuiscono i messaggi da utente ad utente da un punto all'altro della rete. Un server può essere connesso a centinaia di client. La porta stan-

dard per IRC è la 6667 ed in generale i server IRC restano in ascolto sulle porte da 6000 a 7000, anche se possono essere configurati per l'ascolto da qualsiasi porta TCP.

In una delle modalità di comunicazione disponibili molteplici IRC clients si connettono a server IRC per formare un gruppo detto "canale". In questo caso la comunicazione inviata da ogni client al server è inoltrata a tutti i client connessi a quel determinato canale [21].

2.2.3.2 Elementi tipici di un attacco IRC Bot

I bot sono utilizzati dagli attaccanti per infettare le macchine delle vittime dopo che sono state compromesse o l'utente è stato ingannato portandolo ad effettuare l'installazione. Il bot, non appena infettata la vittima, si unisce al canale IRC per cui è stato configurato e si mette in attesa di comandi da parte dell'attaccante. Il meccanismo di funzionamento è rappresentato dall'immagine 2.1.

Vediamo ora la terminologia utilizzata:

- *Bot*: tipicamente un file eseguibile, in grado di eseguire un insieme di funzioni, ognuna delle quali può essere attivata da un comando specifico. Un bot, una volta installato sulla macchina vittima, copia se stesso in una cartella predeterminata e modifica la configurazione del sistema per garantire la propria esecuzione all'avvio. Un bot di tipo off-the-shelf, utilizzato generalmente da attaccanti non molto sofisticati, può essere scaricato da comuni siti di warez ed essere personalizzato impostando il server IRC desiderato, la porta TCP da utilizzare e il nome del canale a cui accedere, oltre ovviamente alle credenziali di accesso. Un attaccante con più esperienza potrebbe anche manipolare il bot, modificando file e cartelle di

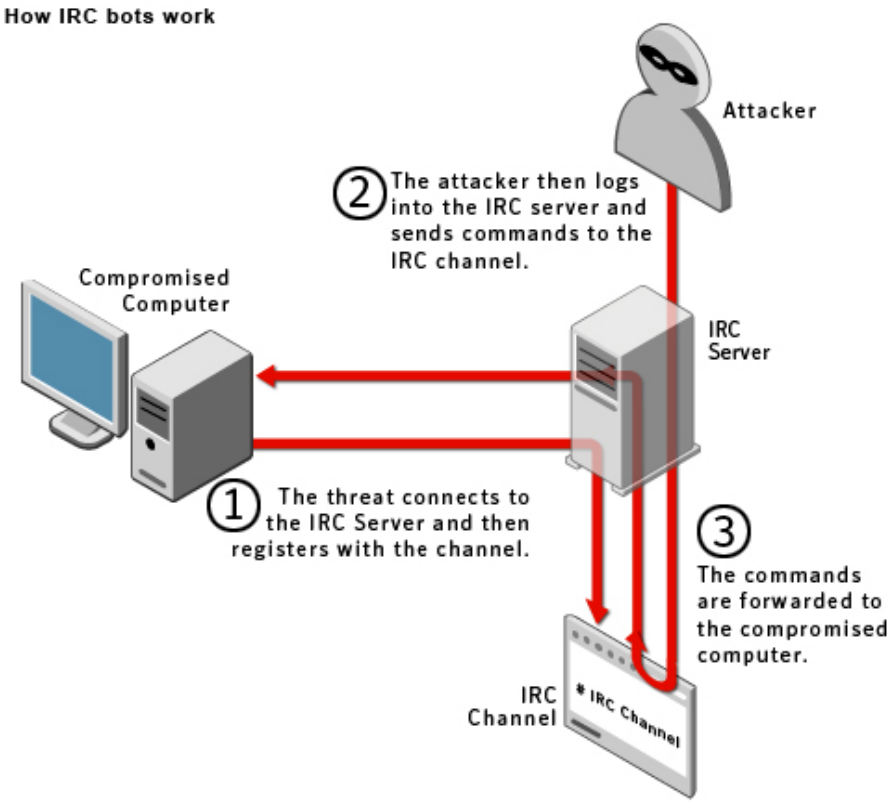

Figura 2.1: Meccanismo di funzionamento di un IRC bot

installazione per rendere meno visibile l'infezione. Una nota importante è che i bot non sono exploit per sistemi operativi od applicazioni, sono invece il payload di worms o affini utilizzati per installare backdoor sulle macchine infette.

- *Macchina Vittima*: è l'host compromesso sui cui il bot malevolo risiede dopo che l'attaccante è riuscito in qualche modo a forzarne l'installazione. Una volta infetto questo host è detto anche "Zombie".

- *Attaccante*: è colui che configura il bot, seleziona la macchina bersaglio dell'infezione, controlla e dirige il bot una volta che quest'ultimo si è unito al canale IRC.

- *Canale di controllo*: è un canale IRC privato creato dall'attaccante come punto d'incontro per tutti i bot una volta installati sulle macchine compromesse ed online, è caratterizzato da un nome e da una password per l'autenticazione.

- *Server IRC*: è un server che fornisce servizi IRC, potrebbe essere un fornitore di servizi pubblici oppure un'altra macchina compromessa

- *Botnet*: tutti i bot una volta connessi ad un canale di controllo formano una botnet.

2.2.3.3 Meccanismi di infezione e controllo

Il processo di costruzione di una botnet ha inizio, in base alle capacità dell'attaccante, o configurando un bot scaricato da internet oppure scrivendo di persona il codice del malware.

Le variabili principali da impostare sono l'indirizzo del server a cui il bot si dovrà connettere una volta installato, la porta TCP su cui il server IRC è

in ascolto, il nome del canale e la password per l'autenticazione all'accesso del canale. Oltre a questo, in base al tipo di bot utilizzato, l'attaccante potrebbe cambiare cartelle e nomi dei file dell'eseguibile per nascondere l'infezione. Altra possibilità è quella di impostare molteplici canali o server per la struttura di controllo tramite IRC: in questo modo il responsabile dell'infezione potrebbe tutelare il proprio accesso alla macchina compromessa anche nel caso un canale od un server risultasse non più disponibile. Per ottenere un comportamento di questo tipo gli attaccanti sono soliti usare servizi di DNS dinamico, ponendosi così nella condizione di poter selezionare ogni volta l'indirizzo IP del server migliore da utilizzare [22].

Una volta configurato il bot l'attaccante procede al tentativo di infezione attraverso l'utilizzo di exploit presenti sulla macchina della vittima oppure ingannando l'utente forzandolo ad installare egli stesso il malware. Il metodo tipico utilizzato per infettare svariati host contemporaneamente è l'utilizzo di exploit basati su vulnerabilità recenti; tale processo può essere automatizzato tramite un worm che scansioni le sottoreti selezionate ricercando le vulnerabilità conosciute ed eseguendo quindi un exploit che porti all'installazione del bot. Un altro metodo consiste nello sfruttare applicazioni web vulnerabili per ingannare l'utente ad eseguire programmi o virus che portano all'infezione il PC vittima. La vittima potrebbe per esempio pensare di installare un comune client IRC che potrebbe però essere stato infettato con un trojan.

Durante l'installazione sulla macchina il bot si copia nella directory di installazione ed aggiorna le chiavi di registro nel caso di sistemi Windows. Successivamente il bot tenta di connettersi ad un server IRC con un nick casuale e la password impostata in fase di configurazione, entrando quindi nel canale controllato dall'attaccante.

Spesso i malintenzionati utilizzano server IRC pubblici per la gestione della

botnet, incappando nel rischio di un ban e della conseguente perdita di controllo dell'infrastruttura costruita. Proprio per evitare questi incidenti è frequente l'utilizzo di servizi quali "dyndns.com" o "no-ip.com", per reindirizzare dinamicamente i bot verso molteplici server IRC.

Una volta entrato nel canale IRC di controllo, il bot si pone in attesa di istruzioni.

L'attaccante accede al canale tramite le proprie credenziali ed è da subito in grado di controllare da remoto le azioni di tutti gli Zombie infettati. In tal modo può cominciare a sferrare attacchi verso altri bersagli oppure installare altri malware sui PC già compromessi.

2.2.3.4 Caratteristiche di una botnet basata su IRC

Il modello fin qui presentato è un modello di tipo centralizzato, di cui il Command & Control (C&C) server rappresenta il cuore.

L'utilizzo della rete IRC presenta innegabili vantaggi:

- bassa latenza delle comunicazioni;
- comunicazione in real time anonima;
- possibilità di comunicare in modalità gruppo oppure con singoli host;
- semplicità di configurazione;
- semplicità di controllo;
- flessibilità delle comunicazioni.

Tuttavia il sistema fin qui utilizzato non è esente da punti deboli, nello specifico possiamo individuare la principale debolezza nel server C&C. Qualora qualcuno dovesse scoprire ed eliminare il server, l'intera botnet diverrebbe inutilizzabile.

Per ridurre tale pericolo i bot moderni hanno introdotti varie strategie di difesa. La prima è detta ip fast-flux [23] e consiste nel configurare il bot per la connessione ad un dominio anzichè all'indirizzo IP del server, che può così essere cambiato frequentemente. Ciò rende molto più difficile individuare e bloccare uno specifico server C&C. Tale modalità di funzionamento presenta comunque un single point of failure individuabile nel nome del dominio utilizzato: una volta bloccato viene abbattuta l'intera infrastruttura di controllo.

Un'ulteriore evoluzione del sistema qui presentato è detta domain flux. Tramite il meccanismo del domain flux ogni bot utilizza un algoritmo di generazione del dominio (DGA) per elaborare una lista di domini. Il bot scorre tale lista finchè non individua un dominio attivo, che sarà identificato dai bot come indirizzo del server C&C. Tale tecnica va sempre più diffondendosi ed è correntemente utilizzata da bot quali Kraken, Srizbi e Conficker. Sebbene molto brillante tale idea non è però esente da difetti. Il problema centrale di questa tecnica è infatti il DGA: se si riuscisse a prevedere il successivo dominio generato dalla lista e lo si potesse registrare prima dell'attaccante si potrebbe "rubare" la botnet. Tale pratica è stata in effetti testata con successo da un gruppo di ricercatori che è stato in grado di prendere possesso di una botnet basata su Torpig [23].

2.3 Le WAPI

Le WAPI [2] sono le Application Programming Interface sviluppate dal progetto WOMBAT per lo studio e analisi di malware

Scopo del progetto WOMBAT è sviluppare nuovi metodi per indagare le minacce rivolte alla rete Internet ed ai net citizen. Al fine di raggiungere questo obiettivo la ricerca si è rivolta in tre direzioni complementari:

1. raccolta in tempo reale di set di dati grezzi riguardanti gli aspetti di sicurezza della Rete;

2. sviluppo e applicazione di tecniche di analisi per l'arricchimento dei dati raccolti al punto 1;

3. identificazione delle cause e comprensione del fenomeno in esame.

La conoscenza acquisita attraverso le sopracitate operazioni è quindi condivisa con tutte le parti interessate, facilitando il dimensionamento e l'indirizzo degli investimenti destinati al settore della sicurezza. Più in generale il progetto mira alla creazione nell'utenza Internet di una sensibilità maggiore verso la problematiche relative alla Security.

All'interno di questo contesto è stato sviluppato lo strumento delle WAPI, un'interfaccia comune per l'accesso alle informazioni contenute nei differenti dataset curati dai partner del progetto.

Nello specifico i dataset disponibili alla consultazione tramite questa interfaccia sono 9, divisibili in 4 categorie:

- Tool per l'analisi dei malware
  - Anubis
  - Virustotal

- Honeypots server side
  - SGNET

- Honeypots client side
  - Honeyspider
  - Harmur
  - Wepawet
  - Shelia
- Altri
  - Forth
  - Bluebat

Nella sezione relativa ai tools per l'analisi dei malware possiamo contare su Anubis, una sandbox monitorata con accesso a internet liberamente fruibile dal web e caratterizzata da un elevato numero di analisi giornaliere. Un altro tool disponibile in questa sezione è il dataset di Virustotal, che può essere definito come un "aggregatore" di antivirus. Virustotal memorizza infatti il risultato delle scansioni di molteplici antivirus collegandoli al checksum md5 del file in esame. Il database è in continua espansione in quanto l'interfaccia per la sottomissione del file è pubblica e largamente utilizzata.

Passando alle honeypot abbiamo primo di tutto SGNET, una honeypot lato server destinata alla raccolta di informazioni relative agli attacchi di code injection.

Le honeypot lato client comprendono invece Honeyspider, un crawler molto sofisticato per l'analisi di malicious url, Wepawet, che non è un crawler ma che si focalizza sull'analisi degli script di un determinato indirizzo, e Shelia, un honeyclient per l'ispezione di email ed indirizzi web caratterizzato da un elevato grado di interazione.

Un discorso a parte sempre in questa categoria merita HARMUR, un framework per il crawler interno di Symantec che contiene una elevatissima quantità di dati, anche storici, sugli url esaminati.

Il dataset Forth è invece praticamente un ponte verso servizi non direttamente fruibili attraverso le WAPI. Nello specifico è molto utile per le informazioni relative alle query di tipo whois [24].

Bluebat non è stato invece utilizzato all'interno di questa analisi, in quanto si tratta di una honeypot basata su bluetooth destinata specificatamente a studiare questo metodo di infezione.

A seguire presentiamo un approfondimento sui dataset effettivamente utilizzati nel corso dello studio da noi condotto, esaminandone le caratteristiche ed il funzionamento.

2.3.1 Anubis

Il dataset di Anubis consta di un elevato numero di report derivanti dall'esecuzione del malware in una sandbox; tale sandbox, in quanto disponibile e liberamente accessibile dalla rete internet, è caratterizzata da un cospicuo numero di sottomissioni giornaliere che consentono una costante espansione del dataset stesso.

La sandbox si presenta come un tool per l'analisi comportamentale di eseguibili per sistemi Windows, prestando un'attenzione speciale all'esame del malware. La sottomissione di un file ad Anubis porta alla generazione di un report contente informazioni sufficienti a consentire all'utente di comprendere scopi e azioni del malware oggetto di studio. Il report che viene generato include dati riguardo le modifiche al file system ed ai registri di Windows, i nuovi processi in esecuzione e soprattutto un log contenente tutto il traffico di rete causato

dall'infezione.

L'analisi del comportamento del file in esame viene effettuata lanciando l'eseguibile in un ambiente Windows emulato ed osservandone l'attività. Le informazioni raccolte si concentrano su aspetti rilevanti nell'ambito della sicurezza così da rendere il processo molto più snello e orientato al campo applicativo di interesse dell'utente finale. Come facilmente intuibile il fulcro dell'intero dataset è il malware, i dati si dipanano da esso fino a coprire tutte le informazioni indispensabili alla ricerca. Così come nel funzionamento generale delle WAPI, anche nel dataset di Anubis il file è identificato tramite il proprio checksum md5.

Ad ogni determinato malware è assegnato un elenco di task, che rappresenta le analisi ci cui è stato oggetto. A partire da questi task è possibile accedere alle informazioni relative alle modifiche sul registro di Windows, le modifiche al file system e il traffico di rete generato dalla sandbox. Nello specifico ai fini dell'analisi che verrà presentata a seguire si sono rivelati di estrema importanza i dati relativi al traffico IRC, http, UDP e TCP ed alle query DNS.

2.3.2 VirusTotal

Il dataset di VirusTotal costituisce una preziosa risorsa per l'identificazione dei file oggetto di studio. In modo analogo a quanto avviene con Anubis, il dataset altro non è che una collezione di report relativi all'analisi di sample di malware sottomessi dagli utenti. Anche in questo caso la disponibilità di un'interfaccia web liberamente fruibile consente un continuo ampliamento del malware conosciuto, garantendo un aggiornamento costante delle informazioni relative ai file esaminati.

VirusTotal è orientato all'identificazione del malware, che è ottenuta sottoponendo il file sotto indagine ad un'ampia gamma di antivirus e memorizzandone

i relativi output [25]. Tali informazioni permettono di confrontare le diverse denominazioni utilizzate dai produttori di antivirus e selezionarne il più idoneo allo studio che si sta conducendo.

Al momento attuale VirusTotal utilizza 43 antivirus per l'analisi: l'interfaccia della WAPI consente di visualizzare quanti di essi riconoscano l'md5 del file in esame e come si sia evoluto il detection rate per quello specifico file a partire dalla prima volta che è stato esaminato.

Si noti che tramite le WAPI non è possibile inviare effettivamente il sample di un malware per l'analisi: tale operazione è possibile solo interagendo con il sito web di VirusTotal o tramite email. Quella che invece viene effettuata è una ricerca nell'archivio contenente lo storico di tutti i report per il checksum md5 dell'applicativo.

Ovviamente qualora nel dataset non dovessero essere presenti report relativi al file in questione non è detto che si possa escludere che si tratti di malware: il file potrebbe non essere mai stato sottomesso a VirusTotal oppure, eventualità più remota, essere così recente da non essere ancora rilevabile dagli antivirus utilizzati.

Una caratteristica la cui importanza sarà spiegata più avanti è il fatto che tramite le WAPI questo dataset presenti sempre il report relativo a tutti gli antivirus disponibili: tale caratteristica facilita l'implementazione di algoritmi che possano stabilire quale vendor abbia il miglior rateo di riconoscimento relativamente ad una specifica famiglia di malware.

2.3.3 Harmur

Harmur, tra tutti i dataset utilizzati, è quello maggiormente orientato all'analisi di domini e server di cui si cerchi di determinare la natura malevola.

L'interconnessione tra le classi di questo dataset è così elevata che non è possibile identificare un vero e proprio punto di partenza per la fruizione dei contenuti, tuttavia appare sensato pensare che un'analisi condotta utilizzando Harmur debba partire da un dominio o da un server. Ad ogni dominio presente nel dataset è associato un elenco di server, lo storico cioè di tutti gli indirizzi IP host di quel dominio. Da ciascuno dei server è possibile risalire a tutti gli hosted domains; è inoltre possibile sfogliare un elenco di minacce individuate dal crawler di Harmur sulla macchina in questione.

Le informazioni disponibili sui domini comprendono data di creazione del dominio, data della prima osservazione, nome ed indirizzo email del registrant e generalità del registrar. Purtroppo tali informazioni si rivelano spesso incomplete, rendendo la funzione di ricerca di domini appartenenti allo stesso registrant impossibile da automatizzare.

Delle minacce ospitate dai server (dato comunque disponibile anche a livello di dominio) si conoscono tipo di minaccia, data della prima identificazione, descrizione e tag. I tag relativi al tipo di threat sono a fondamentali in quanto capita sovente che la descrizione generica non riporti informazioni significative: in tal caso è necessaria un'analisi dei tag per cercare di comprendere con quale tipo di minaccia si stia effettivamente avendo a che fare.

2.3.4 Forth

All'interno delle WAPI l'interfaccia di Forth rappresenta un "ponte" per servizi non WAPI-enabled. Nello specifico il dataset presenta per i domini informazioni analoghe a quelle individuabili tramite Harmur: registrant, registrar, date di creazione e modifica. Nello stesso modo a partire dal dominio è possibile risalire all'indirizzo IP del server (in questo caso non si ha però a disposizione un archivio

storico) e visualizzare i dati riguardo all'AS di appartenenza ed alle informazioni geografiche.

Sempre nell'ambito dell'analisi del server, Forth rileva se l'IP della macchina in esame appartenga ad una lista di attaccanti nota, se sia un server C&C riconosciuto o se sia responsabile di attività di spamming.

Un difetto di questo dataset è che molto spesso a partire da un dominio non è possibile risalire direttamente al suo server, anche quando questo è ancora attivo. Tale situazione è però facilmente risolvibile cercando manualmente l'indirizzo del server e quindi visualizzandone le informazioni con Forth.

Il fatto che le informazioni geografiche siano direttamente accessibili attraverso le WAPI rende possibile evitare l'utilizzo di soluzioni esterne come quali IP2Location, come invece avviene solitamente [26, 5].

Capitolo 3

Metodologia ed implementazione

La metodologia di analisi di seguito introdotta è rivolta allo studio di dataset di malware utilizzati per la creazione di bot.

Il lavoro sarà suddiviso in 3 fasi distinte come evidenziato in figura 3.1. Inizialmente verrà affrontato il problema dell'identificazione dei malware contenuti nel dataset, esaminando il nome con cui gli antivirus identificano i campioni in esame. Successivamente si procederà all'analisi delle richieste dns effettuate dai file presenti nel dataset, che porterà quindi ad un'analisi live dei server così individuati. In ultima analisi si approfondirà lo studio di quei domini che risultano essere caratterizzati da un'elevato numero di query da parte del dataset, si cercherà di capire se quei domini possano avere una qualche relazione e se alcuni di essi presentino delle peculiarità interessanti.

Ogni fase ha richiesto un attento lavoro di controllo dei risultati parziali, spesso portando allo sviluppo di tool ad-hoc per la revisione manuale dei dati. Sebbene una minima parte di queste operazioni di pulizia e trattamento dei

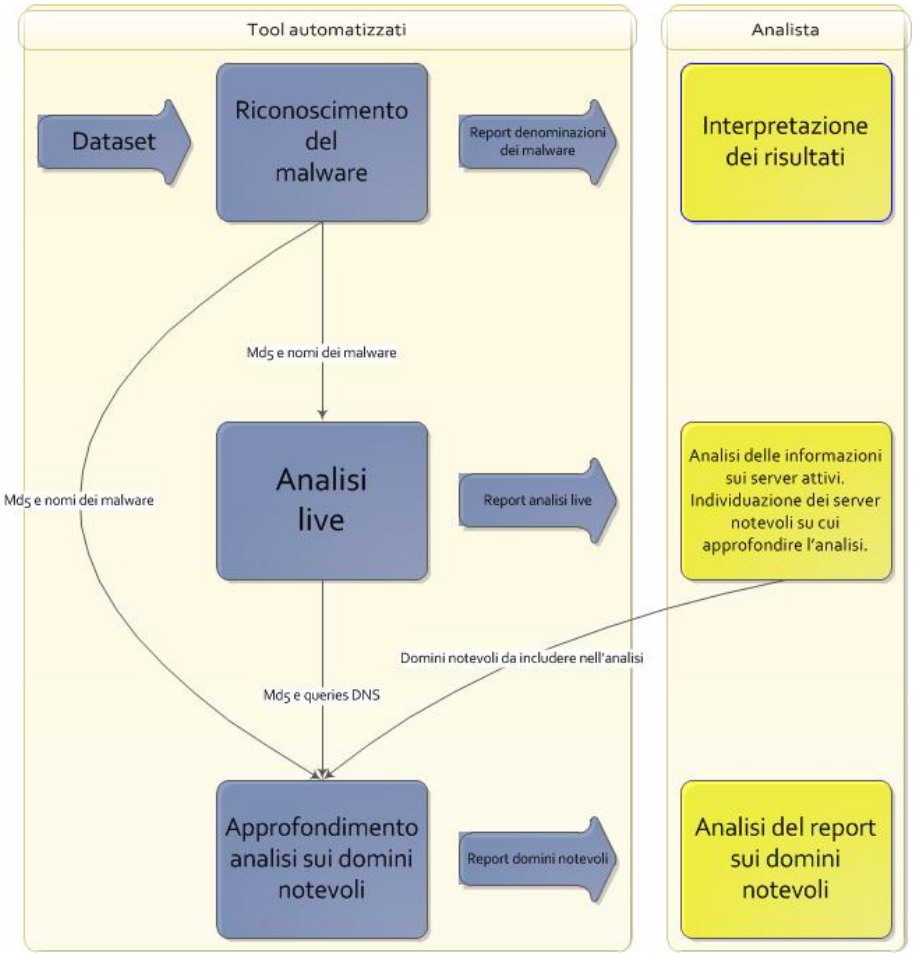

Figura 3.1: Fasi previste dalla metodologia di analisi

risultati possa essere automatizzata, l'attività di supervisione da parte del ricercatore è fondamentale per la corretta esecuzione del procedimento. Evidenziata quindi l'impossibilità di sviluppare un applicativo in grado di eseguire in totale autonomia l'analisi, quello che è possibile sviluppare è una serie di tool parziali che possano essere di supporto all'utente per le operazioni più generiche.

Nel seguito della trattazione verranno chiaramente evidenziati i momenti in cui l'intervento attivo del ricercatore è necessario.

3.1 Riconoscimento del malware

Al fine di sviluppare correttamente l'analisi del dataset sotto esame è innanzitutto necessario raccogliere informazioni su ciò che è già noto dei malware da studiare. Internet costituisce un'ottima fonte di informazioni a riguardo, ma prima di avviare una ricerca in tal senso si deve stabilire con quale tipo di malware abbiamo a che fare.

Il primo tool sviluppato è orientato proprio alla determinazione della natura dei malware da esaminare. Tale software ricerca gli md5 checksum dei file nel dataset, restituendo un report contenente le denominazioni con cui l'antivirus prescelto per l'analisi riconosce i file oggetto dello studio. In tal modo è quindi possibile ottenere la percentuale di rilevamento delle singole famiglie di malware.

Lo script è in grado di determinare automaticamente ed in tempo accettabile quale antivirus sia più indicato per l'analisi dei file. L'algoritmo utilizzato per la determinazione dell'antivirus migliore seleziona un sottoinsieme di tutti quelli disponibili. La scelta degli antivirus di questo sottoinsieme è ricaduta sui sei vendor che a nostro giudizio sono più rigorosi nell'assegnamento dei nomi ai malware individuati. Durante la ricerca è infatti capitato sovente che diversi produttori non rispettassero completamente le naming convention dichiarate, causando molta confusione nella successiva analisi dei risultati.

La funzione di ricerca dell'antivirus con la maggior detection rate per una determinata analisi costituisce un prezioso aiuto, tuttavia non deve intendersi come sostituta del buon senso e dell'esperienza del ricercatore. Diversi test condotti durante lo sviluppo dello studio presentano un rate di riconoscimento compreso tra il 39 ed il 46%.

La scelta manuale di uno qualsiasi degli altri antivirus è una strada sempre percorribile, tuttavia per i sei antivirus disponibili nella selezione automatica si applica, contestualmente alla ricerca dei riscontri nel dataset, una pulizia delle

stringhe relative ai nomi dei file, in modo da eliminare le informazioni relative al ceppo e semplificare l'interpretazione del grafico rappresentante il malware contenuto nel dataset.

Ad ogni esecuzione della ricerca viene automaticamente preparato un semplice report per il ricercatore, accompagnato da un file csv che verrà utilizzato per il proseguimento dell'analisi. Ciò che ci si aspetta a questo punto è una chiara individuazione di quali malware siano presenti all'interno del dataset. I risultati posso essere più o meno chiari a seconda dell'antivirus utilizzato e del tipo di file in esame. Ciò che può rendere meno chiara l'interpretazione dei dati sino a qui raccolti è la presenza di risultati basati sulle euristiche degli antivirus. Capita infatti di frequente che un determinato virus non venga identificato con il suo nome ma venga riportato secondo l'euristica che lo ha individuato. Ovviamente in questo caso non c'è alcun modo di sostituire l'interpretazione umana dei risultati: sta alla sensibilità dell'utente stabilire se alcune delle categorie individuate possano afferire allo stesso malware.

Discorso analogo può essere applicato alla determinazione dei ceppi presenti nel dataset. Eccezion fatta per i sei antivirus di cui si è parlato precedentemente, la differente nomenclatura usata dai produttori di antivirus non rende possibile lo sviluppo di un metodo generale per stabilire se file riconosciuti con nomi diversi possano essere ceppi dello stesso worm. Anche in questo caso sta al ricercatore verificare tale possibilità ed aggiornare di conseguenza l'analisi. Si noti che il problema delle mancanza di una nomenclatura comune tra i produttori di antivirus non è una limitazione esclusiva dell'approccio qui presentato, bensì un problema generale ben noto nel mondo della sicurezza informatica.

Ovviamente se la presenza di più ceppi dovesse rendere meno chiaro il risultato dell'output (per garantire una maggiore leggibilità il numero massimo di fette presenti nel grafico a torta è 8) è sempre possibile per l'analista ope-

rare direttamente sui dati grezzi presenti nei file di output creati dal tool per migliorare la rappresentazione.

Lo script consente all'utente di selezionare la cartella contenente i file da esaminare e quale antivirus utilizzare per la ricerca dei risultati. Gli output sono invece un file csv contenente md5 del file esaminato e la denominazione con cui viene riconosciuto, un grafico a torta che rappresenta la percentuale dei malware presenti all'interno dei file (per i sei antivirus disponibili nella ricerca automatica tali risultati sono già filtrati con l'eliminazione del ceppo) ed un report contenente il grafico e le percentuali di rilevamento dei malware individuati.

La differenza rispetto alla visualizzazione del solo grafico è che in questo report si visualizzano tutti i malware rilevati, non solo i sette più presenti. I produttori di antivirus per i quali è stata implementata la ricerca automatica sono Sophos, Microsoft, Antiy-AVL, Avast, Kaspersky e F-Prot. Data la lentezza delle operazioni di hashing dei file e del reperimento dei risultati tramite la WAPI è stata implementata all'interno dello script una progress bar contenente una stima del tempo residuo necessario. Tale stima è solo indicativa ma comunque sufficientemente affidabile.

3.2 Analisi live dei domini

Passo successivo nell'esame del dataset è la raccolta di informazioni sui domini a cui tali malware si connettono. In questo fase l'analisi può cominciare a prendere un numero considerevole di direzioni diverse. Anche in questo caso è perciò richiesta una certa discrezionalità da parte dell'analista.

L'approccio seguito in questo lavoro parte dal reperimento degli md5 dei file da esaminare, come nel caso appena presentato. Avendo già effettuato l'hashing nella fase precedente non si ritiene necessario ripetere ancora tale operazione. Tuttavia al fine di rendere quest'analisi indipendente si fornisce la duplice

possibilità di sfruttare il file csv generato dalla script di ricerca dei nomi dei malware oppure di ripetere l'hashing dei file della directory, in caso non fosse stato effettuato precedentemente.

L'idea base è controllare le dns queries dei malware per verificare verso quali domini cerchino una linea di comunicazione. Risalendo quindi a questi domini se ne verifica la natura e si controlla se abbiano delle particolarità o dei collegamenti fra loro. In particolar modo appare interessante verificare se vi siano un gruppo ristretto di individui od organizzazioni dietro la maggior parte del traffico generato dai malware.

A partire dal checksum del file di nostro interesse, l'interfaccia delle WAPI presenta una funzione per la ricerca delle dns queries effettuate dall'eseguibile analizzato. Siamo così in grado di identificare i domini oggetto delle queries come canale di comunicazione per il controllo del malware. Malgrado gli innegabili vantaggi riscontrabili in un procedimento che segua questo metodo, è d'obbligo osservare che durante il caso di studio da noi condotto non è stato possibile procedere in questi termini. Il problema nel nostro caso risiede nel fatto di avere utilizzato una versione dimostrativa del server WAPI, il cui database non era aggiornato da molto tempo; tale situazione non ha consentito di rilevare i dati relativi alla totalità del traffico di rete dei malware né di raccogliere un numero sufficiente di campioni per il proseguimento dello studio.

Si è reso quindi necessario un procedimento alternativo. Come già detto Anubis dispone di un'interfaccia web per la sottomissione di file e la creazione automatica di un report relativo alla loro esecuzione su di una sandbox. Tutti i file in nostro possesso sono stati quindi inviati per l'analisi, ottenendo un report xml per ogni file esaminato.

In base all'esperienza maturata con lo strumento delle WAPI siamo in grado di stabilire che in tali risultati saranno presenti una gran quantità di risultati

classificabili come "bad requests". È successo molto spesso di osservare query rivolte ad indirizzi del genere di "wpad" oppure "...". A tal proposito all'interno del tool è proposta innanzitutto una scrematura delle dns queries che porti all'eliminazione dei risultati evidentemente scorretti. Contestualmente a questa operazione si è ritenuto opportuno estrarre dalla query dns il dominio di secondo livello dell'url, in modo da poter operare direttamente a livello di dominio come consentito dall'interfaccia delle WAPI.

Si noti che la procedura sin qui seguita in questa fase è potenzialmente applicabile anche ad una qualsiasi lista di domini su cui si voglia effettuare un'indagine.

Una volta ottenuti i domini di secondo livello di tutte le query prese in esame occorre innanzitutto una visualizzazione immediata della distribuzione delle query su cui si sta operando. A tal proposito il tool sviluppato genera un grafico riportante il numero di query che fanno riferimento ai domini in esame.

Nel caso il dataset dovesse presentare un'elevata eterogeneità ci si più aspettare una distribuzione più o meno uniforme delle query sui domini osservati. Al contrario un dataset contenente poche famiglie di malware o dal comportamento simile ci si aspetta presenti un comportamento sul modello di una "long tail", come nel caso di studio da noi osservato. Per analizzare meglio questi dati, nel report che viene creato vengono riportati il numero totale di query valide ed il numero di domini presi in esame, così come il numero di query riscontrate per ogni dominio.

3.2.1 Individuazione dei server

Una volta osservato il comportamento delle queries effettuate dai file in esame siamo in grado di esplorare più a fondo l'ecosistema che si sta poco a poco rivelando tramite l'osservazione dei server su cui questi domini vengono ospitati. Il primo passo dello studio di questi server è l'individuazione degli IP address

di quei server che tuttora risultano attivi.

Lo strumento che l'interfaccia delle WAPI permette di utilizzare a questo proposito è Forth, che come già detto costituisce un'insieme di utilità per l'analisi dello spazio web. Sebbene molto comoda questa soluzione si è rivelata ben presto inappropriata: un confronto manuale dei domini rilevati da Forth con i domini effettivamente esistenti ha infatti evidenziato notevoli mancanze nel dataset, rendendo necessario procedere diversamente. La ricerca dei server viene quindi effettuata tramite una dns query, esaminando poi i server così ottenuti tramite Forth. Le informazioni che Forth mette a disposizione a riguardo dei server di nostro interesse, a differenza dei dati sui domini, risultano complete ed affidabili.

Nello specifico ciò che ci interessa al momento è individuare la distribuzione geografica dei server in esame, gli Autonomous System a cui appartengono gli IP, l'eventuale presenza di server già identificati come malevoli e le versioni dei web server utilizzati. La distribuzione geografica viene analizzata a livello di nazione, nonostante le WAPI mettano a disposizione informazioni sulle coordinate GPS del server. Un'eventuale analisi approfondita per il reperimento di tali dati può in un secondo momento essere effettuata manualmente dal ricercatore qualora dovesse ritenere importante un'individuazione più precisa del server. Soprattutto nel caso dovesse presentarsi un elevato numero di server localizzati nella medesima nazione.

Negli studi da noi condotti un livello di dettaglio così approfondito non si è rilevato necessario data la presenza dei server in numerosi paesi differenti. Per quanto riguarda gli Autonomous System (AS), si ricorda innanzitutto che si tratta delle organizzazioni che detengono il controllo di un determinato pacchetto di indirizzi IP.

La determinazione degli AS ai quali gli IP in nostri possesso fanno riferimento può consentire di individuare una o più organizzazioni caratterizzate da un

elevato livello di tolleranza nei confronti del fenomeno in fase di studio. L'individuazione di tale comportamento può costituire una base per un'indagine più approfondita della attività legate all'organizzazione individuata. A tal proposito occorre però spendere alcune parole riguardo agli AS appartenenti ad organizzazioni che forniscono servizi di DNS dinamico. Tali organizzazioni saranno facilmente incontrate in gran numero in un'analisi di questo tipo, prestandosi particolarmente ad utilizzi illeciti da parte di individui od organizzazioni che abbiano istituito delle botnet.

Nella fase successiva ci si propone di stabilire se le reti a cui si appoggiano i server da noi individuati abbiano introdotto una sorta di "standardizzazione" all'interno del sistema. Nello specifico si intende rilevare se sia presente un qualche web server predominante rispetto ad altri, in quanto l'uso di un template standard potrebbe indicare un'organizzazione comune dietro a più server malevoli. Ovviamente i dati qui ottenuti devono essere confrontati con i dati disponibili circa le percentuali di diffusione dei web server su scala globale. Ad esempio se i server malevoli esaminati dovessero presentare un'incidenza del 75% di un determinato web server contro una media globale del 20% il dato potrebbe risultare estremamente interessante.

L'utilizzo di Forth per lo studio dei server rende disponibili altri strumenti efficaci per la determinazione della natura dei server. Nello specifico tale dataset consente di rilevare l'eventuale presenza dell'IP del server in oggetto in una lista nota di IP malevoli, identificati in 3 classi: C&C server, spammer ed attacker.

Data la natura dei malware per cui questo tool è stato studiato ci si aspetta un'elevata presenza di C&C server, tuttavia non è affatto da escludere che tali server possano esseri attivi anche nel campo dello SPAM. Un'eventuale correlazione tra questi due dati si rivelerebbe interessante.

Come nel caso dell'individuazione delle famiglie di malware, contestualmente

alla creazione di un report viene generato un file csv contenente i dati grezzi raccolti durante l'analisi automatica, al fine di consentire una revisione manuale dei dati presentati dal report.

3.3 Approfondimento dell'analisi sui domini notevoli

L'analisi fin qui condotta sui domini individuati si concentra in particolar modo su quei server che risultano ancora attivi al momento dell'osservazione. Facendo riferimento al grafico relativo alle dns queries generato alla sezione precedente è quindi possibile che i domini caratterizzati da un elevato numero di richieste ma non più in attività rimangano esclusi dalle osservazioni fin qui compiute.

Lo strumento utilizzato fornisce tuttavia qualcosa in più, cioè la possibilità di recuperare informazioni storiche sui domini da noi presi in considerazione. Tale possibilità deve però essere sfruttata intelligentemente, in quanto l'enorme mole di informazioni ricavabili da questo dataset rischia di far perdere di vista l'obiettivo principale, cioè lo studio del comportamento dei malware da esaminare e le eventuali relazioni esistenti tra malware diversi.

Sostanzialmente ciò su cui si deve porre innanzitutto attenzione in questa fase è la selezione del giusto numero di domini da esaminare. Un numero troppo esiguo potrebbe facilmente non rivelare risultati significativi, al contrario un numero troppo elevato richiederebbe un tempo di elaborazione esagerato non garantendo comunque la leggibilità dei risultati.

Al fine di evitare tale eventualità prima dell'esecuzione del tool è necessario determinare una soglia per le queries dns: i domini interessati da un numero di queres superiore alla soglia saranno inclusi nell'analisi mentre gli altri, meno

rilevanti, ne verranno esclusi. Considerata l'esperienza maturata durante lo sviluppo del software si è ritenuto che per un dataset di malware di circa diecimila unità una soglia pari a un cinquantesimo delle richieste dns totali possa essere ideale. Ovviamente per dataset di dimensioni diverse starà all'analista stabilire il valore della soglia anche esaminando l'output del tool esaminato nella sezione precedente.

Ciò di cui necessita lo script per l'esecuzione consiste fondamentalmente negli output dei due programmi fin qui presentati, cioè un file csv contenente l'elenco dei checksum md5 dei malware e il relativo nome ed un file csv contenente sempre il checksum e il dominio di secondo livello oggetto della query.

3.3.1 Relazioni tra il dataset ed i domini

Il primo punto in cui si intende indagare in questo ambito è verificare se una famiglia in particolare di tutte quelle individuate all'interno del dataset abbia la particolarità di interagire con uno specifico dominio.

Le informazioni che abbiamo raccolto fino a questo punto consentono di incrociare le stringhe che identificano i malware individuati con le queries da essi effettuate. Il risultato di tale esame viene rappresentato su un grafo riportante i domini, i nomi dei malware ed il numero di malware appartenenti ad una determinata famiglia che effettuano queries verso i domini di nostro interesse. In questo modo risulta facile individuare la presenza di eventuali isole, che indicheranno quindi una forte relazione tra un malware ed un determinato dominio od insieme di essi.

Un'altra osservazione possibile è la somiglianza tra diverse famiglie. Per somiglianza intendiamo un comportamento simile nell'interazione con i domini esaminati, come può accadere nel caso di malware con due denominazioni di-

verse ma con richieste dns rivolte verso i medesimi domini. Tale circostanza si sarebbe osservata in modo massiccio nel caso in cui i nomi dei malware non fossero stati preventivamente ripuliti dalle informazioni riguardante i singoli ceppi, tuttavia l'osservazione di questo fenomeno è ancora particolarmente frequente specialmente in quei casi in cui il malware viene identificato con il nome di un'euristica.

Originariamente si era pensato di includere nel grafo anche un nodo riportante la denominazione "others", che avrebbe indicato l'interazione di una famiglia con un dominio non incluso in questa fase dello studio. Varie osservazioni hanno tuttavia portato a stabilire che tale caratteristica avrebbe in modo negativo impattato sulla leggibilità del grafo non introducendo tuttavia informazioni significative ed è quindi stata rimossa.

La direzione di ricerca successiva è complementare a quella appena illustrata e si propone di partire dai domini per individuare le famiglie di worm che ad essi si collegano. Per far ciò si è pensato di procedere alla creazione di un grafico a torta per dominio esaminato. Tale grafico riporta le percentuali di famiglie di malware che hanno il dominio in questione come oggetto della richiesta dns.

Ciò che si vuole cercare di capire in questo momento è se il confronto tra grafici di domini differenti possa rivelare comportamenti analoghi. Se un determinato grafico riporta percentuali molto simili a quelli di un altro grafico potrebbe essere il primo indizio che i due server in oggetto abbiano effettivamente un qualche tipo di relazione. Tale situazione potrebbe venire a crearsi nel momento in cui due domini venissero ad esempio utilizzati per garantire ridondanza al collegamento C&C verso il malware, oppure banalmente i domini potrebbero reindirizzare verso il medesimo server. Un'altra possibilità verosimile è che uno di questi domini venga utilizzato per il controllo del canale di comunicazione verso il bot, mentre l'altro dominio potrebbe essere l'hosting di

un qualche malware che l'attaccante cerca di scaricare ed installare nella macchina obiettivo. Un'eventualità di questo tipo verrà esaminata meglio in seguito, quando si individuerà il tipo di malware ospitato dai domini in esame.

3.3.2 Relazioni fra i domini

Ci addentriamo ora nello studio delle relazioni tra i domini che abbiamo sin qui ottenuto. Per far ciò si rivela interessante la verifica della natura delle dns queries effettuate dai malware oggetto dello studio.

Il primo grafico generato in questo caso è un grafico a torta riportante il numero di query effettuato da ogni malware. Ciò che stiamo cercando è verificare se esista un numero consistente di malware che effettuano query verso più di un dominio. Qualora questo numero dovesse rivelarsi importante sarà infatti possibile studiare tali malware per determinare la presenza ricorrente di specifici domini all'interno delle richieste dns dei malware.

La verifica della presenza di domini ricorrenti è un classico caso di associative mining e viene quindi affrontato con l'algoritmo FP-growth. In questo caso è necessario dimensionare correttamente il supporto minimo per gli itemsets individuati. Nel caso specifico, come nel caso della soglia impostata per la selezione dei domini, un buon compromesso per una quantità di circa ventimila queries è pari a un centesimo delle richieste totali. Ancora una volta, tale numero è strettamente dipendente dalla natura del dataset in fase di studio e dovrebbe essere impostato di volta in volta dall'analista a seconda delle proprie esigenze.

Gli itemsets frequenti possono rivelare informazioni estremamente molto interessanti. Se il confronto tra i pie-chart riportante le percentuali di malware per dominio poteva portare indizi verso le probabile relazione tra domini differenti, in questo momento tali indizi potrebbero diventare delle certezze. Se il

supporto minimo è ben dimensionato si possono infatti in questo momento individuare gruppi di domini che compaiono frequentemente all'interno della lista di queries effettuate dai malware; questo lascia supporre che dietro ai server di tali domini possa celarsi lo stesso individuo od organizzazione, o quantomeno che l'attaccante necessiti di interagire con questi server per portare a termine un attacco.

Una volta stabilite queste somiglianze è giunto il momento di verificare effettivamente se i domini fin qui individuati condividano qualche server. Per far ciò si procede alla ricerca di server e nameserver per ogni dominio. Alle informazioni storiche rilevate in questa fase si aggiunge lo stato corrente del server, ottenendo così per i domini di interesse una lista di IP.

Gli IP così ottenuti vengono quindi confrontati e si cerca di individuare se esistano server in comune tra più domini. Il risultato di tale ricerca viene presentato su un grafo che consente di evidenziare i domini che condividono almeno un server. Come risultato in questo caso siamo quindi in grado di provare che dietro determinati domini si cela lo stesso individuo od organizzazione.

Ovviamente è molto probabile che i risultati ottenuti in questa fase non facciano altro che confermare indizi già emersi precedentemente riguardo le connessioni rilevate tra i diversi domini.

3.3.3 Altre minacce ospitate dai domini notevoli

In ultima analisi si procede all'individuazione delle minacce ospitate dai domini in esame. Fin qui l'esame ha riguardato il malware del dataset di partenza, quello che si vuol fare invece ora è verificare se i domini ottenuti dallo studio di tali malware siano host per altri tipi di minacce o meno.

Attraverso le WAPI è possibile accedere, per ogni IP di nostro interesse, ad

una lista di minacce rilevate sul server accompagnate da una rapida descrizione. L'idea in questo caso è di controllare l'esistenza di tali minacce sulla lista di server in nostro possesso, allo scopo di identificare a quale tipo di attacco può essere esposta la vittima dell'aggressione. Per far ciò per ogni dominio viene creato un grafico a torta riportante il tipo di minaccia ospitata.

Le categorie più frequenti in questo caso indicano la presenza di virus, browser exploit, rogue AV e minacce generiche alla navigazione. I risultati possono essere interpretati in base alle informazioni trovate precedentemente per capire quando si tratti di server C&C malevoli, quando di server destinati ad ospitare software malevolo oppure quando il dominio possa risultare soltanto come vittima di un attacco da parte del pc infetto.

Capitolo 4

Case study

Obiettivo di questo capitolo è lo studio e l'analisi di un dataset di malware la cui caratteristica comune è il fatto di utilizzare il canale IRC per la comunicazione tra attaccante e computer infetto. Per l'analisi sono state utilizzate le API sviluppate dal progetto Wombat, che offrono la possibilità di interrogare un ampio numero di risorse contenenti informazioni su malware, domini e server che li ospitano.

Il dataset di partenza consta di circa 14000 malware, di cui circa 3000 sappiamo utilizzare un packer. Si ricorda che un packer altro non è che uno stumento utilizzato per la compressione del file eseguibile utilizzato nel campo dei malware per prevenire il riconoscimento da parte dei software antivirus.

Innanzitutto si è proceduto all'identificazione di tali file attraverso il checksum md5. In questo caso si è verificato se l'md5 del malware in questione sia noto ai più comuni antivirus. Nello specifico per uniformità si è scelto di utilizzare la nomenclatura offerta da Symantec. Dei 14000 file oggetto dell'analisi ne sono stati selezionati 8910 (vedremo in seguito il motivo), generando riscontri su 5843 malware conosciuti, distribuiti come mostrato nell'illustrazione 4.1.

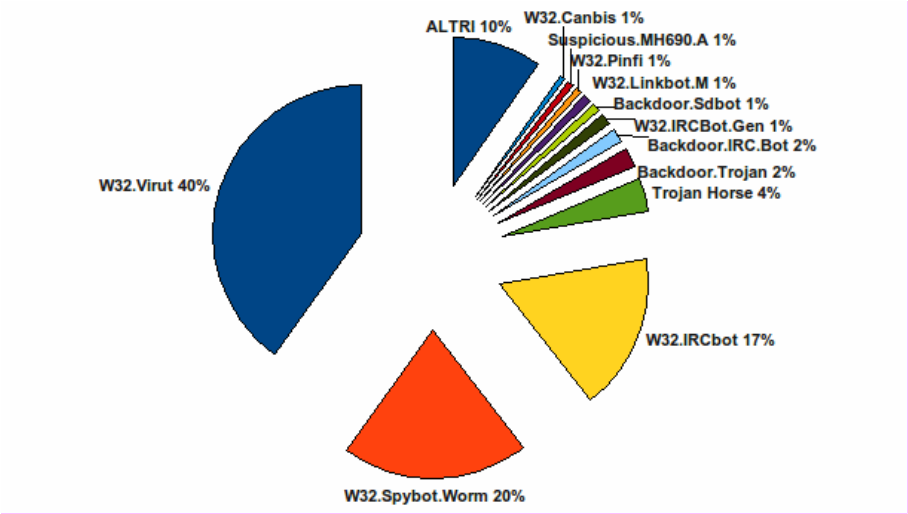

Figura 4.1: Malware riconosciuti da VirusTotal

Possiamo notare come circa il 40% dei malware è identificato come appartenente alla famiglia dei W32.Virut [27], un worm che infetta principalmente file eseguibili e si diffonde attraverso drive fissi, removibili e network. Una preponderanza così massiccia di una famiglia specifica di worm merita un approfondimento: possiamo così notare nell'illustrazione 4.2 in che percentuale siano presenti i diversi ceppi del malware. A seguire si individua un 20% di W32.Spybot.worm, che altro non è che un'euristica per il riconoscimento di una famiglia di worm che si diffonde attraverso Kazaa e mIRC. Discorso analogo riguarda W32.IRCbot, un'euristica per worm che si diffondono attraverso il canale IRC.

Abbiamo quindi confermato che il fattore comune di questi malware è l'utilizzo di un canale IRC. Nello specifico osserviamo che i worm individuati funzionano in modo pressoché analogo. Sostanzialmente il malware, una volta penetrato in un computer, apre un canale di comunicazione con un server IRC. L'attaccante si collega al medesimo server e invia i comandi al canale IRC che li inoltra alla macchina infetta, potendola controllare anche senza stabilire una

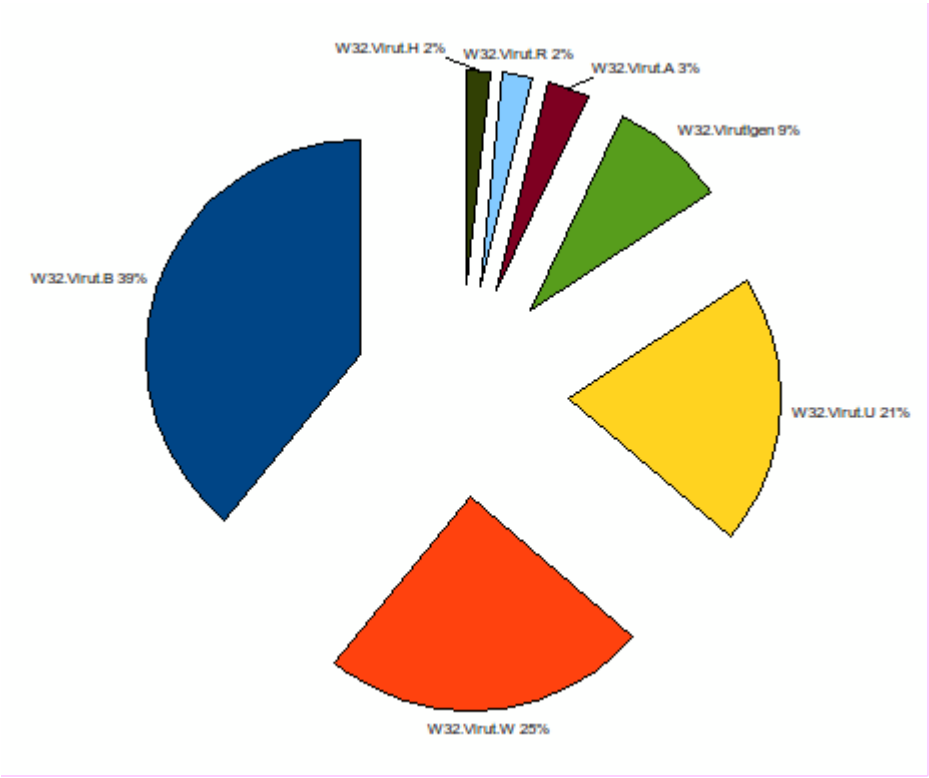

Figura 4.2: Ceppi della famiglia Virut presenti nel dataset

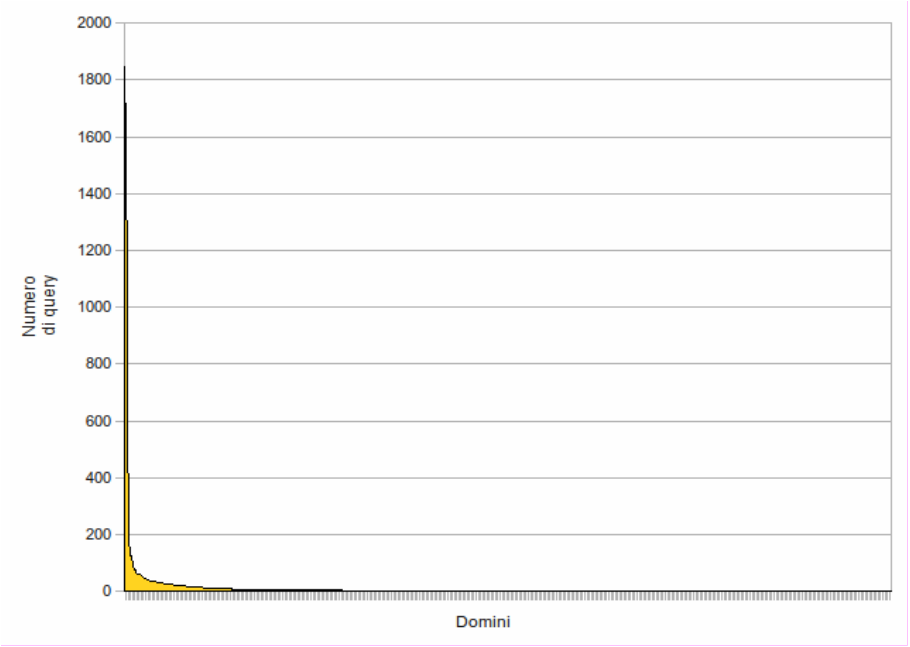

Figura 4.3: Distribuzione delle query dns

connessione diretta.

Risulta quindi evidente che per l'analisi del dataset a nostra disposizione potrebbe rivelarsi intelligente partire dall'individuazione di quei server IRC a cui si connettono i malware. Per far questo occorre partire dall'esame delle query dns effettuate dai worm analizzati. I 14000 file generano un totale di 19788 query dns che, scremate le bad requests, si riducono a 13066 query effettuate da 8910 malware. Questa osservazione ci porta a ridurre a 8910 i file in esame. Le query analizzate rimandano ad un totale di 1055 domini e sono distribuite come mostrato nell'illustrazione 4.3.

Dai dati risulta che l'81% delle richieste DNS dei malware ha come oggetto il 10% dei domini. Tale risultato ci porta a pensare che effettivamente ci sia una elevata correlazione tra i malware in esame.

Passiamo quindi all'analisi dei domini. L'analisi è resa difficoltosa dalla scar-

sità di informazioni che le WAPI sono in grado di fornire. Dati quali l'indirizzo email del registrant, la data di registrazione o il registrar sono spesso omesse, non rendendo così possibile un'analisi accurata. Procederemo quindi in due modi differenti. Prima prenderemo in esame i domini ancora attivi, per i quali sono disponibili numerose informazioni, poi ci addentreremo nell'analisi più specifica di alcuni domini notevoli, caratterizzati da una grande quantità di richieste DNS ma di cui disponiamo meno informazioni.

4.1 I domini attivi

Dei 1055 domini a cui siamo arrivati ve ne sono effettivamente attivi ad oggi 585, i cui server sono così distribuiti:

1. Stati Uniti 380 IPs
2. Germania 46
3. Olanda 28
4. Francia 16
5. Gran Bretagna 13
6. Giappone 13
7. Russia 10
8. Ucraina 10
9. Canada 5
10. Romania 5

11. Altri 59

Per quanto riguarda gli AS di appartenenza dei server abbiamo una situazione di questo tipo:

1. PAH-INC - GoDaddy.com, Inc. 39 IPs
2. DYNDNS - Dynamic Network Services, Inc. 38
3. ENOMAS1 - eNom, Incorporated 32
4. OVERSEE-DOT-NET – Oversee.net 29
5. THEPLANET-AS - ThePlanet.com Internet Services, Inc. 23
6. XS4ALL-NL XS4ALL 21
7. CASTLE-ACCESS - Castle Access Inc 16
8. NOIP-VITAL - Vitalwerks Internet Solutions, LLC 16
9. ONEANDONE-AS 1&1 Internet AG 14
10. OVH OVH 11
11. Altri 321

A differenza della situazione che ci si presentava con la localizzazione geografica dei server in questo caso non abbiamo una maggioranza netta, per ora ci limitiamo ad osservare l'elevata incidenza di servizi di IP dinamico. La mappa nell'illustrazione 4.4 mostra la dislocazione geografica dei server individuati.

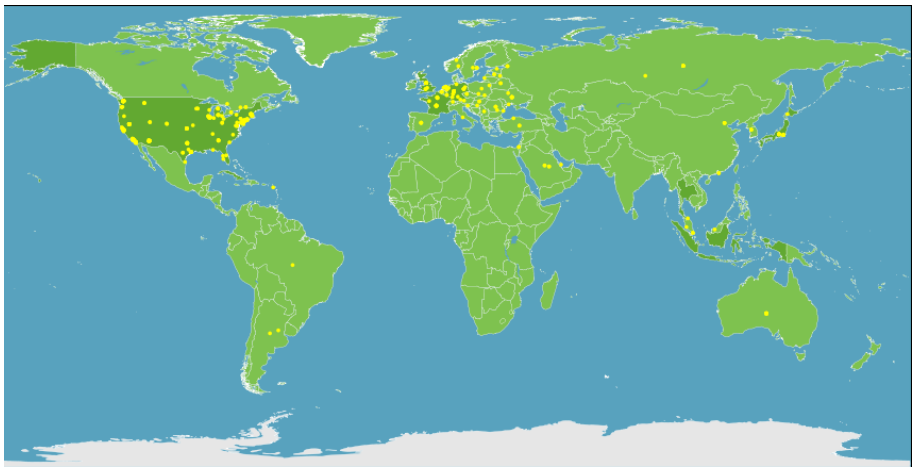

Figura 4.4: Dislocazione geografica dei server

Osservando la distribuzione dei server relativi ai domini da noi presi in considerazione ed i relativi server DNS, notiamo un elevato grado di correlazione, come evidente dall'illustrazione 4.5. Per questo motivo possiamo ipotizzare che i server IRC "malevoli" non facciano largo utilizzo di botnet come infrastruttura di hosting. In tal caso si sarebbe infatti osservata una distribuzione più omogenea dei server sullo spazio IP, in quanto chiunque può creare un proprio server IRC, mentre i server DNS sarebbero rimasti concentrati in ristretti range di IP.

Forth ci fornisce informazioni sulla natura dei server che hostano i domini fin qui analizzati, consentendoci di capire quali di questi siano CCserver riconosciuti, quali spammer e quali siano domini attaccanti noti. Nello specifico rileviamo che 76 dei 560 server in esame hanno indirizzi IP conosciuti come attaccanti, 74 sono C&C server riconosciuti e 18 sono attivi come spammer.

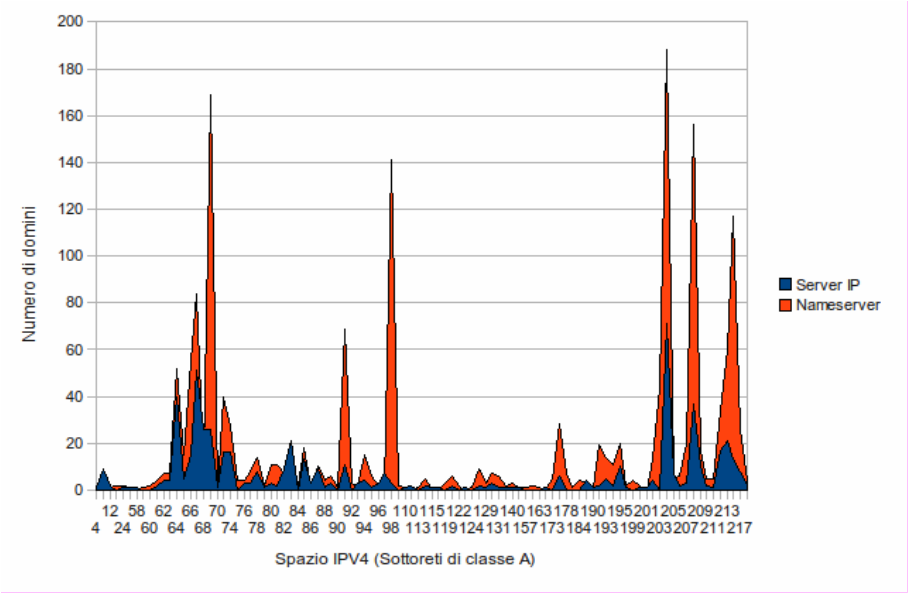

Figura 4.5: Distribuzione dei server sullo spazio IP

4.2 I domini principali

Come precedentemente affermato l'analisi fin qui effettuata fa riferimento a domini e server tuttora attivi, tuttavia vale la pena dare un'occhiata più da vicino a 6 particolari domini, che da soli coprono il 53 % delle query DNS rilevate.

I domini in oggetto sono:

1. ircgalaxy.pl 1847 query
2. ntkrnlpa.info 1589 query
3. installstorm.com 1321 query
4. ghura.pl 1291 query
5. zief.pl 471 query
6. mgts.by 359 query

Figura 4.6: Server in comune tra 2 domini notevoli

per un totale di 6878 richieste DNS.

Per i sopracitati domini le WAPI dispongono di una considerevole quantità di informazioni, che consentono un'analisi più approfondita. Per prima cosa pare interessante verificare se questi domini siano effettivamente in qualche modo collegati fra loro, ed in effetti dall'illustrazione 4.6 si evince come i domini "ircgalaxy.pl" e "zief.pl" abbiamo più server in comune. I due domini hanno inoltre lo stesso registrar e, seppure le date di creazione risultino differenti, al momento dell'analisi l'ultima modifica riportata dal servizio whois differisce solo di alcuni secondi da un sito all'altro.

Questi indizi ci portano a pensare che dietro entrambi i domini vi sia lo

stesso individuo od organizzazione.

In base alle informazioni raccolte sui server possiamo dire degli altri domini. Possiamo notare alcune somiglianze, ad esempio l'host oggetto delle query rivolte a "ircgalaxy.pl" è "proxim.ircgalaxy.pl", analogo comportamento è riscontrabile con l'host "proxim.ntkrnlpa.info". Tuttavia non sono individuabili prove concrete di una correlazione fra i due domini, quindi possiamo continuare a trattarli come 2 domini ben distinti.

Un'osservazione importante si individua però quando si cercano maggiori informazioni su questi domini. Si nota infatti che i record MX [28] dei domini ghura.pl e ircgalaxy.pl puntano allo stesso ip, segno che entrambi i domini condividono il mail server. In più l'esame dei server su cui sono hostati questi domini mostra 2 server con indirizzi ip attigui. Possiamo dire a questo punto che ci sono forti indizi che ci portano a pensare che i tre domini con suffisso "pl" appartengano allo stesso individuo od organizzazione. Un'analisi che potrebbe rivelarsi interessante è verificare se alcuni dei malware in esame si colleghino a uno o più di questi domini contemporaneamente. Ebbene, dati alla mano questo risulta vero in particolar modo per i domini "ircgalaxy.pl", "installstorm.com" e "ghura.pl". Bel 1272 degli 8910 malware presi in considerazione in totale effettua una query dns per tutti e soli questi 3 i domini. Tale comportamento porta a pensare che in effetti una relazione tra questi 3 server IRC deve necessariamente esistere. Tale comportamento è quindi la causa principale dell'elevato numero di query dns rispetto al numero dei file: solo per i malware che effettuano connessioni verso questi 3 domini abbiamo 2544 richieste in più. Sempre nello stesso ambito notiamo che i malware che effettuano richieste dns verso più di un dominio sono 2115, i restanti file effettuano una sola richiesta. Si veda l'illustrazione 4.7.

Come detto, il 76% dei malware effettua una sola connessione, il 17 % ne

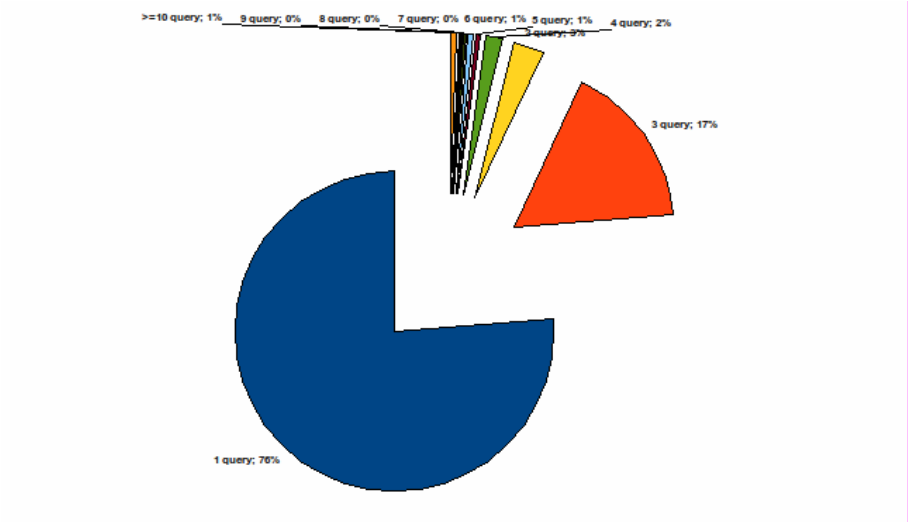

Figura 4.7: Numero di query effettuate da ogni malware

effettua 3 (su 1490 malware che effettuano 3 connessioni 1272 sono quelle di cui si discute qualche riga sopra) ed un ristretto 3% solamente 2. Il restante 4 % è caratterizzato da malware che effettuano più di 3 connessioni. In 60 casi notiamo che il numero di query dns effettuate supera le 10 unità.

Sappiamo quindi a grandi linee quale sia il comportamento di questi malware. Quello che possiamo cercare di scoprire adesso è se vi sia una qualche relazione tra i ceppi dei worm che abbiamo analizzato e i domini ai quali si collegano. La serie di grafici 4.8 a seguire chiarisce meglio la situazione.

Si nota ad esempio che per il dominio “ntkrnlpa.info” abbiamo una fortissima presenza di W32.Virut.W, la restante parte è composta da euristiche che non escludono l’appartenenza dei file alla classe Virut. Come ci si poteva attendere i grafici di “ircgalaxy.pl”, “installstorm.com” e “ghura.pl” si somigliano molto, se infatti di “installstorm.com” e “ghura.pl” si può dire che siano praticamente uguali, “ircgalaxy.pl” differisce per la presenza di un 20% di W32.Virut.A. Di “zief.pl” non si può dire molto. I malware vengono riconosciuti

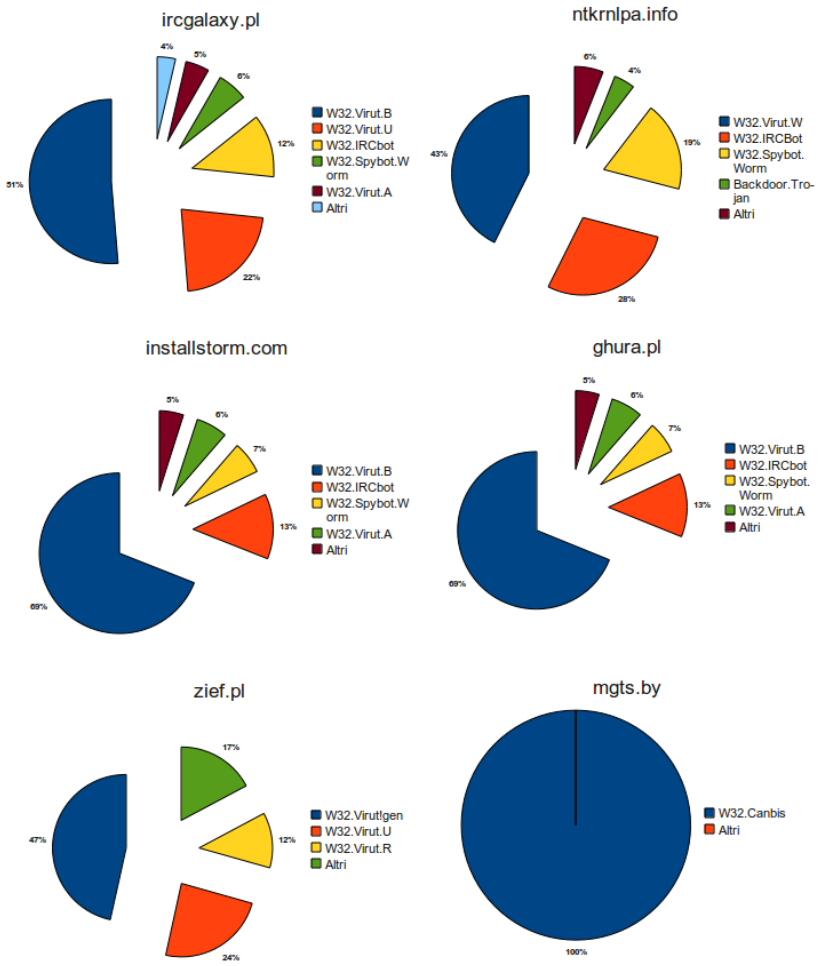

Figura 4.8: Malware per dominio

come appartenenti alla famiglia Virut ma non si evidenzia un ceppo dominante.

"mgts.by" rappresenta invece un caso differente. Dei 359 malware che tentano il collegamento a questo dominio ne vengono riconosciuti solamente 41, tutti del tipo W32.Canbis. Il basso rate di riconoscimento potrebbe essere dovuto ad una diffusione recente del malware. Ricordiamo infatti che la metodologia applicata non prevede una scansione in real time del file, bensì una ricerca nell'archivio fornito dalle WAPI per l'md5 del file in esame.

Estendendo l'analisi siamo in grado di indagare più a fondo la natura di questi domini. Le informazioni sui server su cui sono ospitati tali domini ci consentono infatti di individuare quali altri threat siano collegati a questi server. L'indagine è possibile sui domini "ircgalaxy.pl", "installstorm.com", "zief.pl" e "ghura.pl". Degli altri domini il dataset utilizzato non fornisce sufficienti informazioni.

Il totale dei threat rilevati è 19066, di questi 18715 sono riconducibili a server che hostano il dominio "zief.pl", 193 "irgalaxy.pl", 151 "installstorm.com" e 7 "ghura.pl". I grafici a seguire mostrano le percentuali di threat relative ai singoli domini. La situazione di "ircgalaxy.pl" evidenzia 71 minacce classificate come generico rischio alla navigazione, 49 chiaramente identificate come malware e 41 browser exploit. In misura minore troviamo worm e rogue antivirus.

Meno chiaro è invece lo stato dei server di "installstorm.com": si nota che delle 150 minacce rilevate ben 100 vengono identificate come "unknown, statistical detection", 40 come virus e 10 come browser exploit. Cercando maggiori informazioni sulle minacce classificate come "unknown" si osserva che si tratta di file richiesti dal malware Virut.

Per quanto riguarda "zief.pl" si nota un'enorme quantità di minacce generiche, ben 17802 su un totale 18715. Nello specifico si rileva che tali minacce generiche sono il larga parte browser exploit e virus. A causa dell'enorme quan-

tità di threat generici il grafico a seguire non evidenzia a sufficienza la presenza di ben 394 minacce classificate come rogue antivirus, i 214 virus e i 173 browser exploit. Delle minacce totali relative a questo domini ben 17897 fanno riferimento ad un solo server.

Dell'ultimo dominio, "ghura.pl", non si hanno a disposizione molte informazioni. I threats individuabili sono solamente 7, di cui 6 identificati come generici. Tali minacce generiche si rivelano "crawler end message" e fake scanner, dato compatibile con l'indicazione del settimo threat che è riportato come un rogue antivirus.

Capitolo 5

Conclusioni e sviluppi futuri

5.1 Conclusioni

Il lavoro qui presentato descrive l'implementazione di una metodologia per l'analisi di dataset di malware orientati alla creazione di bot. La tecnica si rivela sufficientemente flessibile da poter essere applicabile con determinati accorgimenti a generici malware che prevedano l'interazione con risorse residenti su internet. Il lavoro partee dallo studio delle dns queries effettuate dai campioni presenti nel dataset ed è mirato a rivelare l'infrastruttura di controllo dietro alle botnet create dai malware esaminati. Le 3 fasi presentate corrispondono agli altrettanti tool che sono stati sviluppati per consentire l'applicazione della metodologia a reali casi di studio.

La validazione degli strumenti sviluppati in questo lavoro di tesi è stata eseguita applicando la metodologia ad un dataset composto da malware basati su IRC. I tool hanno consentito di individuare domini e server responsabili dell'infrastruttura di controllo dei bot esaminati, evidenziando inoltre delle connessioni tra domini privi di legami evidenti.

L'analisi presentata non può prescindere da un'attenta supervisione da parte di un operatore umano, tuttavia il livello di automazione raggiunto consente di ridurne al minimo le interazioni. Lo sviluppo di un tool che eviti l'utilizzo manuale delle WAPI permette inoltre una maggiore immediatezza nel reperimento dei risultati desiderati, sfruttando anche il fatto che il software sia stato sviluppato in modo da mitigare alcuni punti deboli delle API ed integrando fra loro informazioni provenienti da più dataset.

5.2 Sviluppi futuri

La validazione del lavoro è stata affrontata tramite l'analisi di un dataset di malware IRC based. Un possibile sviluppo è l'applicazione della metodologia qui presentata per lo studio di un più ampio spettro di dataset di malware, trovandosi così nella condizione di poter confrontare i risultati di diversi dataset utilizzando uno schema comune.

Un eventuale miglioramento per i tool presentati consisterebbe invece nell'implementazione di un'interfaccia grafica per una maggiore immediatezza di utilizzo. Allo stesso modo potrebbe rivelarsi interessante lo sviluppo di un'interfaccia web per la visualizzazione globale dei risultati derivanti dalle analisi condotte sui singoli dataset.

Bibliografia

[1] Worldwide Observatory of Malicious Behaviors and Attack Threats. WOMBAT Project Description. `https://wombat-project.eu/wombat-project-description.html`, 2007

[2] Worldwide Observatory of Malicious Behaviors and Attack Threats. *The 2nd WOMBAT workshop*, settembre 2009.

[3] Wikipedia. Malware. `http://it.wikipedia.org/wiki/Malware`, marzo 2011

[4] Kriss Kendall, Chad McMillan. Pratical Malware Analysys, 2007

[5] Moheeb Abu Rajab, Jay Zarfoss, Fabian Monrose, and Andreas Terzis. A multifaceted approach to understanding the botnet phenomenon. In *Proceedings of the 6th ACM SIGCOMM conference on Internet measurement*, 2006

[6] Computer Economics. Malware report 2005: the impact of malicious code attacks. `http://www.computereconomics.com/article.cfm?id?1090`, 2006

[7] Jack Clark. Rustock botnet sends 39 percent of all spam. `http://www.zdnet.com/news/rustock-botnet-sends-39-percent-of-all-spam/460146`, agosto 2010

[8] Reuters. Stuxnet rattled Iran but atom work goes on: report. `http://www.reuters.com/article/2011/02/16/us-nuclear-iran-stuxnet-idUSTRE71F1X720110216`, febbraio 2011

[9] GData. Protezione dei propri dati. `http://www.gdata.it/security-labs/informazioni/suggerimenti-e-indicazioni/protezione-dei-propri-dati.html`, marzo 2011

[10] VMware. Virtualization Basics. `http://www.vmware.com/virtualization/virtual-machine.html`, marzo 2011

[11] VMware. Understanding snapshots. `http://www.vmware.com/support/ws55/doc/ws_preserve_sshot_understanding.html`, marzo 2011

[12] Wikipedia. Debugger. `http://en.wikipedia.org/wiki/Debugger`, marzo 2011

[13] Wikipedia. Static program analysis. `http://en.wikipedia.org/wiki/Static_program_analysis`, marzo 2011

[14] Ulrich Bayer, Andreas Moser, Christopher Kruegel, and Engin Kirda. Dynamic Analysis of Malicious Code. *Journal in Computer Virology, Springer Computer Science*, maggio 2006

[15] Wikipedia. Runtime packer. `http://en.wikipedia.org/wiki/Runtime_packer`, marzo 2011

[16] Microsoft. Guida alla difesa antivirus a più livelli. `http://technet.microsoft.com/it-it/library/dd536184.aspx`, maggio 2004

[17] Wikipedia. Dynamic program analysis. `http://en.wikipedia.org/wiki/Dynamic_program_analysis`, marzo 2011

[18] United States Computer Emergency Readyness Team. Understanding Denial of Service attacks. `http://www.us-cert.gov/cas/tips/ST04-015.html`, novembre 2009

[19] Mary Landesman. What is a keylogger trojan? `http://antivirus.about.com/od/whatisavirus/a/keylogger.htm`, marzo 2011

[20] Wikipedia. Internet Relay Chat. `http://en.wikipedia.org/wiki/Internet_Relay_Chat`, marzo 2011

[21] David Caraballo, Joseph Lo. The IRC prelude. `http://www.irchelp.org/irchelp/new2irc.html`, gennaio 2000

[22] Wikipedia. Dynamic Dns. `http://en.wikipedia.org/wiki/Dynamic_DNS`, marzo 2011

[23] Brett Stone-Gross, Marco Cova, Lorenzo Cavallaro, Bob Gilbert, Martin Szydlowski, Richard Kemmerer, Christopher Kruegel, and Giovanni Vigna. Your botnet is my botnet: analysis of a botnet takeover. *Proceedings of the 16th ACM conference on Computer and communications security (CCS '09)*, 2009

[24] L. Daigle. WHOIS Protocol Specification. `http://tools.ietf.org/html/rfc3912`, settembre 2004

[25] VirusTotal. About VirustTotal. `http://www.virustotal.com/about.html`, marzo 2011

[26] Jianwei Zhuge, Thorsten Holz, Xinhui Han, Jinpeng Guo, and Wei Zou. Characterizing the IRC-based botnet phenomenon. *Peking University & University of Mannheim Technical Report*, 2007.

[27] Symantec. W32.Virut. `http://www.symantec.com/security_response/writeup.jsp?docid=2007-041117-2623-99`, marzo 2010

[28] Wikipedia. MX record. `http://en.wikipedia.org/wiki/MX_record`, marzo 2011