Verfolgen und Abschwächen von *Malicious Remote Control Networks*

Thorsten Holz

Arbeitsgruppe "Embedded Malware"
Ruhr-Universität Bochum
thorsten.holz@rub.de

Abstract: *Malicious Remote Control Networks* sind Netze von kompromittierten Maschinen, die unter der Kontrolle eines Angreifers stehen und von diesem für rechtswidrige Aktionen benutzt werden (z. B. Versenden von Spam-Nachrichten oder Stehlen von privaten Informationen). Die bekannteste Instanz von diesen Netzen sind *Botnetze*.

Im Rahmen dieser Arbeit wurde eine generische Methodik entwickelt, um gegen die Grundursache dieser Netze, nämlich den Kommunikationskanal zwischen dem Angreifer und den infizierten Maschinen, vorzugehen und diese Netze zu stoppen. Dazu wurden verschiedene Techniken und Tools entwickelt, um *Malicious Remote Control Networks* im Detail zu studieren. In empirischen Studien im Internet wurde die praktische Umsetzbarkeit der entwickelte Methodik erfolgreich demonstriert.

1 Einführung

Angriffe gegen Nutzer sind einer der negativen Seiteneffekte im heutigen Internet. Das Ziel eines Angreifers ist es typischerweise, die Maschine des Opfers zu kompromittieren und Kontrolle über diese zu erlangen. Die Maschine wird dann beispielsweise dazu benutzt, so genannte *Distributed Denial-of-Service* Angriffe (DDoS Angriffe) durchzuführen, Spam-Nachrichten zu verschicken oder vertrauliche Daten zu stehlen. Die Möglichkeiten eines Angreifers sind fast unbeschränkt, da er die komplette Kontrolle über das Opfersystem hat. Aus der Sicht eines Angreifers sind solche Angriffe noch effizienter, wenn er es schafft, eine große Anzahl an Maschinen gleichzeitig zu kompromittieren. Um alle diese Maschinen kontrollieren zu können, setzt der Angreifer ein *Malicious Remote Control Network* auf, das heisst einen Mechanismus, der es dem Angreifer erlaubt, effizient eine große Anzahl an kompromittierten Maschinen zu kontrollieren, um diese für rechtswidrige Aktionen zu benutzen. Die bekannteste Art dieser Netze sind *Botnetze* [CJM05], es existieren aber noch weitere Arten dieser Netze, die im Rahmen der Dissertation studiert wurden.

Weil diese Netze einer der Hauptfaktoren derzeitiger Missbräuche im Internet sind benötigen wir neuartige Ansätze, um sie in einem automatisierten und effizienten Prozess stoppen zu können. In dieser Arbeit konzentrieren wir uns auf dieses offene Problem und stellen eine allgemeine Methodik vor, um die Grundursache hinter *Malicious Remote Control Networks* zu stoppen. Dazu studieren wir die Grundprinzipien hinter diesen Netzen und entwickeln basierend darauf eine generische Vorgehensweise, um gegen diese Netze

vorzugehen. In empirischen Untersuchungen konnten wir zeigen, dass die vorgeschlagene Methodik allgemein ist und auf verschiedene Arten von *Malicious Remote Control Networks* angewandt werden kann. Des Weiteren wurden im Rahmen der Dissertation einige neuartige Techniken und Tools zum automatisierten "Sammeln" und zur automatisierten Analyse von Schadsoftware entwickelt. Im Folgenden wird ein kurzer Überblick über die Kernbeiträge der Dissertation gegeben; ausführliche Informationen zu diesem Themenkomplex sowie detalliertere Einblicke sind in der Dissertation selbst verfügbar [Hol09].

2 Technische Grundlagen

Im Folgenden geben wir einen kurzen Überblick zu den grundlegenden Themengebieten *Honeypots/Honeynets* und *Botnetze*, die im Rahmen der Arbeit behandelt wurden. In beiden Bereichen wurden technische Grundlagen erarbeitet, neuartige Tools und Techniken entwickelt sowie empirische Untersuchungen im Internet durchgeführt.

Honeypots und Honeynets. Der englische Begriff *Honeypot* bezeichnet für gewöhnlich einen Gegenstand, von dem eine gewisse Attraktivität ausgeht, die bestimmte, nicht nur tierische, Interessenten anzulocken vermag [Lon01]. Sie eignen sich demnach als Köder, um Aufmerksamkeit auf einen Gegenstand zu lenken. Dieses Prinzip der Köderung kann auch im Bereich der IT-Sicherheit angewendet werden: Hier werden *elektronische Köder* ausgelegt, um das Verhalten von Angreifern leichter zu studieren. Elektronische Köder sind Netzwerkressourcen (bspw. Computer, Router oder Switches), deren Wert darin besteht, angegriffen und kompromittiert zu werden [Spi02, PH07]. Diese *Honeypots* haben keine spezielle Aufgabe im Netzwerk, sind aber ansonsten nicht von regulären Komponenten zu unterscheiden und dienen als Lockmittel für Angreifer. Sie sind mit spezieller Software ausgestattet, welche die anschließende Forensik eines Angriffs deutlich erleichtert. Im Gegensatz zu einer herkömmlichen forensischen Untersuchung erlauben beispielsweise gezielte Veränderungen im Betriebssystem das direkte Mitschneiden aller Aktivitäten eines Angreifers. Durch die Vielfalt der so gewonnenen Daten kann man schneller und genauer dessen Angriffswege, Motive und Methoden erforschen.

Der Honeypot-Ansatz ist sowohl aus praktischer als auch aus theoretischer Sicht interessant für den Bereich der IT-Sicherheit. Aus praktischer Sicht erlauben detaillierte Informationen über das Verhalten von böswilligen "Crackern" (den so genannten *Black-Hats*), die Abwehrmaßnahmen in unterschiedlichen Umgebungen effizienter und effektiver zu gestalten. Effizienter werden Abwehrmaßnahmen dadurch, dass man sich auf relevante Angriffe konzentrieren kann (Netzwerke von Honeypots, so genannte *Honeynets*, können hierzu empirische Beiträge liefern). Effektiver werden Abwehrmaßnahmen, indem mit Hilfe der Honeypots neue Schwachstellen und Verwundbarkeiten entdeckt und schnell analysiert werden können. Aus theoretischer Sicht bilden Honeypots eine interessante Instanz des *dual use*-Prinzips der IT-Sicherheit: Man bekämpft Angreifer mit ihren eigenen Methoden. Die Fähigkeiten und die Werkzeuge, die ein Verteidiger anwendet, unterscheiden sich kaum von denen der Angreifer.

Im Rahmen der Arbeit wurden einige Techniken und Tools im Bereich Honeypots entwickelt, mit denen Schadsoftware, die sich automatisiert in einem Netz verbreitet (zum Beispiel Computer-Würmer oder Bots), studiert werden kann [BKH+06, ZHH+07, GHW07].

Botnetze. Ein *Botnetz* ist ein Netz aus kompromittierten Maschinen, die unter der Kontrolle eines Angreifers stehen: Der Angreifer kann Befehle zu den einzelnen Bots senden und diese führen diese Kommandos aus. Botnetze sind damit eine Instanz von *Malicious Remote Control Networks* und die momentan am weitesten verbreitete Art dieser Netze.

Abbildung 1 zeigt den schematischen Aufbau eines Botnetzes. Der Angreifer kommuniziert über den so genannten *Command and Control* (C&C) Server mit den kompromittierten Maschinen: Die Bots verbinden sich zu dem C&C-Server, über den sie die Kommandos des Angreifers empfangen. Bei den ersten Generationen von Botnetzen verlief die Kommunikation über das *Internet Relay Chat* (IRC) Protokoll. Heutzutage ist dies schätzungsweise noch bei 40-50 Prozent der Fall. Immer beliebter bei Angreifern wird die Kommunikation über HTTP ohne persistente Verbindung zum C&C-Server: Die Bots kontaktieren periodisch den C&C Server und fragen neue Befehle an. Des Weiteren existieren Botnetze mit proprietären Kommunikationsprotokollen, bei denen der Angreifer ein eigenes Kommunikationsformat entwickelt hat.

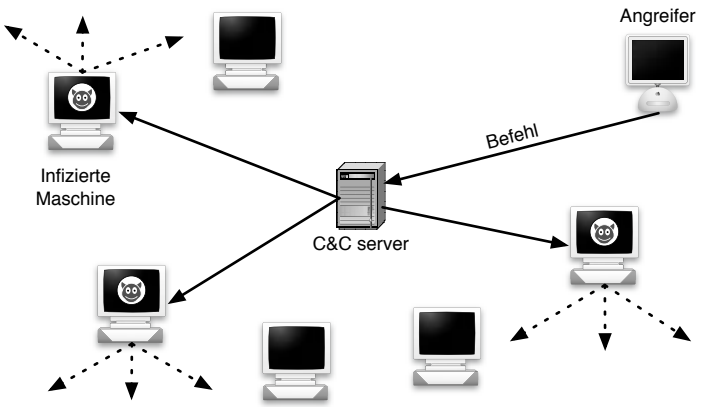

Abbildung 1: Schematischer Überblick zu Botnetzen mit zentralem C&C Server.

Neben Botnetzen mit einem zentralen C&C Server existieren Botnetze, die ein Peer-to-Peer Protokoll als Kommunikationsmechanismus implementieren: Die einzelnen Bots werden als Teil der Kommunikationsinfrastruktur benutzt und deshalb ist es deutlich schwieriger, diese Art von Botnetzen zu studieren und dagegen vorzugehen.

Botnetze werden von Angreifern häufig zum Versenden von Spam-Nachrichten, zur Durchführung von *Distributed Denial-of-Service* (DDoS) Angriffen oder zum Stehlen von Information von den kompromittierten Maschinen benutzt. Somit sind Botnetze eine der Wurzeln von Onlinekriminalität und bieten eine Plattform für diese Art von Verbrechen. Im Rahmen der Arbeit wurden Botnetze systematisch studiert [FHW05, GHW07, HSD+08] und es wurden effektive Erkennungsmethoden entwickelt [GH07].

3 Methodik

Die Grundidee unserer Methodik zur Abwehr von *Malicious Remote Control Networks* besteht aus drei Schritten und macht sich die grundlegenden Eigenschaften dieser Netze zu Nutze [FHW05, HSD+08]: Die Grundursache des Problems besteht in dem Kommunikationskanal zwischen dem Angreifer und den kompromittierten Systemen. Wir versuchen, diesen Kanal automatisiert zu entdecken und zu stören bzw. ihn komplett abzuschalten.

Dazu benutzen wir im ersten Schritt Honeypots, um gegenwärtige Angriffe im Internet zu erforschen und wir können beispielsweise Kopien von Schadsoftware (engl.: *Malicious Software*, kurz *malware*), die sich selbständig verbreitet, in einem automatisierten Prozess sammeln. Wir analysieren die gesammelten Daten in einem automatisierten Verfahren und extrahieren Informationen über den Fernsteuerungsmechanismus. Dazu wurden Techniken aus dem Bereich der Programmanalyse weiterentwickelt, um den C&C Server zu finden, mit dessen Hilfe Kommandos zu den infizierten Maschinen gesendet werden. Im zweiten Schritt benutzen wir die extrahierten Informationen und infiltrieren das *Malicious Remote Control Network*. Dies kann beispielsweise umgesetzt werden, indem wir das Verhalten eines Bots simulieren und uns in den Fernsteuerungskanal einschleusen. Letztendlich benutzen wir im dritten Schritt die in der Infiltrierungsphase gesammelten Informationen, um das Netz abzuschwächen. Dies geschieht beispielsweise, indem der Kommunikationskanal geschlossen wird, so dass der Angreifer keine Befehle mehr zu den kompromittierten Maschinen senden kann. Das Abschalten des Kommunikationskanals kann nicht vollständig automatisiert werden, aber unser Verfahren liefert wertvolle Informationen, die als Indizien für die Ermittlungen in diesem Bereich dienen können. Daneben können die Informationen zur Erkennung von infizierten Maschinen benutzt werden.

Im Rahmen der Arbeit demonstrieren wir die praktische Umsetzbarkeit der vorgeschlagenen Methodik. Insbesondere zeigen wir, dass die Infiltrierung der Netze möglich ist und relevante Informationen zum Netz gesammelt werden können. Wir untersuchen verschiedene Typen von *Malicious Remote Control Networks* und erörtern, wie wir alle auf automatisierte Art und Weise aufspüren können. Als erstes Beispiel untersuchen wir in Abschnitt 4 Botnetze mit einem zentralen C&C-Server: Wir erläutern, wie die vorgeschlagenen drei Schritte in der Praxis umgesetzt werden können und präsentieren empirische Messergebnisse, die im Internet gesammelt wurden. Des Weiteren stellen wir in diesem Abschnitt kurz vor, wie Botnetze mit einem Peer-to-Peer-basierten Kommunikationskanal studiert werden können: Eine Abschwächung dieser Botnetze ist schwieriger, da kein zentraler C&C-Server existiert, der abgeschaltet werden kann. Dennoch kann unsere Methodik auch auf diese Art von Netzen angewandt werden und wir stellen empirische Messergebnisse vor, die unsere Methode untermauern.

Als dritten Fall analysieren wir in Abschnitt 5 *Fast-Flux Service Networks* (FFSNs). Die Idee hinter diesen Netzen ist, dass der Angreifer die kompromittierten Maschinen nicht direkt missbraucht. Stattdessen benutzt er sie dazu, ein Proxy-Netzwerk mit Hilfe dieser Maschinen aufzubauen, das dann eine robuste Hostinginfrastruktur ermöglicht. Unsere Methodik kann auch auf diese neuartige Art von *Malicious Remote Control Networks* angewandt werden und wir präsentieren empirische Resultate und Ergebnisse, die diese Hypothese bestätigen.

4 Verfolgen und Abschwächen von Botnetzen

Als erste Fallstudie untersuchen wir Botnetze und zeigen, dass die im vorherigen Abschnitt kurz vorgestellte Methodik auf diese Art von Netzen angewendet werden kann.

Automatisiertes Sammeln von Schadsoftware. Wir benutzen das Konzept hinter elektronischen Ködern, um automatisiert neue Instanzen von Schadsoftware zu sammeln. Dazu wurde im Laufe der Dissertation verschiedene Arten von Honeypots entwickelt, die auf unterschiedliche Art und Weise ein verwundbares System emulieren [BKH+06, ZHH+07].

Die Grundidee hinter diesen Honeypots ist die *Emulation* von Schwachstellen: Wir emulieren die Teile eines Netzwerkdienstes, die eine Schwachstelle enthalten und als Einfallstor während eines Angriffs dienen. Wird der Honeypot nun angegriffen kann der Angreifer nicht unterscheiden, ob er ein reales System oder einen Honeypot angreift. Als Resultat können wir automatisiert neue Instanzen von Schadsoftware "sammeln"; an einem typischen Tag können dies Hunderte neue Instanzen sein [GHW07].

Automatisierte Analyse von Schadsoftware. Eine manuelle Analyse der gesammelten Binärdateien ist kaum möglich, insbesondere da die Analyse einer einzelnen Datei Tage oder sogar Wochen dauern kann. Deshalb wurden im Rahmen der Dissertation Techniken und Tools entwickelt, um eine *automatisierte* Analyse von Schadsoftware durchführen zu können. Der Schwerpunkt der Forschung lag auf der *dynamischen* Analyse von Programmen: Das zu untersuchende Programm wird in einer instrumentierten Umgebung ausgeführt und zur Laufzeit wird das Verhalten der Datei beobachtet [WHF07]. Dadurch können beispielsweise Änderungen am Dateisystem oder an der Registry beobachtet sowie der Netzwerkverkehr aufgezeichnet werden.

Nach dem Abschluss der dynamischen Analyse wird ein Verhaltensreport erstellt, der das beobachtete Verhalten zusammenfasst. Im Fall der Analyse eines Bots enthält ein solcher Report die Adresse des C&C Servers sowie Informationen über das verwendete Kommunikationsprotokoll, wir erhalten also wertvolle Informationen über das Botnetz.

Automatisierte Infiltrierung von Botnetzen. Nachdem wir im zweiten Schritt die Adresse des C&C Server bestimmt haben, können wir nun das Botnetz infiltrieren: Wir emulieren das Kommunikationsprotokoll eines Bots und verbinden uns zum C&C Server. Nun können wir beispielsweise die aktuellen Befehle des Angreifers beobachten, andere Bots im gleichen Botnetz identifizieren und weitere Information über das Botnetz sammeln.

Die so erhobenen Daten können dazu benutzt werden, das Botnetz abzuschalten: Wir stellen die erfassten Daten *Computer Emergency Response Teams* (CERTs), Hosting-Providern und teilweise auch der Polizei zur Verfügung, damit diese die C&C Server des Botnetzes abschalten beziehungsweise Informationen über die eigentlichen Angreifer sammeln können. Des Weiteren können auf der Basis der von uns gesammelten Daten die Opfer informiert werden, damit diese ihre Maschine von der Schadsoftware säubern.

Empirische Ergebnisse. Die in den vorherigen Abschnitten erläuterten Schritten können sowohl für Botnetze mit zentralem C&C Server als auch für Botnetze mit Peer-to-Peer Kommunikationsmechanismen umgesetzt werden. In empirischen Untersuchungen demonstrierten wir die technische Machbarkeit sowie die Automatisierbarkeit der Methode: Während das Sammeln sowie die Analyse von Schadsoftware komplett automatisierbar ist und entsprechende Tools entwickelt wurden, erfordert die Infiltrierung von Botnetzen einen beschränkten manuellen Aufwand, um den Kontext des Kommunikationsprotokolls zu verstehen und eine Emulierung zu implementieren. Im Rahmen der Arbeit wurde dies für IRC, HTTP, einige proprietäre Protokolle sowie für das Peer-to-Peer Botnetz *Storm Worm* umgesetzt und eine entsprechende Infiltrierung ermöglicht.

In empirischen Experimenten wurden über einen Zeitraum von vier Monaten etwa 900 Botnetze infiltriert und detaillierte Informationen über diese Netze gesammelt. Mehr als 500.000 Bots konnten in dieser Spanne beobachtet werden und dies illustriert die weite Verbreitung dieser Gefahr. Die typische Größe eines Botnetzes lag zwischen einigen hundert Bots bis hin zu mehr als 40.000 Bots. Zusätzlich konnten mehr als 300 DDoS-Angriffe gegen etwa 100 Opfersysteme erfasst werden. Die gesammelten Informationen wurden an CERTs weitergegeben, die dann gegen die Botnetze vorgegangen sind.

Insbesondere die Infiltrierung von *Storm Worm* war ein wichtiger Beitrag der Arbeit. Es konnte gezeigt werden, dass auch effektive Gegenmassnahmen gegen Peer-to-Peer Botnetze möglich sind. Es konnten Instanzen des Bots gesammelt und automatisiert analysiert werden. Basierend auf diesen Informationen wurde dann ein Tool entwickelt, um dieses Botnetz zu infiltrieren. Abbildung 2 zeigt den schematischen Aufbau des Botnetzes. Kompromittierte Maschinen mit lokaler IP-Adresse (bspw. hinter einem DSL-Router) werden zum Versand von Spam-Nachrichten und für DDoS-Angriffe benutzt. Im Gegensatz dazu bilden kompromittierte Maschinen mit öffentlicher IP-Adresse die Kommunikationsinfrastruktur: Diese Maschinen implementieren die *Distributed Hash Table* (DHT) *Kademlia* [MM02]. Der Angreifer selbst betreibt einige statische Backend-Server, über die Befehle an die Bots gesendet werden. Des Weiteren wurden auch aktive Gegenmaßnahmen gegen *Storm Worm* implementiert und es konnte gezeigt werden, dass mit Hilfe dieser Techniken die Kommunikation effektiv gestört werden kann.

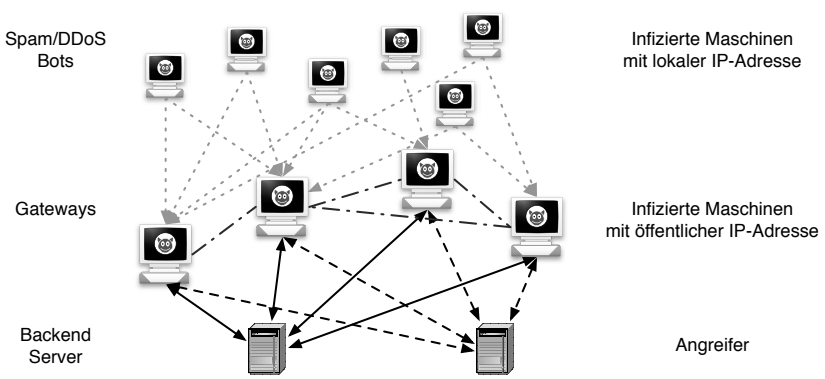

Abbildung 2: Aufbau von *Storm Worm*, ein Peer-to-Peer Botnetz.

5 Verfolgen und Abschwächen von Fast-Flux Service Netzen

Im Rahmen der Arbeit wurde eine weitere Art von *Malicious Remote Control Networks* analysiert und die erste systematische Studie zu *Fast-Flux Service Netzen* (FFSNs) wurde erstellt [HGRF08]. In einem FFSN benutzt ein Angreifer die kompromittierten Maschinen nicht direkt, sondern er baut mit Hilfe dieser Maschinen ein verteiltes Proxy-Netz auf, das dann als robuste Hostinginfrastruktur benutzt werden kann. Dazu benutzt er das Domain Name System (DNS) auf spezielle Art und Weise: Die Zuweisung zwischen einer Domäne und einer IP-Adresse wechselt häufig und eine Domäne weist jeweils auf kompromittierte Maschinen, die eingehende Anfragen an einen zentralen Server weiterleiten.

Abbildung 3 illustriert den Aufbau eines FFSNs. Der Angreifer hat durch die Ausnutzung einer Sicherheitslücke einen *Flux Agent* auf einigen Maschinen installiert. Er kontrolliert die Domäne `thearmynext.info` und weist ihr die IP-Adresse einer kompromittierten Maschine zu. Anfragen an diese Domäne gelangen deshalb zunächst zu einem der *Flux Agents* und werden von dort an das so genannte *Mothership* weitergeleitet und von diesem beantwortet (Schritte 1-4). Der Angreifer wechselt häufig die Zuweisung der Domäne zu den *Flux Agents* und vertuscht so den Aufbau des Netzes. Des Weiteren wird durch diesen Aufbau die IP-Adresse des *Motherships* verschleiert und es ist schwierig, den Standort und die wahre IP-Adresse dieser Maschine zu identifizieren.

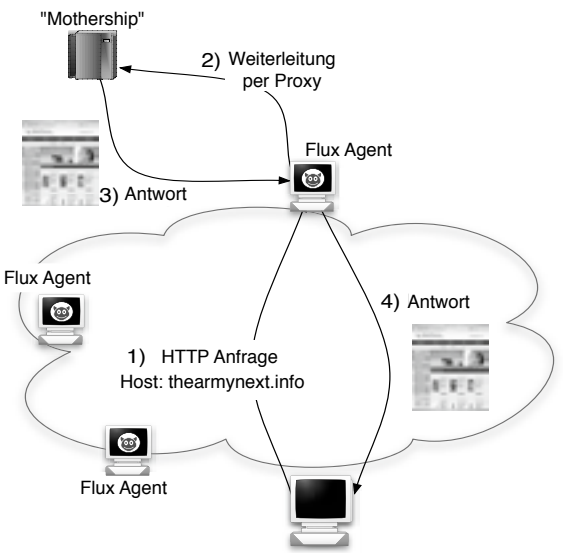

Abbildung 3: Schematischer Aufbau eines *Fast Flux Service Netzes*.

Die in Abschnitt 3 eingeführte Methode kann auch auf FFSNs angewendet werden. Dazu werden in einem ersten Schritt wieder Eintrittspunkte in das FFSN gesucht. Anstatt direkt die *Flux Agents* mit Hilfe von Honeypots zu "sammeln" haben wir einen anderen Ansatz gewählt: Wir beobachten die *Fast Flux* Domänen und können so *indirekt* das FFSN studieren. Durch wiederholte DNS-Lookups können wir die aktuell aktiven *Flux Agents*

feststellen und somit aktive Bots innerhalb des FFSNs identifizieren. Im Gegensatz zu diesem *passiven* Ansatz zur Infiltrierung können wir das FFSN auch *aktiv* infiltrieren, indem wir einen *Flux Agent* auf einem Honeypot ausführen und den Netzwerkverkehr detailliert beobachten. So können weitere Informationen über das FFSN gesammelt und beispielsweise die IP-Adresse des *Motherships* herausgefunden werden.

Automatisiertes Erkennen von Fast-Flux Netzen. Um FFSNs zu studieren wurde zunächst eine Metrik entwickelt, mit deren Hilfe automatisiert entschieden werden kann, ob eine gegebene Domäne die Eigenschaften einer *Fast-Flux* Domäne aufweist. Dazu wurden die grundlegenden Eigenschaften von FFSNs studiert: beispielsweise hat ein Angreifer nur eine ungenaue Kontrolle darüber, in welchem Autonomen System (AS) sich ein *Flux Agent* befindet oder wie lange dieser verfügbar ist. Daraus ergeben sich Einschränkungen für einen Angreifer, die es uns erlauben, eine generische Metrik zu entwickeln, die mittlerweile eine Standardmethode in diesem Gebiet ist.

Basierend auf einem Korpus aus 128 manuell identifizierten *Fast-Flux* Domänen und 5.803 gutartigen Domänen wurde die entwickelte Metrik parametrisiert. Auf Testdaten lieferte die Metrik eine Genauigkeit von 99,98% (Standardabweichung: 0,05%) und somit ist die entwickelte Metrik sehr präzise und kann zur Entdeckung von FFSNs benutzt werden.

Empirische Ergebnisse. Es wurden empirische Untersuchungen zu FFSNs im Internet durchgeführt. Dazu wurden 22.264 Spam-Nachrichten analysiert, die insgesamt 7.389 verschiedene Domänen bewarben. Anhand unserer Metrik konnten 2.197 (29,7%) Domänen als *Fast-Flux* Domäne identifiziert werden und damit konnte gezeigt werden, dass diese Technik häufig von Spammern benutzt wird und Abwehrmaßnahmen nötig sind.

#	ASN	Name des AS und Land	# beobachtete *Flux Agents*
1)	7132	AT&T Internet Services, US	2.677
2)	9304	Hutchison Global, HK	1.797
3)	4766	Korea Telecom, KR	590
4)	3320	Deutsche Telekom, DE	500
5)	8551	Bezeqint Internet, IL	445
6)	12322	Proxad/Free ISP, FR	418
7)	8402	Corbina telecom, RU	397

Tabelle 1: Übersicht zu den sieben häufigsten ASNs, die während der Beobachtung von 33 *Fast-Flux* Domänen über einen Zeitraum von sieben Wochen beobachtet wurden.

Um das Verhalten von FFSNs über einen längeren Zeitraum zu studieren wurden 33 *Fast-Flux* Domänen sieben Wochen lang beobachtet. Alle 300 Sekunden wurde ein DNS-Lookup durchgeführt, um die aktuellen IP-Adressen der aktiven *Flux Agents* festzustellen. Insgesamt wurden in diesem Zeitraum 18.214 verschiedene IP-Adressen aus 818 Autonomen Systemen beobachtet und Tabelle 1 gibt einen Überblick über die Verteilung der *Flux Agents* auf die verschiedenen Autonomen Systeme. Aus der Übersicht wird klar, dass es sich bei FFSNs um ein globales Problem handelt.

6 Zusammenfassung

Im Rahmen der Arbeit wurde eine Methodik entwickelt, um automatisiert gegen *Malicious Remote Control Networks* vorgehen zu können. Dazu wird die Grundursache dieser Netze, nämlich der Kommunikationskanal zwischen einem Angreifer und einer infizierten Maschine, aufgedeckt und dann versucht, diesen effektiv zu stören bzw. komplett abzuschalten. Zur technischen Realisierung der Methodik werden in einem ersten Schritt Instanzen eines Netzes (bspw. Bots oder *Fast-Flux* Domänen) mit Hilfe von Honeypots gesammelt. Diese werden dann automatisiert analysiert, zum Beispiel mit Hilfe von Techniken aus dem Bereich der Programmanalyse. In einem zweiten Schritt wird das Netz dann infiltriert, um "von innen" Informationen über das Netz sammeln zu können. Diese Informationen können dann in einem dritten Schritt dazu benutzt werden, effektiv gegen das Netz vorzugehen, beispielsweise indem die C&C Server abgeschaltet oder die *Fast-Flux* Domänen deaktiviert werden. Die praktische Realisierbarkeit der Methodik wurde umfangreich anhand empirischer Studien demonstriert.

Wir erwarten, dass die in dieser Arbeit vorgeschlagene Methodik auch dazu benutzt werden kann, weitere Arten von *Malicious Remote Control Networks* zu verfolgen und gegen diese Netze vorzugehen. Die entwickelten Tools und Techniken werden bereits von vielen verschiedenen Organisationen benutzt, um diese Art von Netzen zu studieren. Des Weiteren bilden die verschiedenen Honeypots bzw. Ansätze zur automatisierten Analyse von Schadsoftware die Grundbausteine des *Internet Malware Analyse Systems* (InMAS). Dies ist ein im Aufbau befindliches Frühwarnsystem für das Internet, um neue Schadsoftware möglichst früh erkennen zu können.

Weitere Informationen. Innerhalb dieser kurzen Zusammenfassung konnte nur ein grober Überblick zu diesem Themengebiet gegeben werden. Technische Details zu Honeypots, Honeynets und Botnetzen sind ausführlich in einem Buch zu diesem Thema beschrieben [PH07]. Nähere Informationen zu den fachlichen Aspekten sind in der Dissertation selbst [Hol09] bzw. den einzelnen Artikeln verfügbar.

Literatur

[BKH+06] Paul Bächer, Markus Kötter, Thorsten Holz, Felix Freiling und Maximillian Dornseif. The Nepenthes Platform: An Efficient Approach to Collect Malware. In *9th Symposium On Recent Advances in Intrusion Detection (RAID)*, September 2006.

[CJM05] Evan Cooke, Farnam Jahanian und Danny McPherson. The Zombie Roundup: Understanding, Detecting, and Disrupting Botnets. In *Workshop on Steps to Reducing Unwanted Traffic on the Internet (SRUTI)*, Juni 2005.

[FHW05] Felix Freiling, Thorsten Holz und Georg Wicherski. Botnet Tracking: Exploring a Root-Cause Methodology to Prevent Distributed Denial-of-Service Attacks. In *10th European Symposium On Research In Computer Security (ESORICS)*, Juli 2005.

[GH07] Jan Göbel und Thorsten Holz. Rishi: Identify Bot Contaminated Hosts by IRC Nickname Evaluation. In *Hot Topics in Understanding Botnets (HotBots)*, April 2007.

[GHW07] Jan Göbel, Thorsten Holz und Carsten Willems. Measurement and Analysis of Autonomous Spreading Malware in a University Environment. In *4th Conference on Detection of Intrusions & Malware, and Vulnerability Assessment (DIMVA)*, Juli 2007.

[HGRF08] Thorsten Holz, Christian Gorecki, Konrad Rieck und Felix Freiling. Measuring and Detecting Fast-Flux Service Networks. In *15th Network & Distributed System Security Symposium (NDSS)*, Februar 2008.

[Hol09] Thorsten Holz. *Tracking and Mitigation of Malicious Remote Control Networks*. Dissertation, University of Mannheim, April 2009.

[HSD+08] Thorsten Holz, Moritz Steiner, Frederic Dahl, Ernst Biersack und Felix Freiling. Measurements and Mitigation of Peer-to-Peer-based Botnets: A Case Study on Storm Worm. In *USENIX Workshop on Large-Scale Exploits and Emergent Threats (LEET)*, 2008.

[Lon01] Longman Group Ltd. (Hrsg.). *Longman Dictionary of Contemporary English*. Langenscheidt-Longman GmbH, 2001.

[MM02] Petar Maymounkov und David Mazieres. Kademlia: A Peer-to-peer Informatiion System Based on the XOR Metric. In *Workshop on Peer-to-Peer Systems (IPTPS)*, 2002.

[PH07] Niels Provos und Thorsten Holz. *Virtual Honeypots: From Botnet Tracking to Intrusion Detection*. Addison-Wesley, Juli 2007.

[Spi02] Lance Spitzner. *Honeypots: Tracking Hackers*. Addison-Wesley, November 2002.

[WHF07] Carsten Willems, Thorsten Holz und Felix Freiling. CWSandbox: Towards Automated Dynamic Binary Analysis. *IEEE Security and Privacy*, 5(2), März 2007.

[ZHH+07] Jianwei Zhuge, Thorsten Holz, Xinhui Han, Chengyu Song und Wei Zou. Collecting Autonomous Spreading Malware Using High-Interaction Honeypots. In *9th International Conference on Information and Communications Security (ICICS)*, Dezember 2007.

Thorsten Holz wurde am 15. Juni 1981 in Trier geboren. Herr Holz studierte in den Jahren 2000 bis 2005 Informatik mit Nebenfach BWL an der RWTH Aachen und erhielt sein Diplom im September 2005 mit Auszeichnung. Die Diplomarbeit mit dem Titel "New Fields of Application for Honeynets" wurde mit dem zweiten Platz des CAST-Förderpreises IT-Sicherheit ausgezeichnet. Zwischen Oktober 2005 und Juni 2009 war Herr Holz am Lehrstuhl für Praktische Informatik 1 der Universität Mannheim (Prof. Felix Freiling) beschäftigt. Im April 2009 schloss er seine Promotion mit der Note *summa cum laude* ab, das Thema der Dissertation war "Tracking and Mitigation of Malicious Remote Control Networks". Zwischen Juli 2009 und März 2010 forschte Herr Holz an der Technischen Universität Wien als Projektassistent und im April 2010 wurde er von der Ruhr-Universität Bochum zum Juniorprofessor für das Themengebiet "Embedded Malware" ernannt.

Der Forschungsschwerpunkt von Herrn Holz liegt im Bereich Systemsicherheit, insbesondere den Aspekten Honeypots/Honeynets, Analyse von Schadsoftware und Botnetze. Die Ergebnisse der Forschung wurden in mehr als 30 begutachteten Artikeln bei internationalen Workshops und Konferenzen publiziert und Herr Holz wirkte im Rahmen seiner Arbeit in mehr als 20 Programmkomitees mit.