

DDOS-AS-A-SERVICE

Investigating Booter Websites



José Jair Cardoso de Santanna

DDoS-AS-A-SERVICE

INVESTIGATING BOOTER WEBSITES

José Jair Cardoso de Santanna

Graduation Committee:

Chairman: prof. dr. P.M.G. Apers
Supervisor: prof. dr. ir. A. Pras
Co-supervisor: prof. dr. L. Z. Granville
Co-supervisor: dr. R. de O. Schmidt

Members:

prof. dr. O. Festor, TELECOM Nancy - University of Lorraine, France
prof. dr. J. Schönwälder, Jacobs University, Germany
prof. dr. M.J.G. van Eeten, Delft University of Technology, The Netherlands
prof. dr. ir. L.J.M. Nieuwenhuis, University of Twente, The Netherlands
prof. dr. ir. B.R.H.M. Haverkort, University of Twente, The Netherlands

Funding Sources:

Flamingo Network of Excellence (EU FP7 318488)
Distributed Denial-of-Service Defense—D3 (NWO 628.001.018)



CTIT Ph.D. thesis Series N°. 17-448
Centre for Telematics and Information Technology
P.O. Box 217, 7500 AE
Enschede, the Netherlands

ISSN 1381-3617
ISBN 978-90-365-4429-0
DOI <https://doi.org/10.3990/1.9789036544290>

Cover design by Davi Souza.

Type set with L^AT_EX. Printed by IPSKAMP.



Copyright ©2017 José Jair Cardoso de Santanna
This work is licensed under a Creative Commons
Attribution-NonCommercial-ShareAlike 3.0 Unported License.
<http://creativecommons.org/licenses/by-nc-sa/3.0/>

DDoS-AS-A-SERVICE

INVESTIGATING Booter WEBSITES

DISSERTATION

to obtain
the degree of doctor at the University of Twente,
on the authority of the rector magnificus,
prof. dr. T.T.M. Palstra,
on account of the decision of the graduation committee,
to be publicly defended
on the 17th of November 2017 at 14:45

by

José Jair Cardoso de Santanna

born on the 11th October 1987
in Belém, Brazil

This dissertation has been approved by:

prof. dr. ir. A. Pras (supervisor)
prof. dr. L.Z. Granville (co-supervisor)
dr. R. de O. Schmidt (co-supervisor)

on the 17th August 2017

Acknowledgments

First of all, I thank God, “someone” that I’ve learned to love, believe, and fear, observing His daily acts in my life in a mysterious way. I thank Him that taught me that cultivating my dreams with hard work, hope, and patience, followed by watering from my sweat and tears, I would harvest with joy (and a lot of joy). I thank Him for (somehow) helping me to stand when I was broken and lonely, for always making me feel I am special, as in the story that little David faced the giant Goliath (the Bible, Book 1 Samuel, Chapter 17).

I am thankful to my best friend, the present and future love of my life, Priscillinha. She is the “little parrot” that has decided to fly with this one crazy parrot (*i.e.*, myself). Thank you for supporting me even when you did not understand what was going on. Thank you for respecting our differences and discuss them with such care and love. Thank you for being always there, even when sleeping. I love loving you.

I am also extremely thankful to my determined mother, Fátima. Dear mother, I always listen and respect what you told me, even while complaining (and sometimes I complained quite a lot). Note that even your most recurring advise was rigorously respected: “study comes first!” So, here I am, my sweet mother! And now, am I allowed to start building a family of my own? Actually, I already know the answer, of course, and I am thankful for your blessing.

I also thank my sister, Lorena, who once taught me that each person has their own time and pace. You are the one that better understands and protects me, even one-ocean far away. Also thank you for bringing to my life the best brother-in-law out there. I would like to extend my gratitude to my other siblings, João and Angélica, who taught me that flying far from home is possible, and also to Eliana (Jupirinha), who is not blood of my blood but is just as if it was.

I would like to thank my four families: the Cardoso’s, the Santanna’s, and also the van Duren’s, and the Hul’s. Thank you for your support and examples that were set to me. Although you are completely different among each other, we all have the same values of loving and protecting our families. Soon we will be all legally linked to each other. I am also thankful to my great friends, of course, who care for me: Rafa, Sanjka, Sofia, Luinha, Anuj, Orsi, Ricardo, Pedro, Renata, Muleque, Cecília, Tiago, Líga, Andrea, and Niko. Thanks for

making me feel at home wherever we are! A special thanks goes to few friends from Belém, Brazil, who even far never let me without prayers: Diogo, Élida, João, Camila, Marcelo, and Ádria. Thank you very much!

During the last four years (2013–2017), I've made so many friends around the world! In this category, I need to acknowledge my three flatmates (Niko, Martin, and Andrea) who taught me how to share a flat and parts of my life. And there is Anja, Victor, and Alex that I had the pleasure to share great experiences traveling together. Of course I would not forget all the (party) people I met in Enschede and all over the world, being part of enjoyable moments that helped me relaxing a bit from my academic life.

Of course special thanks to my supervisors. I thank Aiko for the long talks about life and a lot of great lessons given and taken. Thank you for allowing me to try, fail and succeed, growing me stronger every time. Aiko, I really admire you! Lisandro, I also admire you because of your hard work and objectivity. You really inspire me! My gratefulness to my friend Ricardo. Thanks for being more than I supervisor, for letting me learn from your experiences. I am proud to be the first Ph.D. student in your curriculum. You did great! I also would like to thank Jeanette for “supervising me”, as well as the entire group, as a great mother. We are very lucky to have you caring for us.

I thank my working colleagues: Anna, Björn, Rick, Mattijs, Mozhdeh, Morteza, Luuk, Wouter, Roland, Bernd, Stysia, Jessica, Hamed, and my Brazilian predecessors Ricardo, Rafael, Giovane, Idilio and Tiago for preparing me for the long discussions about thesis with my promoter. Special thanks to Björn for the blessed words during our short walks. I also thank the great young minds that I had a pleasure to somehow support on their researches: Jochum, Dirk, Jarmo, Max, Mark, Roeland, Wouter, Justyna, Joey, Calvin, Kareem, and many others like Jan Harm and Romain Durban whom I had the pleasure to work with. You are one of the main reasons why I love the academic career. Seeing you flying high makes me super proud! In the same way that I supported students, some people supported and were essential for my work to be what it is. I would like to thank the guys from SURFnet (Roland, Xander and Wim), the guys from NBIP (Gerald and Pim), Daphné and David Douglas.

I would like to conclude my acknowledgements thanking my father, Santanna, my true hero! He is the man with the biggest heart that I ever met. Although a terrible partner, you are a great father (at least for me, “teu velho Sabidú e Mirurica”). Thanks for quitting football just to see me grow strong. You made it! Your kids are all grown-ups, happy and united. This was one of your last goals, wasn't it? Dear father, even when your memory fades away, my memory will keep you alive. In the last moments, I will be there making you love the unknown, in the same way that you teach me to love it. Then, when the moment comes, go in peace! I will always love you.

Agradecimentos

Primeiramente, eu agradeço a Deus, quem eu aprendi a acreditar, temer, amar e observar atuando misteriosamente na minha vida. Eu agradeço a Ele que me ensinou que plantando com trabalho duro, esperança e paciência, e regando com o meu suor e lágrimas, eu colheria com alegria (muita alegria). Eu agradeço a Ele que (de alguma forma) me ajuda a permanecer em pé quando eu me sinto em cacos e sozinho. Obrigado por me fazer sentir especial, como na estória do pequeno Davi contra o gigante Golias (Bíblia, livro 1 de Samuel, capítulo 17).

Eu agradeço à minha melhor amiga e amor, meu presente e futuro, Priscillinha. Ela que é o “pequeno papagaio” que decidiu voar junto comigo. Obrigado por me apoiar mesmo sem entender. Obrigado por respeitar as nossas diferenças e discuti-las com zelo e amor. Obrigado por estar sempre comigo (mesmo enquanto dormia). Eu amo te amar.

Eu agradeço à minha forte mãe, Fátima. Amada mãe, eu sempre te escutei e te respeitei (mesmo quando eu reclamei [bastante]). Perceba que até o seu mais frequente conselho foi seguido: “O estudo vem em primeiro lugar”. Aqui estou eu, minha doce mãe! Agora estou liberado para construir minha própria família? (Eu já sei a resposta. Obrigado pela sua bênção).

Eu agradeço à minha doce irmã, Lorena, que me ensinou que cada pessoa tem o seu próprio tempo. Ela que é “a minha cópia da pérola” que “permaneceu por mais tempo dentro da concha”. Você é a pessoa que mais me entende e protege (mesmo com um oceano de distância). Obrigado por decidir me dar o melhor cunhado que eu poderia ter. Em seguida, eu agradeço aos meus irmãos João e Angélica que me ensinaram com atitude que sair para longe de casa era possível. E não posso deixar de agradecer à Eliana (Jupirinha), que não é meu sangue, mas eu sinto como se fosse. Obrigado por apoiar o João e a Angélica e estender o seu carinho para mim.

Obrigado às minhas quatro famílias: os Cardosos, os Santannas, e também os van Duren e os Huls. Obrigado pelo apoio e exemplo de vocês. Embora sejam completamente diferentes, nós temos os mesmos valores: amar e proteger as nossas famílias. Em breve, todos estaremos legalmente conectados. Obrigado também aos meus grandes amigos: Rafa, Sanjka, Luinha, Anuj, Orsi, Ricardo, Pedro, Renata, Muleque, Cecília, Tiago, Lígia, Andrea e Niko. Obrigado por fazerem eu me sentir em casa em qualquer lugar que seja! Também tenho

um agradecimento a poucos amigos (de Belém), que mesmo longe nunca me deixaram sem orações: Diogo, Élida, João, Camila, Marcelo e Ádria. Deus abençoe vocês.

Nesses quatro anos (2013–2017), eu fiz muitos amigos em todos os lugares do mundo. Aqui eu preciso agradecer aos meus três colegas de apartamento: Niko, Martin e Andrea. Obrigado por me ensinarem a compartilhar a minha casa e partes da minha vida. Em seguida, tem a Anja, o Victor e o Alex, com quem eu tive um prazer em dividir inesquecíveis experiências enquanto viajamos juntos.

Agora vem os agradecimentos aos meus orientadores. Eu agradeço ao Aiko pelas longas conversas sobre a vida e pelas grandes lições dadas. Obrigado por me deixar tentar, falhar e me manter forte lutando. Eu admiro você! Eu also admiro o Lisandro pelo seu trabalho duro e foco. Você me inspira. Então vem um agradecimento para o meu amigo Ricardo. Obrigado por ser mais que um orientador. Obrigado por me permitir aprender com as suas experiências. Eu me sinto lisonjeado em ser o primeiro doutorando do seu currículo. Você é foda! Nessa categoria, eu gostaria de agradecer ainda à Jeanette por me “orientar” e ser uma mãe para todos os colegas do grupo de trabalho.

Eu também agradeço aos meus colegas de trabalho: Anna, Björn, Rick, Mattijs, Mozhdeh, Morteza, Luuk, Wouter, Roland, Bernd, Stysia, Jessica, Hammed e aos meus predecessores brasileiros Ricardo, Rafael, Giovane, Idílio e Tiago, por me prepararem para as longas discussões sobre a tese com o meu orientador (Aiko). Um agradecimento especial ao Björn pelas abençoadas palavras durante nossas caminhadas. Eu também agradeço às brilhantes e jovens mentes que eu tive o prazer de apoiar (de alguma forma) nas suas pesquisas: Romain, Jochum, Dirk, Jarmo, Max, Mark, Roeland, Wouter, Justyna, Joey, Calvin e Kareem. Mais que agradecimento, eu gostaria de pedir desculpas se eu não dei tudo o que vocês precisavam, no momento em que vocês precisavam. Vocês são um dos principais motivos de eu amar a vida acadêmica. Ver vocês voando alto me deixa muito orgulhoso! Aqui uma menção honrosa para todos os estudantes que estavam em uma das minhas aulas e se tornaram tão entusiasmados quanto eu. Obrigado pela atenção, sorrisos e apoio de vocês.

Meu último agradecimento vai para meu amado pai, Santanna, meu sempre herói! Ele é o homem com o maior coração que eu conheço. Embora seja um terrível parceiro, ele é um grande pai (pelo menos para mim, teu velho Sabidú e Mirurica). Obrigado por parar de jogar futebol para me ver crescer. Você conseguiu! Seus filhos estão grandes, felizes e unidos. Não era esse o seu último objetivo? Meu amado pai, mesmo quando a sua memória desaparecer, minha memória lhe manterá vivo. E nos últimos momentos, eu estarei lá com você lhe fazendo amar o desconhecido, da mesma forma que você me ensinou a amar. Então, quando for chegado o momento, vá em paz! Eu sempre lhe amarei.

Abstract

Do you like to have Internet connectivity and the millions of services accessible via the Internet? Whether you like it or not, the fact is that our society relies on Internet connectivity for all sort of activities (from shopping to entertainment, from controlling critical infrastructures to allowing the management of social and health records). Distributed Denial of Service (DDoS) attacks are the main threat to the availability of these millions of Internet services. DDoS attacks are intentional acts in which attackers orchestrate devices distributed over the Internet, with the aim of overloading the memory, the processor or the network link of a target system.

Why should you care about DDoS attacks? If your Internet home connection would be the target of a DDoS attack, then not only your connectivity is gone, but also your telephone and TV programs. This is because many homes have triple-play-service (a package offered by Internet providers that includes TV programs and telephone service together with the Internet connectivity). Looking from a company perspective, in 2015, small and medium companies reported spending more than \$US50,000 recovering from a DDoS attack, while large corporations reported an average \$US410,000. This figure increased drastically in 2017: large corporations reported \$US2.5M in revenue loss as a consequence of a DDoS attack. Given the rapid increase observed above, we can expect that these costs will continue to rise, just as our society's increased dependence on networked services.

DDoS attacks first appeared in the late 1990's, and there are more than 35K academic papers indexed by Google Scholar that address the DDoS attack problem. Although this seems to imply that the problem is a well-studied one, DDoS attacks are still in continuous (and alarming) growth. In this thesis, we take a novel approach to address this problem. Instead of limiting our focus on improving the detection and mitigation of dozens of different DDoS attack types, we also focus on investigating the people and organizations involved in attacks. Our goal with this thesis is to understand the technical and non-technical characteristics of DDoS attacks to support further mitigation actions.

The research in this thesis was mainly possible because we observed (around 2013) the change in how and who performs DDoS attacks. Until 2013, DDoS attacks were something that only a (relatively) skilled hacker could perform,

and that required specialist knowledge. In 2013, however, things changed. The hacker community began offering DDoS attacks via Websites easily findable via the most popular searching engines (Google and Bing). Websites called “booters” and “stressers” offer, for very affordable prices, for example, starting from less than \$US5, to perform as many DDoS attacks as requested for a month period. Booters removed the need to have technical skills to perform attacks and fulfill a demand of teenagers that learned to buy DDoS attacks to get personal advantage. For example, teenagers attacked their schools using booters to prevent having online exams, for weeks. Teenagers also started using booters to win online games by attacking the home connection of their opponents.

Booter attacks were not only used by teenagers but also by their owners. This is when a booter unleashes their actual power. For example, over Christmas 2015, the owners of a booter called “Lizardstresser” used their own infrastructure to attack Microsoft and Sony, making these companies completely unreachable for hours. There is also the attack record in 2016 against the DNS company Dyn (using the Mirai botnet), which also involved a booter owner (who released the code of the botnet Mirai). In addition to those very powerful attacks, between 2014 and 2017 booters were considered by network security companies to be responsible for the majority of DDoS attacks worldwide. Both, the increase in attack power and frequency makes the investigation in this thesis even more critical and timely.

The main contributions of this thesis are that we show: (1) how to find booters, (2) how to detect their clients accessing and using them, (3) the characteristics of their attacks, (4) what third-party companies are used by them to maintain their operations, (5) which booters are the most dangerous and (6) which ethical arguments can be used to support mitigation actions against them. Finally, while the core of this thesis is based on scientific publications, its impact does not stop there. A number of solutions proposed in this thesis are actively deployed by network operators worldwide. In addition to this, the methodologies in this thesis are used by the Dutch High Tech Crime Unit for collecting evidences for prosecution cases.

Samenvatting

Wil ook jij Internet connectiviteit en toegang tot miljoenen diensten via het Internet? Of je het nu wel of niet leuk vindt, feit is dat onze samenleving niet meer kan functioneren als de Internet toegang tot diverse diensten zou wegvallen (van winkelen tot entertainment, van het controleren van kritieke infrastructuren tot het beheren van sociale en medische administratie). Distributed Denial of Service (DDoS) aanvallen zijn de belangrijkste bedreiging voor wat betreft de beschikbaarheid van deze diensten. DDoS aanvallen zijn intentionele handelingen waarbij aanvallers gebruik maken van diverse systemen die over het Internet op een gedistribueerde manier samenwerken, met als doel het geheugen, de processor of de netwerkverbinding van een doelsysteem te overbelasten.

Waarom zou je om DDoS aanvallen moeten bekomen? Als jouw Internetverbinding thuis het doelwit is van een DDoS aanval, dan is niet alleen jouw Internetverbinding verdwenen, maar ook jouw telefoon en TV programma's. Dit komt omdat veel huishoudens geabonneerd zijn op zogeheten triple-play-services, een dienstenpakket waarbij Internet aanbieders TV programma's, telefoondiensten en Internet toegang in combinatie aanbieden. In 2015 heeft een gemiddelde MKB organisatie meer dan \$US50.000 moeten uitgeven om de gevolgen van een DDoS aanval ongedaan te maken; bij grote bedrijven was de gemiddelde schade \$410.000. Deze getallen zijn sindsdien dramatisch toegenomen; in 2017 melden grote bedrijven al een omzetverlies van \$US2,5M per DDoS aanval. Gegeven deze snelle toename is de verwachting dat deze kosten verder zullen stijgen, net zoals onze afhankelijkheid van netwerkdiensten in onze maatschappij.

DDoS aanvallen bestaan al sinds het eind van de jaren 1990. Er zijn via Google Scholar meer dan 35 duizend wetenschappelijke artikelen te vinden over dit onderwerp. Alhoewel dit lijkt te impliceer dat het DDoS probleem goed in kaart is gebracht, groeit het aantal DDoS aanvallen nog steeds op een alarmerende wijze. In dit proefschrift kiezen wij daarom voor een andere benadering. In plaats van ons te beperken tot het verbeteren van technieken om DDoS aanvallen te detecteren en te bestrijden, kijken we in dit proefschrift ook naar de mensen en organisaties die direct of indirect bij deze aanvallen betrokken zijn. Het doel van dit proefschrift is de technische en niet-technische eigenschappen van DDoS aanvallen beter te begrijpen, zodat

effectievere maatregelen tegen dergelijke aanvallen mogelijk worden.

De motivatie om dit onderzoek te beginnen komt uit 2013, toen we tijdens discussies met SURFnet niet alleen veranderingen zagen in de manier waarop DDoS aanvallen werden uitgevoerd, maar ook ontdekten wie verantwoordelijk waren voor dergelijke aanvallen. Tot 2013 konden DDoS aanvallen alleen door hackers met specialistische kennis worden uitgevoerd. In 2013 veranderde er iets. Hackers begonnen de techniek om DDoS aanvallen te verrichten via websites beschikbaar te stellen aan derden. Deze websites, die eenvoudig via zoekmachines zoals Google of Bing te vinden zijn, heten “booters” of “stressers”. Voor een beperkt bedrag, vaak al vanaf \$US5, is het mogelijk om een maand lang net zoveel aanvallen uit te voeren als gewenst. Dankzij booters is het niet langer nodig om technische vaardigheden te hebben om een DDoS aanval uit te voeren. Rond 2013 begonnen dan ook tieners booters te gebruiken om tijdens online games de thuisverbinding van tegenspelers aan te vallen, met het doel eenvoudig te winnen. Vervolgens werden ook scholen aangevallen, waardoor wekenlang online examens onmogelijk werden.

Booter aanvallen worden niet alleen geïnitieerd door tieners, maar ook door booter eigenaren. Op dergelijke momenten tonen booters hun ware kracht. Tijdens de Kerst 2015 hebben de bezitters van de booter met de naam “Lizardstresser” bijvoorbeeld hun eigen infrastructuur gebruikt om Microsoft en Sony aan te vallen; een aanval waardoor deze bedrijven voor vele uren onbereikbaar werden. Een ander voorbeeld is de aanval in 2016 tegen het DNS bedrijf Dyn, waarvoor het Mirai botnet is gebruikt. Naast deze voorbeelden worden booters tussen 2014 en 2017 verantwoordelijk gehouden voor het merendeel van de DDoS aanvallen wereldwijd. De toename in aanvalskracht en frequentie zorgen ervoor dat het onderzoek dat in dit proefschrift is beschreven uiterst belangrijk en actueel is.

De belangrijkste bijdragen van dit proefschrift zijn dat we beschrijven: (1) hoe booters gevonden kunnen worden, (2) hoe onderzocht kan worden wie deze booters gebruikt, (3) wat de karakteristieken zijn van booter aanvallen, (4) welke partijen direct of indirect de werking van booters ondersteunen, (5) welke booters het meest gevaarlijk zijn, (6) welke ethische argumenten spelen tijdens het nemen van maatregelen tegen booters. Ten slotte moet worden opgemerkt dat, alhoewel de kern van dit proefschrift gebaseerd is op wetenschappelijke publicaties, de impact daar niet stopt. Een aantal methoden die in dit proefschrift zijn beschreven worden inmiddels wereldwijd actief gebruikt door netwerkbeheerders. Daarnaast worden delen van dit onderzoek gebruikt door het Team High Tech Crime van de Nederlandse Politie, voor het verzamelen van bewijs.

Contents

| | | |
|----------|--|-----------|
| 1 | Introduction | 3 |
| 1.1 | The Internet and DDoS Attacks | 3 |
| 1.2 | DDoS Attack Evolution | 4 |
| 1.3 | Goal and Research Questions | 9 |
| 1.4 | Approach and Thesis Organization | 10 |
| 2 | Finding Booters and Detecting Their Clients | 17 |
| 2.1 | Motivation and Challenges | 18 |
| 2.2 | Crawler: Listing Suspect Booters URLs | 20 |
| 2.3 | Scrapper: Collecting URL Information | 23 |
| 2.4 | Classifier: Determining Booter Websites | 27 |
| 2.5 | Booter List Usage | 37 |
| 2.6 | Concluding Remarks | 42 |
| 3 | Characterizing Clients Usage of Booters | 47 |
| 3.1 | Client Records in Booter Databases | 48 |
| 3.2 | Methodology and Our Database Schema | 49 |
| 3.3 | Aspects to be Automatically Analysed | 55 |
| 3.4 | Booter Database Consistency | 56 |
| 3.5 | Automated Analysis | 58 |
| 3.6 | Concluding Remarks | 72 |
| 4 | Distinguishing Booters Based on Their Attacks | 77 |
| 4.1 | What is Advertised on Booter Websites | 78 |
| 4.2 | Measuring Booter Attacks | 80 |
| 4.3 | Booter Attacks Analyses | 84 |
| 4.4 | Booters Behind DDoS Protection Services | 92 |
| 4.5 | Concluding Remarks | 94 |
| 5 | Identifying Third-Parties and Ranking Booters | 99 |
| 5.1 | Introduction | 100 |
| 5.2 | Identifying Third-Parties | 101 |
| 5.3 | Ranking Booters | 106 |

| | | |
|-------------------------|--|------------|
| 5.4 | Concluding Remarks | 109 |
| 6 | Ethical Arguments for Booters Mitigation | 113 |
| 6.1 | Introduction | 114 |
| 6.2 | Revisiting Booter Characteristics | 114 |
| 6.3 | Justifications for Using Booters | 116 |
| 6.4 | Concluding Remarks | 127 |
| 7 | Conclusions | 131 |
| 7.1 | Summary | 132 |
| 7.2 | Revisiting Research Questions | 133 |
| 7.3 | Moving Forward from Findings | 138 |
| Appendices | | 140 |
| A | List of URLs Containing Booter Databases Dumps | 141 |
| B | List of URLs From the Booter Blacklist Initiative | 143 |
| C | Open Dataset Management | 147 |
| D | SURFnet and Dutch Prosecutor's Recommendation | 149 |
| Bibliography | | 151 |
| About the author | | 161 |

List of Figures

| | | |
|-----|--|----|
| 1.1 | Evolution of Internet-based attacks by Lipson [58]. | 5 |
| 1.2 | Historical evolution of DDoS attacks by Radware [75]. | 6 |
| 1.3 | Increase of DDoS attacks. | 7 |
| 1.4 | Elements involved with booter Websites. | 8 |
| 1.5 | Research questions. | 10 |
| 1.6 | Thesis organization. | 11 |
| 2.1 | Elements and open questions for the development of our methodology. | 19 |
| 2.2 | SURFnet data. | 37 |
| 2.3 | Analysis per quarter of year. | 38 |
| 2.4 | CDF of queries to distinct booters per quarter of year (using the same scale). | 39 |
| 2.5 | Top 10 most accessed booters per quarter of year (using same scale). | 40 |
| 2.6 | CDF of queries performed by users. | 41 |
| 2.7 | Number of access of SURFnet users to booters. | 41 |
| 3.1 | Steps of our methodology. | 49 |
| 3.2 | Generic booter database schema. | 55 |
| 3.3 | Payments per client. | 60 |
| 3.4 | Amount of money paid. | 61 |
| 3.5 | Attack types. | 68 |
| 3.6 | Attacks per client. | 69 |
| 3.7 | Probability for attacks to be relaunched less than 5 minutes later. | 69 |
| 3.8 | Cumulative distribution of attacks against a same target and the duration. | 70 |
| 4.1 | Example of inter-arrival time distribution. | 83 |
| 4.2 | Traffic rate of DNS-based attacks. | 85 |
| 4.3 | Packet size distribution (DNS). | 86 |
| 4.4 | Traffic rate of CharGen-based attacks. | 88 |
| 4.5 | Packet size distribution (CharGen). | 88 |

| | | |
|-----|--|-----|
| 4.6 | Geographical distribution of misused servers. | 89 |
| 4.7 | Continent breakdown per booter. | 90 |
| 4.8 | Percentage of time that 102 Booters are protected, sorted by the year they started to be accessed. | 93 |
| 5.1 | Booter ecosystem elements. | 101 |
| 5.2 | Domain word composition and TLDs distribution (.com and .net highlighted). | 103 |
| 5.3 | Registrars analysis based on Whois information (absolute numbers in y-axis). | 104 |
| 5.4 | Web hosting analysis based on ASes (left), with zoom-in on the ASes hidden by CloudFlare (middle), and the overall merged results (right). | 104 |
| 5.5 | (a) Top ranked booter domain names, up to 3M-th position in Alexa—red star is the current rank; blue circle is the rank of 3 month ago. (b) Distribution of price ranges for each booter, including outliers. (c) Maximum advertised attack rate in Gbit/s. (d) Blue circles: registration of domain names; and red arrow-heads their respective expiration dates. | 108 |
| 6.1 | Booter ecosystem (extended from Figure 1.4). | 115 |
| D.1 | Advise by SURFnet regarding the transparency of our research, in accordance with a Dutch public prosecutor. | 149 |

List of Tables

| | | |
|------|---|----|
| 2.1 | URL types and examples. | 21 |
| 2.2 | Average values of characteristics of 928 URLs (113 booters and 815 non-booters). | 26 |
| 2.3 | Normalized values of characteristics of 928 URLs (113 booters and 815 non-booters). | 27 |
| 2.4 | List of classification approaches order by expected accuracy for more than 3 characteristics ($n > 3$). | 29 |
| 2.5 | Results for distance metrics order by the best classification accuracy rate (CAR). | 31 |
| 2.6 | Results of k-NN approach for the best three distance metrics (Fractional, Manhattan and Cosine distance). | 32 |
| 2.7 | Probability of likelihood of each characteristic given outcome X. | 32 |
| 2.8 | Prior probabilities or base rates. | 33 |
| 2.9 | Naive Bayes classification accuracy. | 33 |
| 2.10 | Odds-ratio of the 15 characteristics using a dataset of 928 URLs (113 booters and 815 non-booters), order by the highest values. . | 34 |
| 2.11 | Results for booter classification using a weighted approach, added to the previous best values achieved via un-weighted approaches. . | 35 |
| 3.1 | Summary of tables in 23 different booter databases dumps. | 50 |
| 3.2 | Dates related to booters for checking their database consistency (DD/MM/YY). | 58 |
| 3.3 | Booter clients and total amount of money paid. | 60 |
| 3.4 | Client IP address(es) and attacks. | 63 |
| 3.5 | Details per Booter about clients using TOR. | 64 |
| 3.6 | Client countries related to attacks. | 65 |
| 3.7 | The same client email account in different booters. | 66 |
| 3.8 | Overall attack numbers and data span. | 67 |
| 3.9 | Booter infrastructure. | 71 |
| 4.1 | Alias of 14 booters, their prices and their maximum attack rate. . | 81 |
| 4.2 | Details of DNS-based attacks. | 87 |
| 4.3 | Details of CharGen-based attacks. | 89 |

| | | |
|-----|---|-----|
| 4.4 | Intersection between sets of misused systems by the tested booters. | 91 |
| 4.5 | Average fraction of time in DPSes, per year. | 93 |
| A.1 | URLs in which we found Booter databases dumps. | 141 |
| B.1 | List of Booter URLs retrieved from booterblacklist.com. | 143 |
| C.1 | Per-chapter datasets and source-code URLs. | 148 |

“Two decades of Internet but seems that nothing has changed except the scale.”

—BRUCE SCHNEIDER,
IN: DATA AND GOLIATH—THE HIDDEN BATTLES TO COLLECT
YOUR DATA AND CONTROL YOUR WORLD, 2015

Introduction

1.1 The Internet and DDoS Attacks

The number of users and devices connected to the Internet is already enormous and continues to grow steadily. In 2016, the number of Internet users was estimated as more than three billion [35], while the number of Internet devices exceeded twenty billion [87]. Alongside the growth in users and devices there is a growing variety of types of Internet usage, such as access to entire digital libraries and news about anywhere at any time. Another example of Internet usage is instantaneous worldwide communication via text, voice and video. The Internet also enables access to entertainment ranging from short videos to entire movies and from television programmes to online games as well as the ability to shop at thousands of online stores.

Furthermore, the Internet is used to access and control critical facilities and systems such as wind farms, water utilities, electricity stations, heating and surveillance systems. The Internet also enables the management and integration of social and health records and has a role in key economic activity, such as stock exchange shares and online payments.

In summary, our society relies on the availability of services and systems connected to the Internet for all sort of activities. The problem is that these systems and services have become the target of attacks. Distributed Denial of Service (DDoS) attacks are the greatest threat to the availability of these systems and services. DDoS attacks are intentional acts in which attackers orchestrate devices distributed on the Internet with the aim of overloading the memory, the processing or the link capacity of a target system.

Targets of attacks can be anything connected to the Internet. They range from specific services or applications (running on a device) to physical/virtual devices. Depending on the intensity of attacks, it may be that it is not only the intended target system that is affected. An attack that overloads the network infrastructure of a target system also affects all the other systems connected to the same infrastructure. For example, in 2016 an attack against a DNS company, DYN [32], affected access to more than sixty large websites (including

Airbnb, Amazon, BBC, CNN, Comcast, HBO, GitHub, Fox News, Netflix, The New York Times, PayPal, Visa, Spotify and Twitter) and millions of users of those websites.

The economic damage caused by DDoS attacks is also increasing. In 2015, a survey [42] reported that, on average, small and medium companies spent more than \$US50,000 recovering from a DDoS attack, while large enterprises spent an average of over \$US410,000. In 2017, the damage is estimated to be around five times greater, as another survey [66] reported an average \$US2.5M in revenue loss as a consequence of a DDoS attack.

DDoS attacks are **not** a new problem. They first appeared in the late 1990's. Since then, as DDoS attacks have increased in both number and power, they have been widely discussed. There are more than 35K academic works retrieved by Google scholar using the search term “ddos attack”. Although this problem has been widely addressed, there is still a need for the research discussed in this thesis. The reason is that DDoS attacks are continuously evolving, as outlined in the next section.

1.2 DDoS Attack Evolution

In this section we describe the evolution of DDoS attacks over five periods of time: 1982–2000, 2000–2003, 2003–2009, 2009–2012 and 2012–2017. After this, we highlight the characteristics of the final period to emphasise the need for and novelty of this thesis compared to previous works.

In 2002, Lipson [58] reported on the technical challenges in identifying Internet-based attacks and attackers [58]. That report presents the evolution of attacks over two decades of observations (from the early 1980's to the early 2000's). One of their main observations was that while attack sophistication increased, the technical knowledge of the average attacker decreased, as shown in Figure 1.1. The increased attack sophistication was due to skilled attackers who built new attack toolkits and improved their techniques to obscure their identity and their attack infrastructure (*e.g.*, packet spoofing). The knowledge of average attackers decreased due to the availability and (re)usage of these toolkits. For Lipson [58], attack sophistication was related to the techniques used to perform attacks, while the strength and frequency of attacks are not necessarily related to sophistication.

The same period reported by Lipson [58] (1982–2000), is described by Radware [75] as ‘the early days’, as depicted in Figure 1.2. Radware [75] confirms what Lipson [58] observed, that hacking techniques and tools evolved. Besides that, Radware [75] describes the first (D)DoS tools (*e.g.*, Trinoo, Tribe Flood Network (TFN), TFN2k and Shaft) that were used against targets on

the Internet (*e.g.*, against the University of Minnesota’s network). In contrast to Lipson [58], who focuses on attack sophistication and attacker knowledge, Radware [75] focuses on the historical evolution of the attacks.

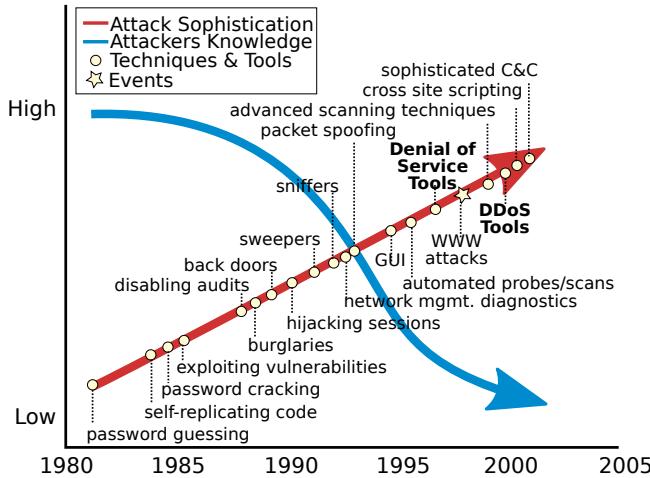


Figure 1.1: Evolution of Internet-based attacks by Lipson [58].

From 2000 to 2003, Radware [75] identifies a period he calls ‘the democratization of DDoS tools’. In this period (D)DoS tools started being shared in online hacker forums. The availability of tools enabled novice attackers without technical skills to perform attacks against well-known websites (*e.g.*, the FBI, eBay, Yahoo, Amazon and CNN) and even against the DNS root server infrastructure.

After this, from 2003 to 2009, came ‘the political agenda and criminal extortion’ period, in which DDoS attacks were politically motivated [75] (*e.g.*, the attack by North Korean hackers against South Korea and Japan [92] and the attacks by Russian hackers against Estonia and later Georgia [103]) or used for extortion purposes (*e.g.*, against Clickbank and SpamCop websites).

Radware [75] describes the period from 2009 to 2012 as ‘hacktivists and the rise of anonymous.’ In this period, DDoS attacks began to be widely used as a form of protest, also called hacktivism. People downloaded DoS tools to voluntarily participate in DDoS attacks against a target of protest. One of the most known DoS tools used for this purpose was the Low Orbital Ion Cannon (LOIC). People, relying on the claim that LOIC protects their identity (while performing attacks), called themselves “the Anonymous group” [69]. A large number of attacks have been carried out by the Anonymous group [102].

and the blooming of social media (*e.g.*, 4chan.org, reddit.com, twitter.com and facebook.com) helped hacktivism groups to convince even more people to join in all sort of protests.

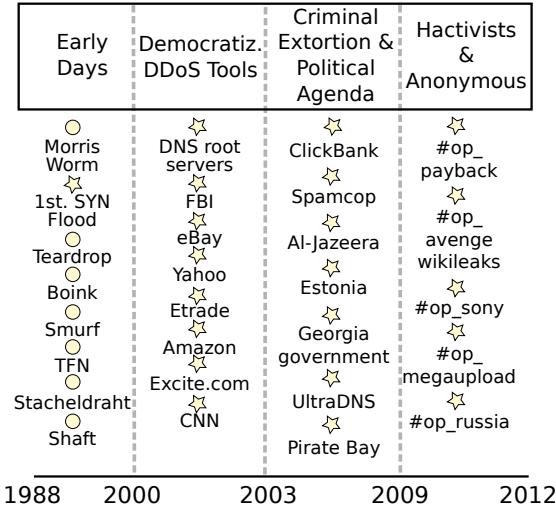


Figure 1.2: Historical evolution of DDoS attacks by Radware [75].

Many other DDoS attacks occurred during the period covered by Radware [75], from 1988 to 2012, and the motivations for attacks in the four periods are not mutually exclusive. We also observe that during the entire period the attacks and tools became more sophisticated, while the knowledge of average attackers decreased, which is the same behaviour described by Lipson [58] for the period between 1988 and 2000.

From 2012, the last year covered by Radware [75], until early 2017, the DDoS attack problem escalated not only in the number of occurrences but also in attack power. We summarise the increase in DDoS attacks in Figure 1.3. These numbers were obtained from Akamai [2, 3, 4, 5, 6, 7, 8] and Arbor Networks [11]. While from Arbor we present the record of attack power per year, from Akamai we present the total number of attacks reported per quarter. We obtain Akamai numbers by combining the increase/decrease percentage of attacks from the majority of their reports with a few other Akamai reports that contain the actual comparative numbers.

The increase in the number and power of attacks shown in Figure 1.3 is quite alarming. In summary, attacks have become more frequent and stronger over the years. For example, in 2016 the record attack peak was 1.1Tb/s, which

is more than twice the 2015 record (500Gb/s), and more than 18 times the 2011 record (60Gb/s). On the frequency of attacks, in the last quarter of 2016 Akamai observed more than 5K attacks, which is 26 times more than in 2012 (200 attacks).

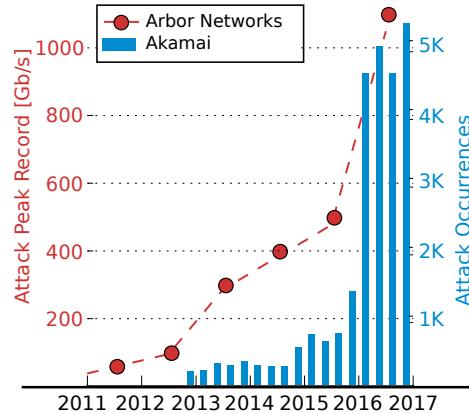


Figure 1.3: Increase of DDoS attacks.

According to security companies [8, 7], in the period between 2012 and 2016 the main responsibility for the increase in frequency and power of DDoS attacks lay with websites that offer DDoS attacks as a service, also called **booters**. From now on in this thesis, we use the word booters to refer to DDoS-as-a-service websites. We consider the rise of these websites to be the new period of the DDoS attack evolution (after the hacktivism and Anonymous periods). Our definition of a booter is a website on the public Internet that offers DDoS attacks against any system on the Internet as a (paid) service. Booters make the technical requirements to perform attacks completely transparent to their clients, who need (in general) only to pay a couple of dollars (via a third-party payment service, such as PayPal or Bitcoin), to launch DDoS attacks, as shown in Figure 1.4.

It is important to emphasise that booters are **not** websites found on the dark web, which would require proprietary software and protocols to access their content (*e.g.*, TOR and I2P). Booters are also **not** downloadable toolkits, but websites that can be accessed via any conventional browser to launch attacks, even from browsers on cellphones. In the hacker community, booter means “responsible for boot down of a given system.” Besides the term booter we also use synonyms found in the literature, such as stresser, ddoser, DDoS-as-a-service and DDoS-for-hire.

There are many reasons why booters contributed to the increase in the frequency and power of attacks. The increasing frequency is explained by the fact that booters (1) remove the need to have technical skills to perform attacks, (2) are easy to find using the most popular searching engines (Google and Bing) and (3) offer very affordable prices, for example, starting from less than \$US5 to perform as many attacks as requested over a month. The increase in attack power may be explained by the fact that booters are competing in their market, and want to guarantee that their service (*i.e.*, DDoS attacks) is better than the one offered by their competitors (other booters). In addition, attackers needed to increase their attack power given that the Internet link capacity of potential targets is growing over time (*e.g.*, based on different technologies such as ADSL, coaxial cable and fibre optics).

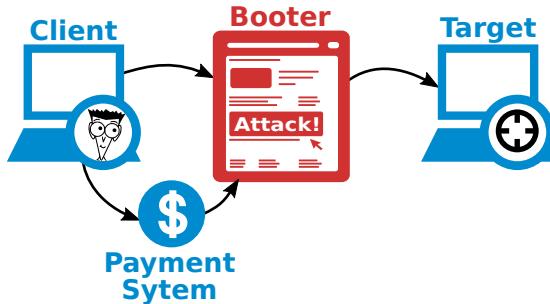


Figure 1.4: Elements involved with booter Websites.

Booters have been used for all sorts of purposes, for example: influencing the political agenda of countries, for criminal extortion and for hacktivism purposes, which are the last two phases of the DDoS attack evolution described by Radware. Booters are also very popular among people without the technical skills to perform attacks, such as online gamers who DDoS each other to gain an advantage in online matches and teenagers who DDoS their schools to prevent online exams from happening.

Comparing the four periods of DDoS attacks reported by Radware [75] (*i.e.*, the periods between 1988 and 2012) with the booters period (*i.e.*, from 2012 to 2016), we observe the same overall pattern: the knowledge of average attackers decreased, while the attack sophistication increased. In the booters period, the knowledge level of average attackers decreased to the point that attackers (*i.e.*, booter clients) need to know almost nothing to perform attacks. Attackers need only to know how to find a booter website (*e.g.*, via Google or Bing), how to pay for attacks (*e.g.*, via Paypal or Bitcoin) and to know at least one identifier of the target system (*e.g.*, IP address, URL of the website or

Skype account).

The sophistication of attack techniques used in the booters period increased more slowly than in the previous periods. For example, the peak record attacks in 2013, 2014 and 2015 have the same characteristics as the attacks in the first period of the attack evolution (the late 1990's). These were reflection and amplification attacks, which exploited UDP services (*e.g.*, DNS and NTP) using packet spoofing. The main difference between the first period of attacks and now is the number and type of exploited devices. While in 1999 the attacks peaked at a few Megabits per second and involved a few hundred devices [23], in 2016, the attack peak record of 1.1Tb/s involved 150K devices, known as Internet of Things (IoT) types of devices. However, this was a SYN flood attack, which is one of the most basic types of attack and was present in the first period of attack evolution.

1.3 Goal and Research Questions

The most important difference between the first four periods of the DDoS attacks (1982–2000, 2000–2003, 2003–2009 and 2009–2012) and the booters period (2012–2017) is related to the ability to identify attackers and attacks. In early DDoS attacks (earlier than 2000), Lipson [58] believed that identifying attackers and attacks would only become more difficult over time. However, we observed that booters expose their operations not only to potential clients but also to the research community and law enforcement agencies, by publicly offering DDoS attacks. Therefore, in this thesis, we take advantage of this observation to understand stakeholders involved with booters. We, therefore, summarise our research goal as follows.

Goal: to support mitigation actions by understanding booter websites, their clients, the infrastructure used to perform attacks and third-party companies (in)directly involved with booters.

We address this research goal in seven steps, as presented in Figure 1.5. We use Research Questions (RQ) to guide us in achieving each step. Our first step is to find booter websites (RQ1: HOW TO FIND BOOTERS?). Once we have identified a list of booters, we perform the second step, that is to detect clients accessing these booter websites (RQ2: HOW TO DETECT CLIENTS ACCESSING BOOTERS?). In our third step, we investigate the usage of booter services by these clients (RQ3: HOW DO CLIENTS USE BOOTER SERVICES?). After we understand which booters exist, which customers access these booters and how these clients use booter services, we move on to the characteristics of booter

attacks. In this fourth step, we focus on distinguishing booters based on their attacks (RQ4: DO BOOTERS HAVE DISTINCT ATTACK CHARACTERISTICS, AND IF SO, WHAT ARE THESE CHARACTERISTICS?). This step enables victims to determine which booter attacked them and to react with legal action against those responsible for the attacks.

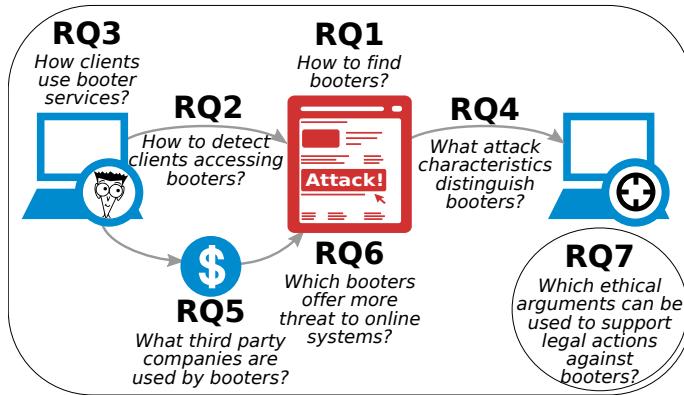


Figure 1.5: Research questions.

While the fourth step supports reactive actions against booters, in the fifth step we focus on supporting proactive actions. First, we identify third-party companies used by booters and discuss how these companies could prevent booter operations (RQ5: WHAT THIRD-PARTY COMPANIES ARE USED BY BOOTERS?). Second, we determine booters that should have a higher priority in mitigation actions (RQ6: WHICH BOOTERS ARE MOST DANGEROUS?). Finally, in the seventh step to address our thesis goal, we use the findings from most of the previous steps to discuss legal and ethical arguments around enforcing mitigation actions against booters (RQ7: WHICH ETHICAL ARGUMENTS CAN BE USED TO SUPPORT MITIGATION ACTIONS AGAINST BOOTERS?).

1.4 Approach and Thesis Organization

The approach used to answer the research questions and, ultimately, address the research goal of this thesis is measurement-based. We develop methodologies to create and analyse datasets and also retrieve data from public and private sources. All datasets and scripts (*i.e.*, source code) used in this thesis are publicly available and are listed in Appendix C.

This thesis contains seven chapters, each (except for the introduction and

conclusion chapters) addressing one or two research questions, as depicted in Figure 1.6.

First, in Chapter 2 we cover RQ1 (*how to find booters?*) and RQ2 (*how to detect clients accessing booters?*). Our approach to answering RQ1 is based on searching for, collecting and classifying any suspect URL pointing to a booter website. Then, we develop a method composed of three parts: a crawler, a scraper and a classifier. This method produces a list of booters that is used as the ground truth to answer the remaining research questions in this thesis. The list of booters generated by our method is the most comprehensive list of booter websites on the Internet (available at <http://booterblacklist.com>).

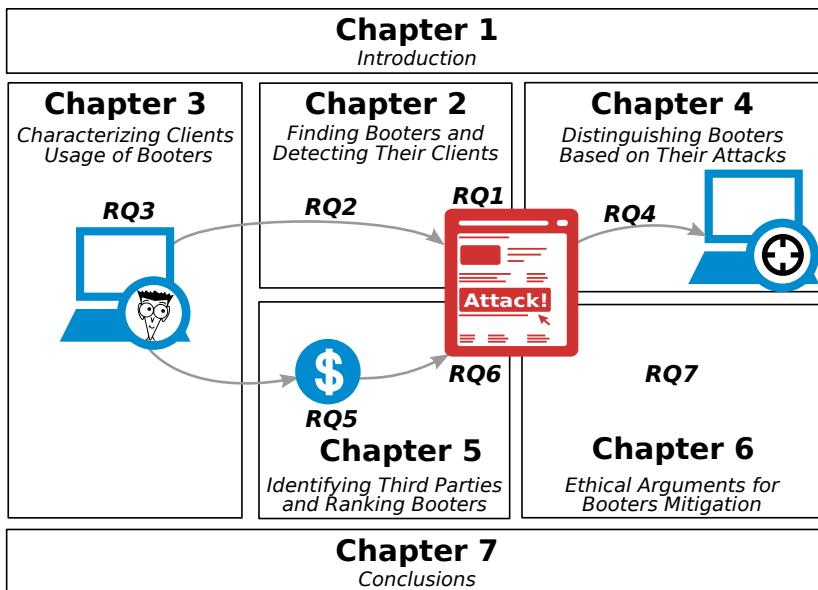


Figure 1.6: Thesis organization.

Our approach to answering RQ2 (*how to detect clients accessing booters?*) is based on monitoring network traffic using a list of booters (obtained from RQ1). We show that traditional network monitoring, based on observing users accessing IP addresses of booters, is not possible. The reason is that the majority of booter websites use the IP of web-hosting companies, thus one IP address points to several websites and not only to the booter (we discuss companies involved with booters in chapter 4 and chapter 5). We therefore use passive DNS monitoring, in which we collect DNS requests from clients to a booter within our list. The content of chapter 2 has been published in:

- J. J. Chromik, J. J. Santanna, A. Sperotto, and A. Pras. Booter websites characterization: Towards a list of threats. In *Brazilian Symposium on Computer Networks and Distributed Systems (SBRC)*, 2015 [20].
- J. J. Santanna, R. de O. Schmidt, D. Tuncer, J. de Vries, L. Granville, and A. Pras. Booter Blacklist: Unveiling DDoS-for-hire Websites. In *International Conference on Network and Service Management (CNSM)*, 2016 [82].
- J. J. Santanna, R. de O. Schmidt, D. Tuncer, J. de Vries, L. Zambenedetti Granville, and A. Pras. Booter List Generation: The Basis for Investigating DDoS-for-hire Websites. *International Journal on Network Management (IJNM)*, 2017 [83].

While in RQ2 we highlight clients accessing booters, in RQ3 (*how do clients use booter services?*) we highlight the attacks requested by these clients. We fully cover RQ3 in Chapter 3. Our approach to addressing RQ3 is based on analysing (leaked and publicly available) booter databases that contain clients' information. These databases are a good source of information to connect booter attacks to clients. We therefore propose a semi-automated analysis methodology that can be applied to any booter database. We also apply this method to fifteen booter databases and reveal our findings. The content of this chapter has been published in:

- J. J. Santanna, R. Durban, A. Sperotto, and A. Pras. Inside Booters: An Analysis on Operational Databases. In *IFIP/IEEE International Symposium on Integrated Network Management (IM)*, 2015 [80].

Then, in Chapter 4, we fully cover RQ4 (*do booters have distinct attack characteristics, and if so, what are these characteristics?*). Our approach to addressing this question is based on measuring the attacks performed by booters. For this purpose we acted as a client of booters and requested these booters to perform attacks against a controlled environment. Then we measured and analysed the attack traffic that was sent to this controlled environment. Besides analysing how booters can be differentiated by their attack characteristics, we analyse whether booters deliver what is advertised on their websites. The content of this chapter has been published in:

- J. J. Santanna, R. van Rijswijk-Deij, A. Sperotto, R. Hofstede, M. Wierbosch, L. Zambenedetti Granville, and A. Pras. Booters-An analysis of DDoS-as-a-Service Attacks. In *IFIP/IEEE Symposium on Integrated Network and Service Management (IM)*, 2015 [81].

- A. Pras, J. J. Santanna, J. Steinberger, and A. Sperotto. DDoS 3.0 - How Terrorists Bring Down the Internet. In *International German Informatics Society (GI) and Technology-Enabled Trading Solutions (ITG) Conference*, 2016 [70].
- J. Steinberger, J. J. Santanna, E. Spatharas, H. Ammer, N. Breuer, B. Kuhnert, U. Piontek, A. Sperotto, H. Baier, and A. Pras. “Ludo”: Kids Playing Distributed Denial of Service. In *TERENA Networking Conference (TNC)*, 2016 [88].

In Chapter 5 we fully cover RQ5 (*what third-party companies are used by booters?*) and RQ6 (*which booters are most dangerous?*). Our approach to answering RQ5 is based on discovering any third-party company (in)directly related to the IP addresses that point to botter websites and the information from the botter domain name. We combine datasets that we collect ourselves with those retrieved from public sources to reveal top level domains, domain registrars, web-hosting companies, cloud-based security providers, payment systems and web-searching companies used by booters. Besides identifying third-party companies, we also discuss how these companies could prevent botter operations.

Our approach to answering RQ6 is based on collecting and analysing data to rank of booters according to the threat they pose to Internet systems. The most obvious approach would be to compare the frequency and attack power of booters' attacks. However, this approach is restricted to large network security companies that can observe attacks against their customers. Therefore, we propose a heuristic for ranking booters based on five aspects: (1) the level of popularity of the botter websites, (2) the price charged, (3) the maximum attack power advertised, (4) the creation and (5) the expiration date of the domain name. The content of this chapter has been published in:

- J. J. Santanna, R. de O. Schmidt, D. Tuncer, A. Sperotto, L. Z. Granville, and A. Pras. Quite Dogs Can Bite: What Booters We Should Go After? and Which Are Our Mitigation Options? *IEEE Communications Magazine*, 55(7):50–56, 2017 [84].

In Chapter 6 we cover RQ7 (*which ethical arguments can be used to support mitigation actions against booters?*). Our approach in addressing RQ7 is based on investigating cases where DDoS attacks are considered ethically acceptable and then proving that booters do **not** fit into these categories. We use the findings from the previous chapters to assure ourselves that the services from booters are likely to be illegal and that their usage is unethical. With these conclusions, we expect to support law enforcement agencies in acting to mitigate booters. The content of this chapter has been published in:

- D. Douglas, J. J. Santanna, R. de O. Schmidt, L. Z. Granville, and A. Pras. Booters: Can Anything Justify Distributed Denial-of-Service (DDoS) Attacks for Hire? *Journal of Information, Communication and Ethics in Society (JICES)*, 15(1), 2017 [25].

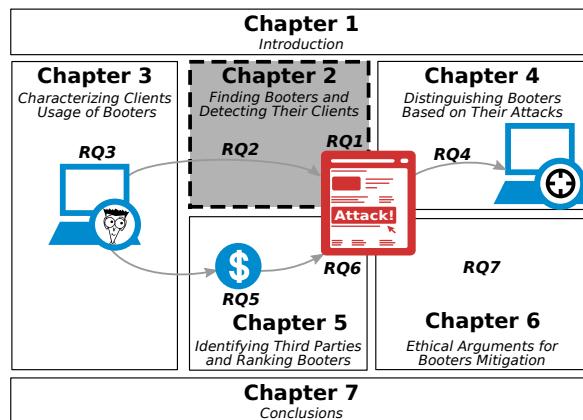
Finally, in Chapter 7, we draw the overall conclusions of the research discussed in the other chapters. In this chapter we also discuss potential future research directions.

"The art of war teaches us to rely not on the likelihood of the enemy's not coming, but on our own readiness to receive him [...] if you know the enemy and know yourself, you need not fear the result of a hundred battles."

—SUN TZU,
IN: THE ART OF WAR, 5th CENTURY BC

Finding Booters and Detecting Their Clients

In this chapter, we aim to answer the question of **how to find booters?** (**RQ1**). A crucial step in enabling an in-depth investigation of the booter ecosystem is finding booter websites. To answer this question we present a rigorous methodology to collect and classify any suspected URL pointing to a booter website. Afterward, we present a case of practical use of a booter list to answers **how to detect clients accessing booters?** (**RQ2**). We apply network-monitoring approaches to detect users within a network infrastructure that accessed booters (from a list generated via RQ1).



The organisation of this chapter is as follows:

- In Section 2.2, we outline our method for identifying keywords and sources of information to collect URLs suspected of being booter websites;
- In Section 2.3, we describe the characteristics that define actual booter websites;
- In Section 2.4, we analyse several classification approaches, to determine which of them fits best into our objective function for booter website classification;
- In Section 2.5, we present a case of practical use of booter lists;
- In Section 2.6, we discuss each part of our methodology and highlight our contributions and the impact of them.

2.1 Motivation and Challenges

Although existing Booter investigations [39, 41, 17] are valuable to the state-of-art, they were neither deep nor broad. These investigations were restricted to a few Booters that lead to very well spread incidents on the Internet. For example, Booters that attacked very well-known targets, such as Microsoft and Sony [94], and Booters that performed very powerful attacks, which achieved hundreds of trillions (Giga) bits per second. It is still not clear how broad is the phenomenon, *i.e.*, what and how many Booters are there. The contribution of this chapter is a methodology for automatic generation of a comprehensive and accurate booter Website list. This list is intended for enabling extensive investigation of the Booter phenomenon. As a consequence, the list generated by our methodology is essential for the remaining parts of this thesis.

Although existing booter investigations [39, 41, 17] are valuable, they have been neither deep nor broad. Previous investigations were restricted to a few booters that led to widespread incidents on the Internet. For example, booters that attacked widely known targets, such as Microsoft and Sony [94] or booters that performed very powerful attacks, which achieved hundreds of trillions (Giga) of bits per second. It is still not clear how broad the phenomenon is, *i.e.*, what and how many booters there are. This chapter contributes a methodology for automatic generation of a comprehensive and accurate booter website list. This list is intended to enable extensive investigation of the booter phenomenon. As a consequence, the list generated by our methodology is essential for the remaining parts of this thesis.

We define three main requirements for our method of creating a list of Booters: automatic, comprehensive, and accurate. Being automatic has the reason on the dynamicity of the Booter phenomenon. Often, new Booters appear and others disappear and a manual strategy for Booter list generation would not be suitable. The comprehensiveness is required to enable understanding how broad is the booter phenomenon. The third requirement, accurate, is critical because we do not want any non-booter Website to suffer investigation or mitigation on account of our listing method.

We define three main requirements for our method of creating a list of booters: it must be automatic, comprehensive and accurate. It must be automatic due to the dynamicity of the booter phenomenon. Often, new booters appear and others disappear and a manual strategy for booter list generation would not be suitable. Comprehensiveness is essential to understand the breadth of the booter phenomenon. The third requirement, accurate, is critical because we do not want any non-booter website to suffer investigation or mitigation due to our listing method.

To meet those three requirements three elements are needed: (1) a crawler,

(2) a scraper, and (3) a classifier. The crawler is responsible for collecting URLs that are suspected to be an actual Booter Website. The scraper, in turn, collects detailed information on the list of suspected URLs. Finally, the classifier, analyses the characteristics of suspected URLs to categorize whether they point to a Booter Websites or not. Each one of these three elements has specific open questions that we address in this chapter. Figure 2.1 shows the elements and open questions for the development of our methodology.

To meet these three requirements, three elements are needed: (1) a crawler, (2) a scraper and (3) a classifier. The crawler is responsible for collecting URLs that are suspected of being booter websites. The scraper collects detailed information on the list of suspect URLs. Finally, the classifier analyses the characteristics of suspect URLs to categorise whether they point to a booter website or not. Each of these three elements has specific open questions that we address in this chapter. Figure 2.1 shows the elements and open questions for the development of our methodology.

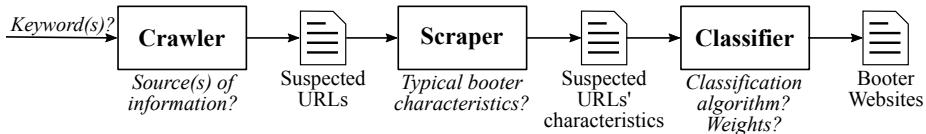


Figure 2.1: Elements and open questions for the development of our methodology.

While the comprehensiveness requirement is connected to the crawler, accuracy is related to the scraper and the classifier. To be comprehensive, the crawler must be able to retrieve information from a consistent source of information. As an illustration, if one wished to fill up the petrol tank of a car, one would not go to a food shop. It is more consistent to go to a petrol station. In addition to the source of information, the crawler must receive a coherent set of keywords to search for the source of information. For example, if one is searching for petrol, one would not ask for information about flowers. In this case, it is more coherent to ask for information using related words, such as petrol, gasoline or diesel.

As a first step towards accuracy in the generation of a booter list, the scraper must retrieve, from the suspect URLs, the characteristics that define a typical booter website rather than a generic website. The second step is to define the best algorithm for booter website classification. There are many algorithms for website classification. Our aim is to find an algorithm that classifies booters and non-booter websites based on the set of characteristics collected by the scraper. Finally, the third step in meeting the accuracy requirement is to investigate the

usage of weights applied to booter characteristics. In the literature, weighted approaches improve the accuracy of website classification. We would therefore like to know whether this is also the case for booter website classification.

In the next section, we identify the sources of information and the keywords that enable us to collect URLs suspected of being booter websites. Then in section 2.3, we describe the characteristics that we use to define booter websites. In section 2.4, we use these characteristics to analyse classification approaches and determine which best fits our objective function for booter website classification. After covering all the requirements of our methodology, in section 2.5, we present a practical use case of booter lists and highlight other potential usages. We conclude this chapter by discussing each part of our methodology, highlighting our contributions and their impact.

2.2 Crawler: Listing Suspect Booters URLs

2.2.1 Defining the Source of Information

There are three locations for finding websites: (1) the public Web, (2) the deep Web and (3) the dark Web. While on the public Web websites are indexed and accessible via conventional search engines (*e.g.*, Google and Bing), on the deep Web websites are deliberately not indexed by search engines (*e.g.*, a webpage behind a login), although the websites can still be accessed via a conventional browser. Websites in the dark Web cannot be accessed using a conventional browser and proprietary protocols or special software are required (*i.e.*, TOR and Freenet).

By definition, the success of booters comes from the fact they are public and easily reachable by their primary customers, *i.e.*, skiddies and laymen. As a matter of completeness, we must therefore partially include the deep and dark Web. However, we focus most of our attention on the public Web, which we also call the Internet in this thesis. There are three main search engines on the public Web: Bing, Yahoo and Google. The latter is recognised as retrieving the most websites [91]. We therefore rely on Google to find booter websites. There are four types of URLs resulting from Google searches, presented in Table 2.1.

In Table 2.1, type 1 URLs usually point to the main page of a website. These URLs may or may not contain the subdomain www and end with all kinds of Top Level Domain (TLD) (*e.g.*, .com, .nl and .net). Type 1 URLs are likely to be the main page of a booter website, but further analysis is needed. We therefore include this URL type in our analysis, as discussed in the next section. Type 2 URLs usually point to a sub-page of a website. There are some exceptions, for example when the webpage+format is “index.html”. In this example, as in a type 1 URL, it is likely to point to the main page of a website. Type 2 URLs

are usually part of a booter site. We therefore include this type in our further analysis.

Table 2.1: URL types and examples.

| URL Type | URL pattern | Examples |
|----------|---|--|
| 1 | [www.]potential-booter.tld | quezstresser.com and databooter.com |
| 2 | [www.]potential-booter.tld/webpage.type | zstress.net/features.php and booter.xyz/members |
| 3 | [www.]domain.tld/[.../]potential-booter | twitter.com/booter and www.safeskyhacks.com/Forums/showthread.php?39-Top-10-DDoser-s-(booters-Stressers) |
| 4 | potential-booter.domain.tld | ebooter.5gbfree.com and booterddos.890m.com |

Type 3 and 4 URLs are not likely to point directly to a booter site. The former is usually a webpage describing a booter, for example blog posts and social network accounts. The latter is usually a page in a subdomain, for instance, websites that show information about booters. Although these two types could potentially point to a booter website, we decided to focus on type 1 and 2 URLs. The reason for this is that booters claim to be private companies. Therefore, we expect booters to have their websites registered in a known TLD, such as “.com.” On <http://booterblacklist.com>, almost 70% of booter sites collected over four years use the TLDs .com and .net.

We made two exceptions to excluding type 3 URLs. The first was when the URL pointed to a Youtube video. In this case, instead of considering URLs from the Google search, we collected any URL within the description of the video that the initial URL was pointing to. We decided to include type 3 URLs from Youtube because it is the most popular way to advertise booter services to their primary customers (*i.e.*, “dummies” and laymen). For the same reason, *i.e.*, venue of advertisement, we included posts from <http://hackerforums.net>. Even though this hacker forum is considered deep Web, it is usually the

first place where booters advertise their services. We included the URLs found in posts in the category “SST” (*i.e.*, stress tester) in the “Market Place” section of the forum.

In addition to automatically collecting URLs from Google, Youtube video descriptions and hackerforums.net, we manually analysed URLs from ahmia.fi and torsearch.es. These two websites are popular search engines in the TOR network, which is considered part of the dark Web. However, preliminary investigations showed that the few URLs returned from these websites consisted of a subset of those already identified from the analysis of the three previous sources of information. This observation supports the belief that booters mainly advertise their services on the public or deep Web. We therefore decided to exclude the dark Web sources from our investigation.

2.2.2 Identifying a Representative Set of Keywords

Our approach to identifying a representative set of keywords relies on the frequency of words found in the meta-information of booter websites, *i.e.*, description and keywords. The representativeness of the set of keywords improves as the number of actual booter sites increases. First, in mid-August 2013, we performed a Google search using only the keyword booter. Then we manually selected the first hundred webpages that were related to booters, for example, booter websites or blog posts that described/analysed/advertised booters.

From those webpages, we automatically collected (*i.e.*, scraped) the meta-information for the landing page of a URL and calculated the word frequency. From the resulting list of words we manually removed (1) generic words, (2) words related to attack types and (3) adjectives. Three examples of each type of word are: denial of service, DoS, DDoS; SYN, UDP, Slowloris; cheap, powerful and efficient. After we sanitised our list, we came up with five keywords: booter, Stresser, DDoSer, DDoS-for-hire and DDoS-as-a-service.

Over four years of research, our methodology for finding a representative set of keywords was used many times. In the final stage of our observations (late 2016), we noted that all URLs identified by the keywords DDoS-for-hire and DDoS-as-a-service were also associated with at least one of the three other keywords. Therefore, we now consider the set of keywords booter, Stresser and DDoSer the most representative for finding websites related to booters. We use these three keywords in the remaining parts of this chapter and thesis.

2.2.3 Crawler Development and Our Training Dataset

After identifying a representative source of information (Google search, Youtube video description and posts from hackerforum.net) and the set of keywords to perform queries (booter, Stresser and DDoSer), we faced a technical problem: the existing web crawling tools are either private or too limited for our particular purpose. The works in [100, 55, 19], for example, return less than fifty results from Google queries. There is a comprehensive list of open source crawlers at https://en.wikipedia.org/wiki/Web_crawler. However, either they do not retrieve a significant amount of results or we had difficulties in deploying them. None of the fifteen crawlers that we managed to deploy allowed us to retrieve information from the deep Web (*i.e.*, hackerforums.net).

To overcome the limitations of existing crawlers and extract as many results as possible, we developed a crawler that mimics user behaviour when visiting a webpage. We applied our crawler to all our sources of information, *i.e.*, Google, Youtube and hackerforums.net. Note that this approach mimics the behaviour of an actual user searching for a booter website. The source code for our crawler is available at <https://github.com/jjsantanna/booter-black-List/tree/master/Crawler>, and it retrieves all URLs that Google makes available to users to access via a browser, *i.e.*, a number that usually exceed 500 URLs per keyword search.

By default, Google omits entries considered “very similar”. We included those entries by adding “&filter=0” to the HTTP request. Another observation is that although Google claims to retrieve millions of results in a few seconds, in practice users can only access a smaller number of results. In applying our crawler to Google and Youtube, the only difference is that we look at Google content to retrieve URLs that potentially point to booters, while for Youtube we look at the video’s description and for hackerforums.net we look at posts.

The total number of distinct URLs collected by our crawler in this preliminary phase was 928, which is used as our training dataset for the remainder of this chapter.

2.3 Scrapper: Collecting URL Information

The second step of our methodology for generating a blacklist of booters consists of acquiring information from each URL collected by our crawler (as described in the previous section). We used two criteria to define which characteristics to analyse. The first relies on the most used features for general website classification. The second relies on our preliminary observations about booters. Based on our first criteria we defined the following features:

1. **Number of pages** [56, 57]: the total number of internal pages in the website;
2. **Content size** [56]: the number of words of visible content on the landing page;
3. **Content dictionary** [57]: defined by the ratio between the number of matching words matching our keywords (*i.e.*, booters, Stresser, and DDoSer) and the content size;
4. **URL length** [56]: the number of characters in the URL, excluding the domain name;
5. **Depth level** [31]: the maximum amount of inbound hyperlinks to reach any internal page on the website;
6. **Outbound hyperlinks** [31]: the number of hyperlinks pointing to other domains (outbound);
7. **Domain expiration time** [57]: time span between the current date and the expected expiration date for the URL's domain name;
8. **Login-form depth level** [59]: number of links required to reach the login form (every booter website contains a login form);
9. **Alexa rank** [43]: the website rank within Alexa's worldwide ranking (<http://alexa.com>).

Based on the second criteria, we included the following characteristics:

1. **Domain age**: the time span of the domain name since its registration. We decided to include this metric after noticing that booter websites usually have earlier registration dates than non-booter websites retrieved when we use the keywords defined in the previous section (*i.e.*, booter, Stresser and DDoSer);
2. **DDoS Protection Service (DPS) subscription**: the time span of the domain name since its registration. We decided to include this metric after noticing that booter websites usually have earlier registration dates than non-booter websites retrieved when we use the keywords defined in the previous section (*i.e.*, booter, Stresser and DDoSer);
3. **WHOIS private**: determines whether sensitive information about a domain name (*e.g.*, contact name, address and email) is retrievable using WHOIS protocol. We noticed that booters are likely to obscure their information;

4. **Resolver indication:** determines whether a website has a service that reveals IP addresses of target systems based on *e.g.*, the domain name, a Skype account or an online game account. We rely on the observation by Krebs [48], which states that booters offer extra services such as IP resolvers;
5. **Terms of services (ToS) page:** indicates whether the website contains a page of rules of use for the service. We observed that booters usually add a ToS page to prevent being blamed for attacks and shift responsibility to their customers.

To determine the representativeness of each of the 15 characteristics we used our training dataset (a list of 928 URLs collected using our crawler and described in the previous section). From a manual analysis, we identified 113 URLs from this list as actual booters and 815 non-booter websites. We then scraped each of the 928 URLs, collecting data related to each of the 15 characteristics of interest. Table 2.2 presents our results.

Table 2.2 shows the average and the normalised values for the 15 characteristics for further determining whether a suspect URL is a booter website. Our first observation is that some characteristics clearly show the difference between booter and non-booter websites. For example, in the first line of Table 2.2, the average number of pages for booters is more than 100 times smaller than for the non-booter websites. There are some characteristics that do not show a clear difference between booters and non-booters, *e.g.*, as shown in the last line of Table 2.2, the average values for booters and non-booters that have a terms of service page. Another important observation is that the results have different scales of values. Therefore, to make them comparable for a further computational purpose, we decided to normalise the results.

2.3.1 Normalizing Characteristic Values

We applied different types of normalisation depending on the values scale of the 15 characteristics. Binary interpolation, linear interpolation and quadratic interpolation are used to normalise the values. Six of the 15 website features are binary, *i.e.*, they are either scored 0 or 1. The characteristics that require binary interpolation are URL type, WHOIS private, DPS subscription, page rank, resolver indication and terms of service page. The normalisation procedure for binary characteristics involves a decision boundary for when its preliminary values result in a score of 1.

The remaining nine features score to significant intervals, which we normalise to the unit interval. This involves selecting a particular range of values to

Table 2.2: Average values of characteristics of 928 URLs (113 booters and 815 non-booters).

| # | Description | booters | non-booters |
|----|------------------------|-------------------|-------------------|
| 1 | Number of pages | 7.88 | 981.75 |
| 2 | Outbound hyperlinks | 0.41 | 14.10 |
| 3 | Domain age | 395.96 | 3564.29 |
| 4 | Page rank | 1.1×10^7 | 3.2×10^6 |
| 5 | Content size | 127.00 | 679.08 |
| 6 | DPS subscription | 0.73 | 0.21 |
| 7 | URL length | 24.93 | 53.65 |
| 8 | WHOIS private | 0.73 | 0.28 |
| 9 | URL type | 1.04 | 1.20 |
| 10 | Domain expiration time | 310.93 | 812.22 |
| 11 | Depth level | 0.92 | 1.75 |
| 12 | Content dictionary | 0.039 | 0.014 |
| 13 | Login-form depth level | 1.38 | 2.06 |
| 14 | Resolver indication | 0.22 | 0.19 |
| 15 | Terms of services page | 0.47 | 0.44 |

apply a linear interpolation. Equation 2.1 describes the linear equation used to normalise the nine characteristics.

$$S_n = 1.0 - \frac{x - min}{max - min} \quad (2.1)$$

The values of min and max denote the interval range we mapped to the unit interval. We carefully selected the max and min for each individual characteristic based on the 924 URLs. Values outside the chosen range are clamped to 0.0 and 1.0 respectively.

$$S_n = ax^2 + bx + c \quad (2.2)$$

Finally, we observed that the number of pages normalisation was better suited to quadratic interpolation. This normalisation approach favours smaller websites and largely dismisses larger websites. For this reason, we consider that metric better reflects the normalised number of pages, presented in Equation 2.2. Applying both values of the number of webpages, range 0 (min) and 50 (max), we obtain the values of a, b, and c in Equation 2.2 as -0.0004, x equal to zero and one, respectively. Table 2.3 shows the normalised results for each of the 15 characteristics in Table 2.2.

The normalisation does not affect our observation of characteristics that are distinctive or similar for booter and non-booter websites. For example,

Table 2.3: Normalized values of characteristics of 928 URLs (113 booters and 815 non-booters).

| # | Description | Transformation | Range | Normalized Values | |
|----|------------------------|----------------|------------------------|-------------------|-------------|
| | | | | booters | non-booters |
| 1 | Number of pages | quadratic | [0-50] | 0.93 | 0.23 |
| 2 | Outbound hyperlinks | linear | [0-2] | 0.84 | 0.19 |
| 3 | Domain age | linear | [-,1180] | 0.78 | 0.14 |
| 4 | Page rank | binary | [2×10^5 , -] | 0.90 | 0.30 |
| 5 | Content size | linear | [50,250] | 0.70 | 0.16 |
| 6 | DPS subscription | binary | - | 0.71 | 0.21 |
| 7 | URL length | linear | [15,30] | 0.36 | 0.07 |
| 8 | WHOIS private | binary | - | 0.71 | 0.29 |
| 9 | URL type | binary | - | 0.96 | 0.80 |
| 10 | Domain expiration time | linear | [183,365] | 0.90 | 0.61 |
| 11 | Depth level | linear | [0,2] | 0.87 | 0.57 |
| 12 | Content dictionary | linear | [0.01,0.05] | 0.49 | 0.24 |
| 13 | Login-form depth level | linear | [0,2] | 0.52 | 0.27 |
| 14 | Resolver indication | binary | - | 0.24 | 0.19 |
| 15 | Terms of services page | binary | - | 0.47 | 0.44 |

the number of pages and the terms of service page shows the same conclusion as before the normalisation. The normalised values of booter characteristics (displayed on a grey background) are crucial to the remainder of this chapter, as they define the actual features of a “typical booter website”.

2.4 Classifier: Determining Booter Websites

The final step of our methodology for collecting booters is the classification of potential booter websites found through the previous two steps (section 2.2 and section 2.3). There are many well-established classification methods that can be used to classify websites, *e.g.*, [1, 34, 54]. However, there is not one single classification method that succeeds for all the cases. In this section, we evaluate the best classification method from eight well-established methods. First (in subsection 2.4.1), we describe the metrics used to measure classification accuracy, which we apply in our analysis (in subsection 2.4.2), to identify the best classification method for booter websites.

2.4.1 Classification Accuracy Metrics

The accuracy of a classification approach is measured in terms of successes and errors, typically given in a confusion matrix [31]; in this matrix a URL is classified into one of the following groups:

- **True positive (T_P):** a website correctly classified as a booter;
- **True negative (T_N):** a website correctly classified as a non-booter website;
- **False positive (F_P):** a non-booter website incorrectly classified as a booter;
- **False negative (F_N):** a booter website incorrectly classified as a non-booter website.

The *classification success* is defined by the Classification Accuracy Rate (CAR) given by:

$$CAR = (T_P + T_N)/n \quad (2.3)$$

where n is the total number of tested websites. The misclassification (error) rate is given by the *false positive error rate* FP_{er} and the *false negative error rate* FN_{er} , which are given by:

$$FP_{er} = F_P/n \quad (2.4)$$

and, respectively:

$$FN_{er} = F_N/n \quad (2.5)$$

where F_P is the total number of false positives and F_N the total false negatives.

Our goal for the booter website classification is based on the following objective function.

$$FO(\text{threshold}) = \begin{cases} \max CAR \\ \min FP_{er} \mid FP_{er} \leq FN_{er} \end{cases} \quad (2.6)$$

2.4.2 Towards the Best booter Classification Method

There are a multitude of website classification approaches. To determine the best classification method for booter websites, we analysed the eight most used methods from the website classification literature. These are presented in Table 2.4. We did not include, for example, the Hamming

distance [61], Genetic algorithms [10], or Support Vector Machines [36, 104] as we considered the results of use of the eight most used approaches in similar fields (*e.g.*, classification of phishing and child pornography websites) to be very promising.

Table 2.4: List of classification approaches order by expected accuracy for more than 3 characteristics ($n > 3$).

| Classification Approach | Reference | $n > 3$ | Complexity | Efficiency |
|-------------------------|---------------|---------|------------|------------|
| Euclidean distance | [1, 18] | weak | low | high |
| Sqr. Euclidean distance | [38] | weak | low | high |
| Manhattan distance | [1, 33] | medium | low | high |
| Cosine distance | [54] | medium | low | high |
| Fractional distance | [1, 34] | strong | medium | high |
| k-Nearest Neighbors | [54, 89, 101] | strong | low | medium |
| Naive Bayes | [56, 57, 43] | strong | medium | high |

From Table 2.4, based on the literature, Euclidean and Squared Euclidean distances are not expected to produce satisfactory classification rates for booter websites because we use 15 characteristics. Fractional distance, k-Nearest Neighbours, and Naive Bayes are expected to produce better results.

The first five classification methods in Table 2.4, are based on distance metrics. These methods aim to classify a vector \vec{v} , which contains a set of n -dimensional characteristics, based on another vector \vec{p} that is our “perfect” set of features. When the distance between the two vectors is smaller than a defined threshold, the vector \vec{v} is classified positively, otherwise it is classified negatively. To meet the objective function defined in the previous section, we must find a threshold that, when compared to the distance between many \vec{v} and \vec{p} , produces the highest CAR and lowest FPer. To analyse each of the five distance metric approaches we considered the following:

- each \vec{v} : is a list of 15 characteristics of a suspected URL (from a new list of 465 URLs collected using section 2.2 and section 2.3);
- \vec{p} : is the normalised values of the 15 characteristics of 928 URLs from our training dataset (previously presented in Table 2.2);
- *threshold*: value varied from 0 to 1 with steps of 0.01.

For each value of the threshold, we calculate the CAR, FPer, and FNer based on the distance between \vec{v} and \vec{p} . We manually analysed the 465 suspect URLs and observed that the set contains 140 booters and 325 other websites. Table 2.5 presents our results. The table is sorted using the highest CAR.

Of all the distance metrics, Manhattan achieves the best result with 92.7% accuracy. Variation of the threshold generates the same two patterns in all the classification methods. The first pattern CAR increases with the threshold value. The second pattern CAR increases proportionally to the decrease in FPer. Both patterns have an exception when FNer > 0. From this point, CAR has a reversal and starts dropping. Due to this, in all the graphs the value of threshold that best fits the objective function is right before FNer is equal to FPer (depicted in the graphs of Table 2.5 as small circles).

In addition to the results for the distance metrics, in Table 2.6 we summarise the results of the k-NN approach. In contrast to the distance metrics, the inputs for the k-Nearest Neighbours metric are defined using actual distance metrics. Another difference is that k-NN requires an empirical value for k , which decides whether a URL is a booter depending on the k th closest URLs (from the trainer dataset). We varied k from 1 to 15 (steps of 1) using each distance metric as input to k-NN and hoping that the k that gives the best value of CAR is lower than 15.

Table 2.6 shows the CAR when using Manhattan, Cosine and Fractional distances. For greater clarity, we omit the other methods. While the former two methods performed better in the previous analysis (Table 2.5), the latter achieved the best CAR using the k-NN approach. Our choice of the values of k was successful, given that the best CAR for all metrics has the value 10, which is smaller than 15. Notice, in Table 2.11, that even the best result of k-NN (considering the Fractional distance and $k = 10$) does not improve the accuracy of any previous metric (Table 2.5). The Manhattan distance metric approach, without considering k-NN, is still the best performer for classification accuracy (92.7%).

To analyse the probabilistic approach Naive Bayes, no threshold parameter is required. This method is entirely dependent on the calculated probabilities of the training dataset (928 URLs). The Naive Bayes classification metric assumes all individual characteristics to be binary, *i.e.*, a feature is either true or false. As some of the booter website characteristics are decimal values in the unit interval, we first have to transform these to their binary equivalents before calculating the individual probabilities. This transformation is accomplished by converting all decimal valued metrics of value higher or equal to 0.5 to 1.0 and similarly the other way around. For instance, if we take a normalized number of pages characteristic score of 0.72, which is equal or higher than 0.5, we convert this score to 1.0. Given the whole training dataset of binary characteristic values, we can calculate all individual characteristic probabilities as resulted in Table 2.7.

Furthermore, booters and non-booters are not evenly distributed in the training dataset. Therefore, we calculate the probabilities of any random feature

Table 2.5: Results for distance metrics order by the best classification accuracy rate (CAR).

| Distance Metric | Graph | Tshld | CAR | FP _{er} | FN _{er} |
|-----------------|-------|-------|-------|------------------|------------------|
| Manhattan | | 0.62 | 0.927 | 0.024 | 0.049 |
| Euclidean | | 0.41 | 0.920 | 0.030 | 0.049 |
| Fractional | | 0.82 | 0.914 | 0.024 | 0.062 |
| Cosine | | 0.78 | 0.914 | 0.049 | 0.037 |
| SQR Euclidean | | 0.66 | 0.908 | 0.030 | 0.062 |

Table 2.6: Results of k-NN approach for the best three distance metrics (Fractional, Manhattan and Cosine distance).

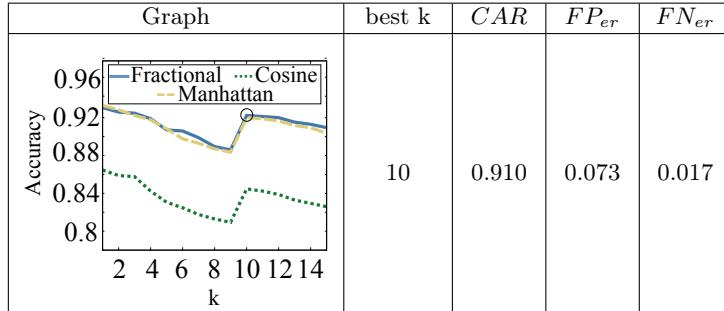


Table 2.7: Probability of likelihood of each characteristic given outcome X.

| Characteristic | X = booter | X=non-booter |
|--------------------------------|------------|--------------|
| P(Number of pages / X) | 0.97 | 0.23 |
| P(URL type / X) | 0.93 | 0.80 |
| P(Average depth level / X) | 0.94 | 0.67 |
| P(Average URL length / X) | 0.37 | 0.06 |
| P(Domain age / X) | 0.89 | 0.14 |
| P(Domain res. duration / X) | 0.89 | 0.62 |
| P(WHOIS private / X) | 0.38 | 0.28 |
| P(DPS / X) | 0.72 | 0.18 |
| P(Page rank / X) | 0.85 | 0.35 |
| P(Average content size / X) | 0.74 | 0.16 |
| P(Outbound hyperlinks / X) | 0.92 | 0.20 |
| P(Category-specific dict. / X) | 0.52 | 0.19 |
| P(Resolver indication / X) | 0.22 | 0.19 |
| P(Terms of services page / X) | 0.44 | 0.36 |
| P(Login-form depth level / X) | 0.82 | 0.66 |

vector being either a booter or a non-booter website. We calculate these prior probabilities by dividing the samples of a given outcome by the total number of samples as shown in Table 2.8. From all the probabilities in Table 2.7 and Table 2.8, we determine the CAR, FPer, and FNer as presented in Table 2.9.

Up to this point of our investigation, among all the classification approaches, *i.e.*, the five distance metrics, k-Nearest Neighbors and Naive Bayes, the best CAR was found with the Manhattan distance (CAR equal to 92.7%). In the next sub-section, we investigate whether we can improve even more this classification

Table 2.8: Prior probabilities or base rates.

| Outcome | Base rate |
|---------------|-----------|
| P(booter) | 0.1001 |
| P(non-booter) | 0.8999 |

Table 2.9: Naive Bayes classification accuracy.

| Classification Approach | CAR | FP_{er} | FN_{er} |
|-------------------------|--------|-----------|-----------|
| Naive Bayes | 0.9132 | 0.056 | 0.032 |

accuracy by adding weights to booter characteristics.

2.4.3 Classification Considering Weighted Characteristics

In the previous classification approaches, we considered that each of the 15 booter characteristics has an equal significance in the classification process. Therefore, we expected that the usage of weights should favour stronger characteristics over weaker characteristics, improving the classification accuracy. We considered two approaches to add weights to booter characteristics: odds-ratio metric [29] and machine learning algorithms. Both are commonly used in the website classification literature.

The definition of odds-ratio is the occurrence of a given characteristic in the positive class compared to in the negative class. For example, consider a list of 100 suspicious URLs from which 40 are actual booters and 60 non-booters; and that 35 of these booters have a terms of service page (ToS), while only 12 of non-booter websites do. So, the odds rate of ToS for booter is 35 (*i.e.*, have ToS) divided by 5 (40 minus 35) that do not have ToS, totaling 7 (35/(40-35)). This means that for every seven booters that have ToS, one does not. Now applying this concept to the non-booters (*i.e.*, 12/(60-12)), we found that for every 0.25 non-booter that has ToS, one non-booter does not. Then, dividing the odds ratio of booters by non-booters, we conclude that the final odds ratio is equal to 28 (7/0.25). The higher the odds ratio, the higher the relevance of a given characteristic for booters when compared to non-booters. Table 2.10 presents the odds-ratio for each of the 15 characteristics in our trainer dataset (928 URLs).

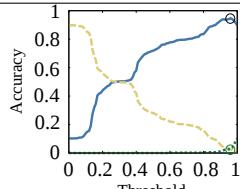
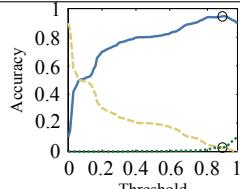
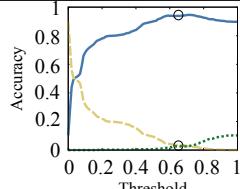
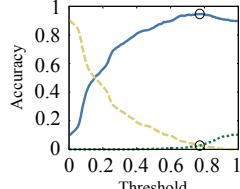
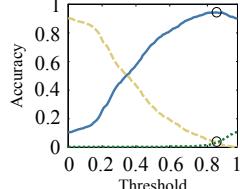
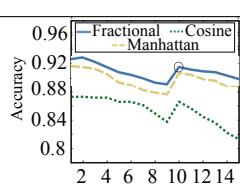
Table 2.10 shows the odds-ratio and respective normalised values for each of the 15 characteristics. We used the normalised odds-ratios as weights to multiply by the “perfect” booter characteristics (\vec{p}). We then repeated all the experiments described in the previous section (*i.e.*, for five distance metrics, k-NN and Naive Bayes). Table 2.11 summarises our findings.

Table 2.10: Odds-ratio of the 15 characteristics using a dataset of 928 URLs (113 booters and 815 non-booters), order by the highest values.

| # | Description | Odds Ratio | Normalized |
|----|------------------------|------------|------------|
| | | | Odds Ratio |
| 1 | Number of pages | 40.97 | 1.00 |
| 2 | Outbound hyperlinks | 22.83 | 0.56 |
| 3 | Domain age | 22.19 | 0.54 |
| 4 | Page rank | 20.93 | 0.51 |
| 5 | Content size | 12.26 | 0.30 |
| 6 | DPS subscription | 9.07 | 0.22 |
| 7 | URL length | 7.00 | 0.17 |
| 8 | WHOIS private | 5.98 | 0.15 |
| 9 | URL type | 6.00 | 0.15 |
| 10 | Domain expiration time | 5.77 | 0.14 |
| 11 | Depth level | 5.03 | 0.12 |
| 12 | Content dictionary | 3.00 | 0.07 |
| 13 | Login-form depth level | 2.92 | 0.07 |
| 14 | Resolver indication | 1.39 | 0.03 |
| 15 | Terms of services page | 1.13 | 0.03 |

Weighted approaches achieve better results in all the cases tested (Table 2.5, Table 2.6 and Table 2.9). For example, the Cosine distance in the unweighted approach reaches for the optimal threshold (0.78) for CAR, FPer and FNer at, respectively, 0.914, 0.049 and 0.037. For the weighted approach, the optimal threshold (0.95) is reached with 0.944, 0.022 and 0.034. To further improve the classification accuracy, we used a second weighted approach: machine learning.

Table 2.11: Results for booter classification using a weighted approach, added to the previous best values achieved via un-weighted approaches.

| Classification Approach | Graph | Tshld | CAR | FP_{er} | FN_{er} |
|-------------------------|---|-------|---------------------|----------------|----------------|
| Cosine |  | 0.95 | 0.944 | 0.022 | 0.034 |
| | | 0.78 | 0.914 | 0.049 | 0.037 |
| SQR Euclidean |  | 0.89 | 0.940 | 0.022 | 0.039 |
| | | 0.66 | 0.908 | 0.030 | 0.062 |
| Euclidean |  | 0.71 | 0.933 | 0.017 | 0.049 |
| | | 0.41 | 0.920 | 0.030 | 0.049 |
| Manhattan |  | 0.75 | 0.931 | 0.019 | 0.049 |
| | | 0.62 | 0.927 | 0.024 | 0.049 |
| Fractional |  | 0.86 | 0.923 | 0.022 | 0.056 |
| | | 0.82 | 0.914 | 0.024 | 0.062 |
| Naive Bayes | — | — | 0.918 ↑ 0.912 | 0.049 0.056 | 0.032 0.032 |
| k-NN |  | 10 | 0.914 | 0.060 | 0.026 |
| | | 10 | 0.910 | 0.073 | 0.017 |

We developed a machine learning algorithm, presented in Algorithm 1, to optimise the weights of booter characteristics. We only applied this machine learning algorithm to the Cosine distance approach because it showed the best CAR, FPer and FNer for weighted approaches.

Algorithm 1 Weight adaptability learning.

```

in:  $\vec{v}, \vec{p}, \vec{w}, CAR, FP_{er}, FN_{er}$ 
in:  $max\_interactions, \mu, \sigma$ 
out:  $\vec{w}', CAR', FP'_{er}, FN'_{er}$ 

1: procedure WEIGHADAPT2ABETTERCAR(input, output)
2:
3:   while ( $i < max\_interactions$ )||(CAR = 1) do
4:     for  $i = 0$  to  $len(\vec{w})$  do  $\vec{w}'[i] \leftarrow \vec{w}[i] * rand\_gauss(\mu; \sigma)$ 
5:     end for
6:      $CAR', FP'_{er}, FN'_{er} \leftarrow cosine\_dist(\vec{v}, \vec{p}, \vec{w}')$ 
7:     if ( then $CAR' > CAR$ )
8:        $CAR \leftarrow CAR'$ 
9:     end if
10:   end while
11: end procedure

```

The primary goal of the algorithm is to update weights towards an optimal weight vector. The inputs are the vectors \vec{v} and \vec{p} , which are multiplied by the vector \vec{w} to calculate the original CAR, FPer and FNer. The values of vector \vec{w} are the normalised values of the odds ratios in Table 2.2. During every algorithm interaction, the weights are multiplied by a random number within a Gaussian function with mean $\mu = 0.5$ and standard deviation $\sigma = 0.5$, generating a new weight vector \vec{w}' (line 4). The values of μ and σ are such as to force the new vector \vec{w}' stays in the interval $[0, 1]$, which is the interval of the original weights \vec{w} . After generating \vec{w}' , the new values of CAR, FPer and FNer are based on Cosine distance (line 6). Then CAR assumes CAR' if a better value is found. The algorithm runs until CAR achieves the highest possible number (*i.e.*, 1) or until the maximum number of interactions is reached. We decided to run one thousand times, expecting the best value to be achieved before this value. In the 824 iterations of the algorithm, we obtained the following optimal weights:

$$\vec{w}' = [1, 0.4, 0.3, 0.47, 0.21, 0.32, 0.17, 0.19, 0.16, 0.18, 0.13, 0.1, 0.04, 0.04, 0.03]$$

where the order of elements follows the order of the 15 characteristics in Table 2.2, that is, the first element of the vector \vec{w}' corresponds to the weight of the number of pages and the last item to the ToS page. The overall conclusion

of this chapter is that the Cosine distance metric is the best approach to classifying booter websites, based on the 15 characteristics defined in section 2.3. Using the optimal weights vector \vec{w}^* , and threshold of 0.95, the Cosine distance achieves a classification accuracy of 95.5%. The source codes of the methods presented in this section, including the machine learning algorithm, are publicly available at <https://github.com/jjsantanna/booter-black-List/tree/master/Classifier>.

2.5 Booter List Usage

In the previous sections we have presented a methodology for generating a comprehensive and accurate list of booter websites. We use this methodology to periodically create updated lists of booters which are openly shared at the Booter Blacklist initiative: <http://booterblacklist.com>. The log history shows that the Booter Blacklist website has been accessed from more than 3000 distinct IP addresses worldwide and downloaded around 700 times between June 2016 and March 2017. Some organisations openly declare that they use the Booter Blacklist to monitor accesses and DNS resolutions related to booter domain names. Among these organisations are the University of Twente in the Netherlands and several NRENs (National Research and Education Networks) such as SURFnet in the Netherlands, CEDIA in Ecuador and CESNET in the Czech Republic.

In this section we analyse data related to (attempted) accesses to booters. The dataset provided by SURFnet consists of DNS requests (*i.e.*, Q(Q)) originated from within the networks they manage and DNS responses (*i.e.*, Q(R), R(ANS), R(ADD) and R(AUT)) related to domain names listed on the Booter Blacklist. However, our analysis focuses on the overall behaviour

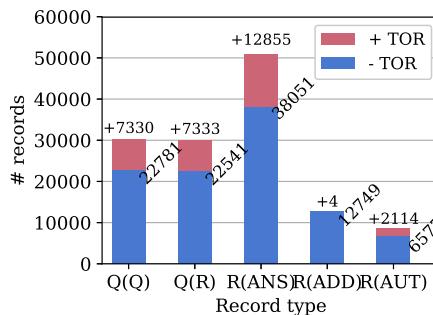


Figure 2.2: SURFnet data.

of SURFnet clients who access booters and, therefore, we only look at DNS requests. The data has been sent to us weekly since June 2015 and SURFnet anonymise the IP addresses of their clients with a SHA-256 encryption algorithm. We analysed a total of 646 days' worth of monitoring data (from 19 June 2015 to 27 March 2017), containing 132335 records and 605 distinct users (*i.e.*, IP addresses). The data provided to us by SURFnet cannot be made public, however, our source codes for data analysis are openly available at https://github.com/jjsantanna/booterblacklist_use_cases.

Figure 2.2 shows the shares of record types in our dataset. This figure also shows that two IP addresses, which are TOR nodes, generated 7330 queries, which is around 24% of all queries $Q(Q)$ in our dataset. Since we cannot determine how many SURFnet users actually accessed booters via these two TOR nodes, we removed them from our analysis. We also removed those users showing outlier behaviours; for example, we removed a single user who tried to access “ddos.cit” 2360 times in a period of 3 months, as well as two other users who performed some sort of scan across variations of “ipstresser.co”. For the remaining data, we divided the DNS requests into quarters: Q1, January–March; Q2, April–June; Q3, July–September and Q4, October–December.

The top graph in Figure 2.3 shows the number of requests to booter domain names from SURFnet clients. The lowest number of queries was seen during the third quarter in both 2015 and 2016, which was probably caused by the vacation period (SURFnet is a NREN). Another observation is that the overall number of queries has decreased from Q3/2015 to Q1/2017. A possible explanation is the reduction in the number of “new users” for booters, *i.e.*, IP

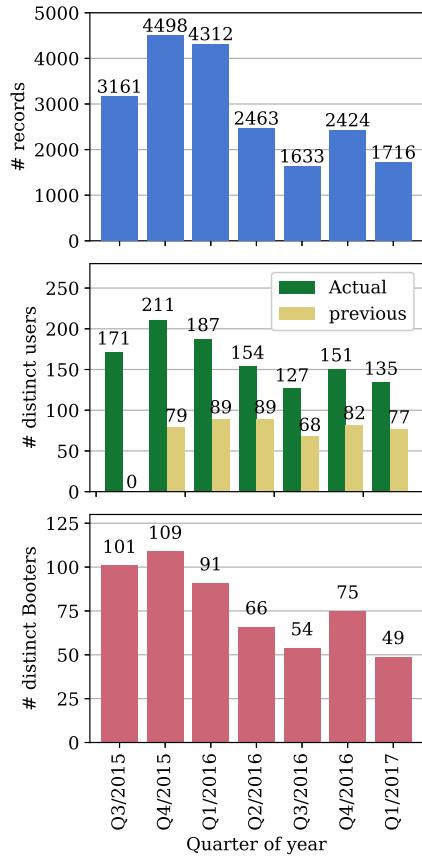


Figure 2.3: Analysis per quarter of year.

addresses accessing booters for the first time. The centre graph in Figure 2.3 shows that while the number of recurrent users (yellow columns) remains roughly the same—with small fluctuations between 68 and 89 users across quarters—the total number of users (green columns) has decreased over time, indicating a reduction in the number of IP addresses accessing booters for the first time.

Another thing that may have contributed to the reduced number of accesses to booters is the reduced number of available booters, as the bottom graph of Figure 2.3 shows. (There is a clear relation between the numbers in the top and bottom graphs of Figure 2.3). The Booter Blacklist used by SURFnet lists 435 booters (of which 115 are currently online). Based on the numbers of distinct accessed booters in the bottom graph, we can conclude that users only access a fraction of the available booters.

Figure 2.4 shows, for each quarter, the cumulative distribution of the number of queries that booters received. We can clearly see that the “long tail” of the distribution decreases over time, with Q1/2016 being the longest tail. Although the tail is reduced, the booters at the tail are mostly the same, namely booter.xyz and mostwantedhf.info. The reduced number of accesses to booters is also visible in Figure 2.4: while 80% of booters received 37 queries or less in Q3/2015, in Q1/2016 80% of booters had 62 queries or less each, and in Q1/2017 the same percentage had only 27 queries or less each.

Figure 2.5 shows the top ten most accessed booters for each quarter. Many

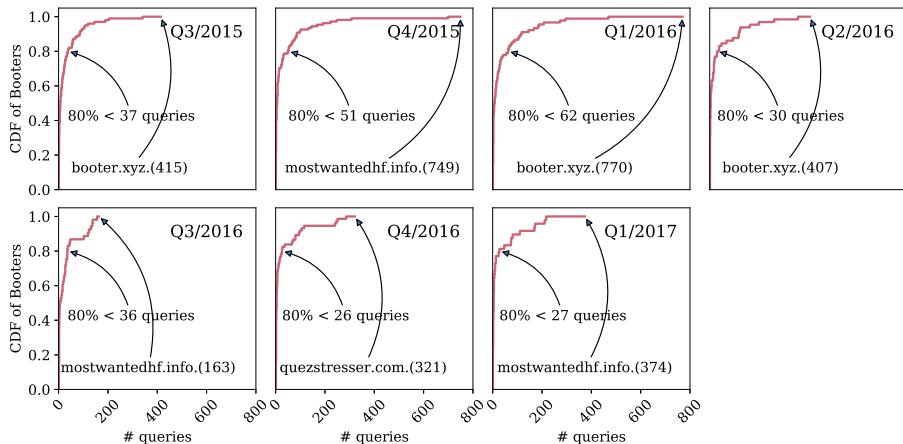


Figure 2.4: CDF of queries to distinct booters per quarter of year (using the same scale).

booters are consistently present in the top ten: booter.xyz, mostwantedhf.info, quezstresser.com, ipstresser.com and vbooter.org. Other booters have appeared, and remained, in the top ten more recently (*e.g.*, ragebooter.net and networkstresser.com). Note that of all the booters ever listed in the quarterly top ten, only one, inboot.me, is not available anymore. It disappeared from our logs after Q3/2016. Additionally, to the best of our knowledge, none of these booters have undergone mitigation actions.

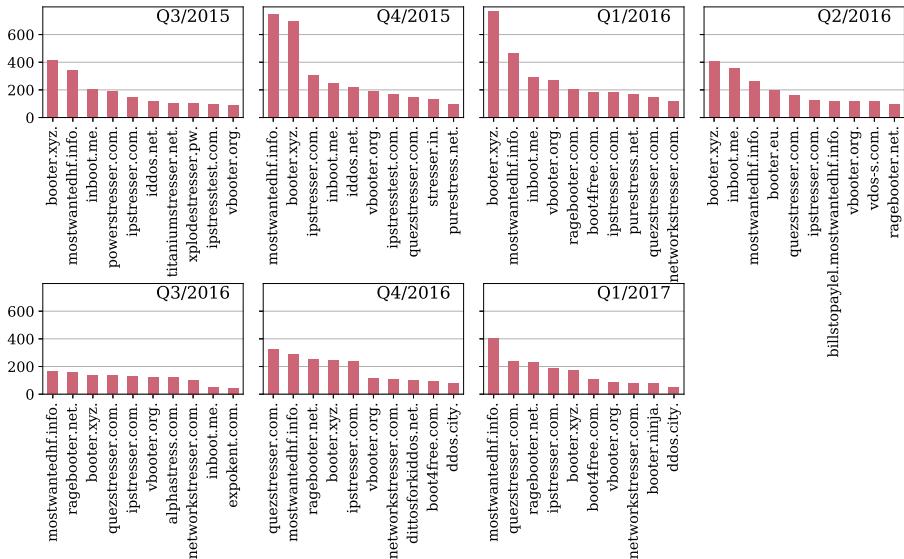


Figure 2.5: Top 10 most accessed booters per quarter of year (using same scale).

Figure 2.6 shows the cumulative distribution of requests to booters by distinct SURFnet users (IP addresses). The distribution “long tail” observed in all quarters except Q3/2016 shows that a few users perform many more requests to booter domain names than others. The median of number of requests is quite stable across quarters, with half of users generating up to 5 requests each. We believe that those users accessing booters only a few times are simply curious. Those at the tail of the distribution, however, are likely to be regular clients of “services” provided by booters.

Figure 2.7 shows the number of queries to booter domain names as sent by the 38 users who accessed booters during the week between 19th March 2017 and 25th March 2017. In this figure, each symbol represents an individual user (without the need for individual identification, the dataset is anonymised). Some

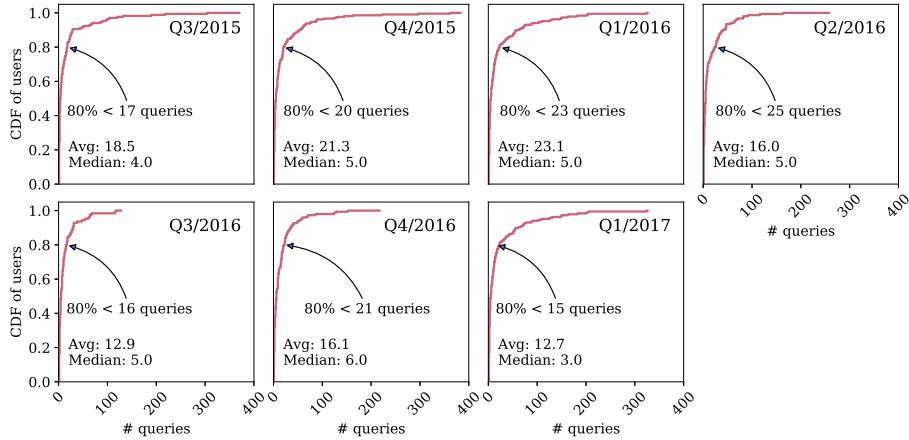


Figure 2.6: CDF of queries performed by users.

of these users (cyan hexagon, cyan triangle and purple pentagon) accessed the sites more than the most common median of accesses shown in Figure 2.4 (*i.e.*, 5 accesses). This analysis clearly identifies those users who should perhaps be closely monitored, to prevent them becoming cybercriminals and, consequently, to reduce the number of DDoS occurrences.

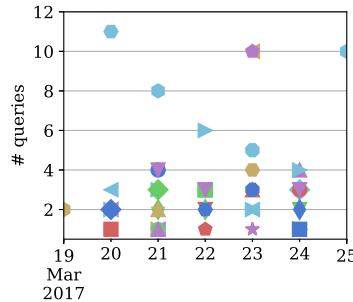


Figure 2.7: Number of access of SURFnet users to booters.

The analysis we present in this section helps to identify popular booters, which are likely to be those launching most DDoS attacks, and therefore sets a priority order for mitigation actions. Analysis of users' behaviour concerning accesses to booters provides supporting information for organisations, such

as SURFnet, to take preventive actions. Such actions are in line with a EUROPOL operation named Tarpit, which aims to raise awareness about the illegal character of DDoS attacks and booters [26].

The Dutch police took part in the Tarpit operation and approached users who performed DDoS attacks from booters [67]. The list of booters used in our work helps the Dutch police to identify users who accessed booters and such information could be further correlated with (leaked) booter databases containing records of users who hired DDoS attacks. As expected, not all accesses to booters result in launching of attacks. For example, as reported in [49], while online and operational, the vdos-s.com booter launched more than 170 thousand attacks hired by its users; however, none of the 67 SURFnet users that accessed this booter had a registered purchase in the leaked database from vdos.com.

2.6 Concluding Remarks

Our goal in this chapter was to create a comprehensive list of booter websites to answer the question of how broad the booter phenomenon is and enable an in-depth investigation of the phenomenon. We defined three requirements, which meant our approach must: (1) be automatic, (2) collect a comprehensive list of booters and (3) be accurate. In section 2.2, we identified relevant information sources to find booters in the public, deep and dark Web. We also identified a representative set of keywords to search for booters in these information sources. Then, in section 2.3, based on both the literature and our observations, we identified 15 characteristics that define a typical booter website. After this, in section 2.4, this set of features was used to determine the best classification algorithm from seven well-known website classification approaches. Finally, we enhanced the accuracy of the studied classification methods with weights and a machine learning algorithm. In the end, we met our accuracy requirement by achieving 95.5% classification accuracy.

By connecting the elements that use information to achieve comprehensiveness (Crawler) and accuracy (Scraper and Classifier) we met our first requirement, to have an automatic approach. For reproducibility, the source code of our entire methodology is available at <https://github.com/jjsantanna/booter-black-List>. Reflecting on our methodology, we identified the following issues:

- **On the comprehensiveness:** the booter phenomenon is a moving target, and the site owners can change their set of keywords and the places where they advertise their services (source of information). We do not consider this volatile behaviour to be a problem. The set of keywords

depends on the frequency of words from booter landing page websites. If booter websites change, our source code will automatically (re)define the new set of keywords. If booters modify the place where they advertise their services, our source code can be easily changed to include, crawl and scrape new sources of information.

- **On the accuracy:** more classification methods can be used to improve our classification approach. However, we are satisfied with our current results. For example, by using the Cosine distance, the 15 characteristics set and the weight vector generated by our machine learning algorithm we were able to classify more than 10,000 websites with three false positives and two false negatives. If anyone would like to try another classification algorithm, our source code can easily be extended.
- **On the ethical implications:** some people may question the moral impact of generating a booter list and potentially facilitating access to booters. However, booters are available on the public Internet to facilitate users finding them and using their services. All the sources of information that we use to find booter websites are public. In addition to which, our methodology actually mimics a customer search. Anyone would therefore be able to retrieve a list of booters manually. Our contribution is to make the collection of the list more comprehensive, accurate and automatic.

Our scientific contribution in this chapter is a methodology to find an extensive list of booter websites. The outcome of our methodology enabled a broader investigation of the booter phenomenon, not restricted to us but available to anyone. To enhance the impact of our solution, we use our methodology to keep an updated public repository with a booter list (<http://booterblacklist.com>).

Between June 2016 and March 2017, there were more than 3000 accesses of the list from distinct users worldwide. More than 700 people/organisations have downloaded our booter list. One of those organisations is the Dutch NREN, SURFnet, which uses our list to monitor their users. In section 2.5, we analyse and report SURFnet observations and additional usage of our list for booter mitigation.

We wish to encourage initiatives such as that of SURFnet and support large-scale operations to mitigate booters and their businesses. For example: action by Paypal on breaking the payment link between booters and their customers, causing the number of attacks from booters to reduce [41]; the operation resulting in the prosecution of booter owners convicted of cyber-crimes [90]; and the operation leading to the prosecution of booter customers [26].

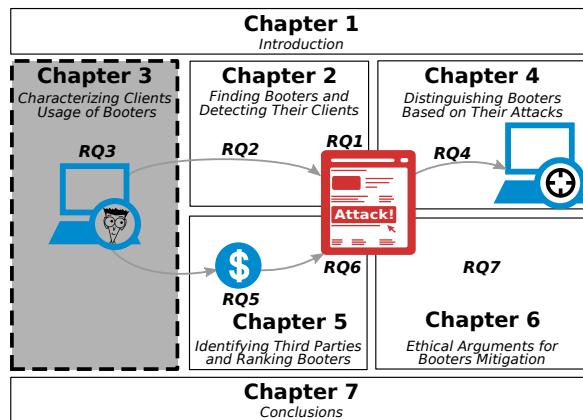
“Society prepares the crime, the criminal commits it.”

—HENRY THOMAS BUCKLE,
IN: THE MISCELLANEOUS AND POSTHUMOUS
WORKS OF HENRY THOMAS BUCKLE, 1872

CHAPTER 3

Characterizing Clients Usage of Booters

In the previous chapter, we detected booter clients via passive network measurements (given a list of booters). Although that type of monitoring reveals the access of clients to booter Websites, it does not disclose, for example, how clients use booter services. In this chapter, we aim at answer **how do clients use booter services? (RQ3)**. To address this question, we propose a semi-automated analysis methodology that can be applied to any booter database that contains clients' information. We use our methodology to analyse fifteen different booter databases.



The organization of this chapter is as follows:

- In Section 3.1, we describe why booter owners store information of their clients;
- In Section 3.2, we manually analyse 23 booter databases to generate a generic database schema;
- In Section 3.3, we define a set of relevant information to be automatic analysed;
- In Section 3.4, we investigate the consistency and completeness of booter databases;
- In Section 3.5, we present the comparative results of 15 booter databases analysed using our methodology;
- In Section 3.6, we express our considerations and highlight our contributions.

3.1 Client Records in Booter Databases

For booter owners, as it is for any other type of service, it is convenient to keep records of clients. While it helps on managing clients' rights, who paid for service and who should be allowed to hire attacks, it is also convenient for keeping pieces of evidence to blame their clients on the responsibility for attacks. Booters usually have terms of service (ToS) that emphasize that the usage of their services (*i.e.*, DDoS attacks) is the responsibility of their clients.

The main evidence that booter owners have against their clients is their operational database, which contains the record of clients who access their booter website and hire attacks against third party systems. Therefore, booter databases are essential for helping to identify clients that ordered attacks. For example, an Europol operation [26] and other two operations by the British National Crime Agency (NCA) [64, 65] ended with a prosecution of booter clients based on databases provided by booter owners (in court cases). There are around twenty cases of legal actions (*e.g.*, arrested, charged and prosecution) of booter clients and owners in the United Kingdom, listed in the Cambridge Computer Crime Database (<http://www.cl.cam.ac.uk/~ah793/ccccd.html>), which more than half of these cases were supported by information in booter databases.

In addition to the case where booter owners provide their database in a court case, for example, in exchange for reduced penalty, there are three other situations where these databases become available. The first occurs when a computer containing a booter database is seized from a crime scene. In the second, security specialists hack and openly share booter databases to help the security community understanding the damage caused by booter and motivate mitigation initiatives [47, 49]. In the third case, booter owners hack each other's database and “anonymously” share it in public websites such as pastebin.com, leakforums.net, and bitleak.net. The reason for this public sharing is market competition, *i.e.*, degrading the reputation of booter competitors to get more clients.

A surprising fact, related to booter databases, is that very often they become publicly available to anyone in the Internet. However, challenges reduce the use of these databases in legal actions against booter clients and owners. Hacked and publicly shared booter databases are difficult to have their consistency and completeness verified. Besides, these hacked databases are ethically and legally questionable for being used, for example, in legal court case. Another challenge is that booter databases contain a large number of records, which implies that it is very time consuming to find the information that can lead to concrete actions. Furthermore, different booters have different database schemas, making it difficult for a generic approach to address all different databases.

The goal of this chapter is to facilitate the automated analysis of any booter database, making easier to reveal the characteristics of booter clients. To validate our automated methodology, we analyse fifteen booter databases and compare the results of our analysis. In addition, we also investigate booter owners and the infrastructure used to perform attacks, within booter databases.

3.2 Methodology and Our Database Schema

In this section, we describe the three steps that define our methodology for semi-automated analysis of booter databases, depicted in Figure 3.1. We also present an additional functionality that our methodology enables (query interface), depicted as dashed boxes in Figure 3.1.

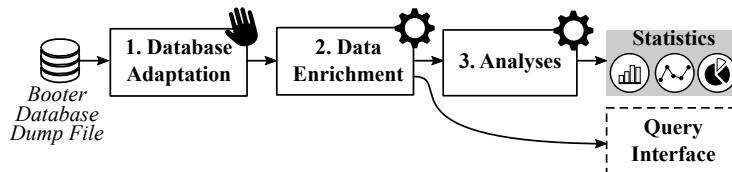


Figure 3.1: Steps of our methodology.

In the first step, the names of tables and columns of the input booter database are adapted using a generic database schema. This step is crucial for our automated methodology because it normalizes the terms utilized in the input file. Then, we enrich the input data with external databases, such as IP geolocation and Autonomous Systems (AS). Finally, using the enriched data, we perform a set of predefined analysis described in section 3.3. In addition to our analysis, we propose a query interface for dynamic queries on the data, which facilitates to find specific information, such as attacks performed by a specific booter client.

In the remaining of this section, we manually investigate the most frequent content of 23 booter databases aiming to define a generic database schema. In Appendix A, we list the URL from where we retrieved each respective database. To avoid ethical issues on the usage of these databases, we rely on the same approach described by Karami and McCoy [40] that is based on omitting all kinds of personal information, such as email addresses and user names, even when these details are known.

In Table 3.1, for each analysed booter database, we present the total number of tables and the number of table names that appeared in multiple databases ('same tables'). Table 3.1 also show the most common table names and identify

whether a booter database contains such exact table name (✓) or contains a similar content within a different table name (!).

Table 3.1: Summary of tables in 23 different booter databases dumps.

| # | Alias | # tables | # same tables | Most common table names | | | | | | | | | | | | |
|----|------------------|----------|---------------|-------------------------|------|--------|-----------|----------|-------|----|-----------|---------|----------|------------|-----------|-----------|
| | | | | users | logs | iplogs | loginlogs | payments | plans | fe | blacklist | gateway | settings | postshells | getshells | slowloris |
| 1 | 212booter | 16 | 10 | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | | | | |
| 2 | superstresser | 15 | 10 | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | | | | |
| 3 | nullbooter | 14 | 10 | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | | | | |
| 4 | stealthstresser | 14 | 10 | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | | | | |
| 5 | flashstresser | 13 | 10 | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | | | | |
| 6 | notoriousbooter | 12 | 10 | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | | | | |
| 7 | hazardstresser | 18 | 10 | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ! | ! | |
| 8 | stealthstresser2 | 12 | 10 | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | | | |
| 9 | pokeboot | 11 | 10 | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | | | | ✓ |
| 10 | nationalstresser | 11 | 9 | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | | | | |
| 11 | vaporizedbooter | 11 | 8 | ✓ | ✓ | | | ✓ | ! | | ! | ✓ | ✓ | ✓ | ✓ | |
| 12 | pandabooter | 11 | 8 | ✓ | ✓ | | | ✓ | ! | | ! | ✓ | ✓ | ✓ | ✓ | |
| 13 | pandabooter2 | 11 | 8 | ✓ | ✓ | | | ✓ | ! | | ! | ✓ | ✓ | ✓ | ✓ | |
| 14 | legionbooter2 | 11 | 8 | ✓ | ✓ | | | ✓ | ! | | ! | ✓ | ✓ | ✓ | ✓ | |
| 15 | galaxybooter | 11 | 8 | ✓ | ✓ | | | ✓ | ! | | ! | ✓ | ✓ | ✓ | ✓ | |
| 16 | xrhostbooter | 28 | 6 | ✓ | ✓ | ! | | ! | ! | ! | | | ✓ | ✓ | ✓ | |
| 17 | bootertw | 18 | 5 | ✓ | ! | ✓ | ! | ! | ! | ! | | | ✓ | | | ✓ |
| 18 | legionbooter | 18 | 4 | ✓ | ! | ! | ! | ! | ✓ | ! | ! | | ✓ | | ! | |
| 19 | vstresser | 8 | 4 | ✓ | ✓ | ! | ! | ! | ✓ | | ✓ | | | | | |
| 20 | urgentbooter | 30 | 3 | ✓ | ✓ | ! | | ! | ! | ! | | | | | | |
| 21 | panicstresser | 12 | 2 | ✓ | ! | | | | ! | ! | | ! | | | | |
| 22 | jaysbooter | 1 | 1 | ✓ | | | | | | | | | | | | |
| 23 | vddos | 1 | 1 | ✓ | | | | | | | | | | | | |

The most important observation based on Table 3.1 is that many tables appeared in various booter databases with the same name or similar content. This observation enables the creation of a generic database schema, which is the outcome of this section. Several other table names are not presented in Table 3.1. For example, the table name “news” appeared in 20 booter databases, the table name “tickets” in 7, “api” in 6, “emailtemplates” and “skype_api” in 5. The other names appeared in less than three databases. Although these tables are part of booter databases, they are not considered for being automated analyzed given our pre-defined list of aspects to be analysed (discussed in section 3.4).

By ordering booter databases on the number of same tables names, in Table 3.1, we observe some clusters of similar databases, such as the ones in lines 1–6, 7–10 and 11–15. There are two explanations for this clustering behaviour. First, it is known that some booters owners use copies of source codes easily found on the Internet to create their booter [80]. Another explanation is that there are multiple booters that have the same owner, and also same database structure [81]. In section 3.5, we compare the records from booter databases to reveal whether they are copies of each other (*i.e.*, exact same record).

A third observation, based on Table 3.1, is that booter databases have a distinct number of tables, ranging from 1, *i.e.*, *jaysbooter* and *vddos* (lines 22 and 23) to 30, *i.e.*, *urgentbooter* (line 20). On average, booters have 13 tables, and at least one table with the exact same name as one of the top-14 most common table names. There are two reasons for the difference on the number of tables. The first and most obvious is that booter owners store different pieces of information, with different levels of details. The second reason is that sometimes only few tables are hacked and publicly made available; for example, when a SQL injection retrieves only a set of records from a table.

Towards defining a generic booter database schema, we analyze in details the content (columns) of each one of the 14 most shared and relevant table names. For each table, we identify three aspects. The first aspect is the description of their main characteristic, which is crucial for further facilitating the differentiation of tables. Second, identify the most common column names found in actual booter databases. Third, determine replacements and adaptations for table and column names aiming consistency and disambiguation. Overall, our replacement strategy has three rules. First, we reuse common table and column names if such names are single words in lowercase (*e.g.*, *username*), instead of words with underscore mark (*e.g.*, *user_name*) or capital letters (*e.g.*, *UserName*). Our second rule is to use table names in the plural, while columns in the singular. Our last rule is to use unique column names to avoid ambiguity.

1. Table users:

- Main characteristic: provide a unique identifier for users;
- Most frequent tables: from the 20 database dumps that contain users information, the most common columns are: ID (found in 20 databases), *username* (in 20), *password* (in 19), *email* (in 17), *expire* (in 15) and *membership* (in 14);
- Replacements: “*id*” instead of “*ID*”, “*useremail*” instead of “*email*” and “*plan*” instead of “*membership*”.

2. Table attacks:

- Main characteristic: store information of attacks launched by users;
- Most frequent tables: all the 7 databases that contain this type of information are composed of the columns username, target, time, port and date;
- Replacements: originally this table is called “logs”. However, we notice that there is a confusion in actual booter databases related to the content of tables “logs”, “iplogs” and “loginlogs”. Therefore, aiming to avoid misunderstands we re-named the table “logs” to “attacks”. In addition, we changed the column name “time” to “duration” to prevent ambiguation with the date in which the attack was performed.

3. Table logins:

- Main characteristic: contain the IP address of users that logged in the booter website;
- Most frequent tables: userID, logged and date;
- Replacements: to merge any table that contains login information, such as “iplogs” and “loginlogs”. Then to change columns “userID” for “userid” and “logged” for “userip”.

4. Table payments:

- Main characteristic: reveal the records related to user payments;
- Most frequent tables: from 13 booter databases contain the information related to payments the most common columns are: user (in 13 databases), paid (in 13), plan (in 12), tid (in 12), date (in 12), and email (in 11);
- Replacements: “username” instead of “user”, “amountpaid” instead of “paid” and “useremail” instead of “email”.

5. Table plans:

- Main characteristic: reveal the features of packages of attacks subscription available;
- Most frequent tables: from 10 booter databases that contain information related to plans, the most common columns are: maximum boot time also called as mbt (found in 10 databases), length (in 10), price (in 10), name (in 9), unit (in 9), description (in 7) and concurrents (in 5);

- Replacements: “maxbottime” instead of “mbt”, “concurrency” instead of “concurrents”, “plandescr” instead of “description”, and “planname” instead of “name”. We did not find usage for the table “length” and “unit”. Therefore, we removed these two columns from our generic database schema.

6. Table friendsenemies:

- Main characteristic: this table contains the information of users flagged as either friend or enemy;
- Most frequent tables: from 11 databases containing information related to either friends or enemies, the most common columns are: ip (in 11 databases), note (in 11), userid (in 9) and type (in 8);
- Replacements: originally, the table was called “fe”. However, we consider easier to understand its meaning, *i.e.*, friends and enemies, instead of the acronym.

7. Table blacklist:

- Main characteristic: this table contain IP addresses to which booter owners deny performing attacks against. Examples of commonly blacklisted organizations are FBI, CIA and HackerForums;
- Most frequent tables: from 9 databases containing information related to blacklisted organizations, the most common columns are: ip (in 9 databases) and note (in 8);
- Replacements: we have not identified reasons to modify this content.

8. Table gateways:

- Main characteristic: reveal the email address of people that profits from booter services;
- Most frequent tables: all 17 databases that contain this type of information had only the column “email”;
- Replacements: we have not identified reasons to modify this content.

9. Table settings:

- Main characteristic: this table shows main information of the booter website;

- Most frequent tables: from 7 databases that contain such type of information, the most common columns are site_url (in 6 databases), site_name (in 5) and site_email (in 4);
- Replacements: “url” instead of ‘site_url’, “sitename” instead of “site_name” and “siteemail” instead of “site_email”.

10. Table webshell:

- Main characteristic: reveals URLs of (mis)used machines that de facto perform attacks;
- Most frequent tables: all 6 databases that contain this type of information had the columns URL, online and lastChecked;
- Replacements: originally there are three tables with WebShells: “postshells”, “getshell” and “slowloris”. These tables have the exactly same columns. The main difference is that the first performs HTTP POST based attack, the second HTTP GET based attacks and the last SLOWLORIS attacks. Therefore, we merge theses three tables related to WebShells and added a column called “attacktype” to differentiate them. We also replace “url” instead of “URL”, “status” instead of “online” and “lastchecked” instead of “lastChecked”.

11. Table servers:

- Main characteristic: instead of WebShells URLs, this table reveals IP addresses of servers that perform attacks;
- Most frequent tables: there is only one booter that added the information of their servers. Although they present 15 different columns, we consider the IP address enough to analyze further booter databases;
- Replacements: no replacement was necessary.

At this point, instead of the 14 tables identified in Table 3.1, we describe only 11. The reason for that is that some tables were merged, such as tables “iplogs” and “loginlogs”, merged into table “login”; and “postshells”, “getshell” and “slowloris”, merged into table “webshell”. The summary of our list with 11 tables with our proposed nomenclature is depicted as our generic booter database schema in Figure 3.2. We differentiate tables related to booter clients (lighter background color) and tables related to the operation of the booter service/website (darker background color). We also highlight (in bold text) the columns that most represent each table. For example, “userid” and “amountpaid” are crucial to identify the “payments” table.

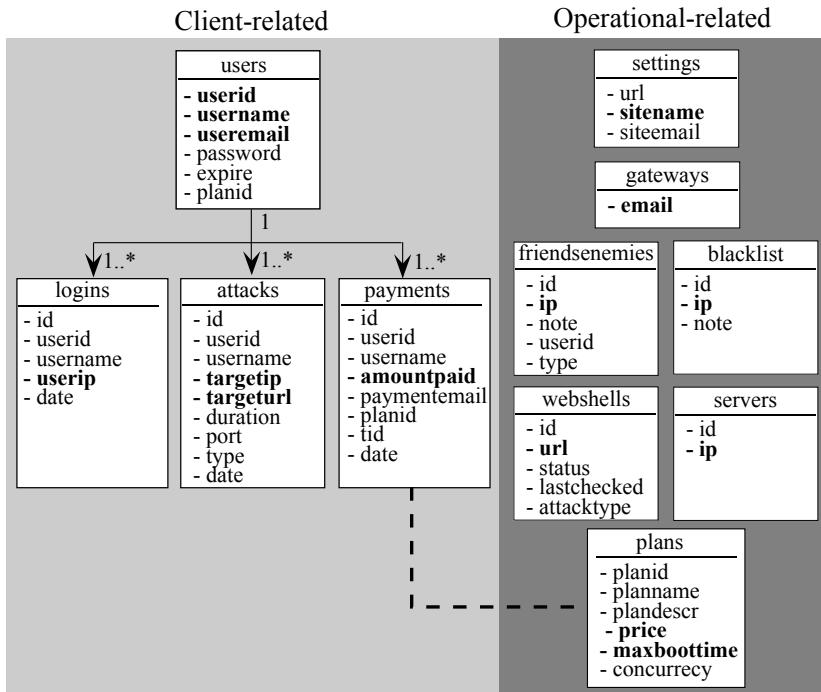


Figure 3.2: Generic booter database schema.

3.3 Aspects to be Automatically Analysed

The goal of this section is to define aspects of booter databases that should be automatically analysed by our methodology. Our approach in defining those aspects is based on observing existing booter database investigations. We consider that existing work already addresses a broad and relevant range of aspects. Besides observing the aspects already investigated, we identify which tables in the generic database schema could be used to address those aspects.

Historically, the first public booter database investigated was the one from *twbooter*, originally accessible at booter.tw (offline since September 2013). This booter database became public after a series of DDoS attacks targeting a computer security blog (krebsonsecurity.com) [47] and the Ars Technica news website (arstechnica.com) [28]. Vijayj [97] is one of the first to analyse *twbooter* database information. His analysis focussed on the identity of booter clients that attacked krebsonsecurity.com. This investigation covered the geolocation of clients, the different IP addresses used to access *twbooter* website

and the relationship between those IP addresses and the TOR network. Vijay [97] also reveals the amount, duration and types of consecutive attacks against krebsonsecurity.com. A similar investigation, by Rever Security [77], also investigated the geolocation of servers used by twbooter to perform attacks against krebsonsecurity.com.

Not restricted to the attacks against krebsonsecurity.com, Schwarz [86] provides a global view of attacks within the twbooter database. He analyses the duration and type of attacks, the geolocation and the types of targets, along with geolocation of twbooter clients. Similar to Schwarz [86] work, there are two investigations performed by Karami and McCoy [39, 40], which provides an even more comprehensive understanding of the twbooter database. In another work by the same authors [41], they performed a similarly thorough analysis applied to three others booter databases: Asylum Stresser (asylumstresser.com), Lizard Stresser (lizardstresser.su) and vDoS Stresser (vdos-s.com). Finally, there is the analysis performed by Bukac et al. [17], which analysed 31 booter databases, covering most of the aspects from investigations previously described. None of the existing works focus on facilitating the analysis of booter databases, but they present statistics about what is found.

In summary, investigations of booter databases focus on: characteristics of clients, attacks, targets and the infrastructure used to perform attacks. There are also some considerations on booter clients analysis regarding the usage of anonymization services, such as proxies, Virtual Private Networks (VPN) services and TOR' network. Therefore, we included all the aspects addressed in the previous work to be analysed in an automated fashion.

3.4 Booter Database Consistency

We now look into the consistency of 15 booter databases. This analysis is important for improving the confidence of the records in booter databases. The databases analysed in this section contain at least 100 logs of attacks; which we consider as a representative number to analyse and compare characteristics.

Most of booter databases available on the Internet come from websites that anonymously share hacked information, such as pastebin.com, leakforums.net and bitleak.net. The unknown source of these databases implies uncertainty on the completeness and the accuracy of the data. It is known that booter owners delete some records from their database to avoid leaving pieces of evidence. There is a testimony of a booter owner [47], from asylumstresser.com, which affirms that “some attack records were regularly deleted from the database”. This affirmation explains why all the fifteen booters that we analyse in the next section have some gaps of information (*i.e.*, sequential identifiers

missing).

Our approach to check the consistency of a booter database is based on comparing three pieces of information: (A) the first attack record in the booter database, (B) the domain name registration date of the booter and (C) the first passive DNS record observed from a global measurement initiative. We collect the domain registration date at `whois.domaintools.com`, although other sources of domain name information can be used. Our source of passive DNS is DNSDB, which collects global scale passive DNS measurements since 2010.

We first check whether the first attack record date is posterior to both, the domain registration (A-B) and the first DNS record (A-C). Then, we investigate whether the domain name registration date (B) is very closer or before the time of the first passive DNS observed (C) ($C - B \approx 0$ or $C - B > 0$). The reason is that when someone wants to register a domain name, the availability of the domain name is checked, which is likely to be observed by DNSDB. Otherwise, the observation by DNSDB will happen at some point after the domain creation. We present our observation in Table 3.8 for each booter that we automatically analyse.

We observed in Table 3.8 that pandabooter, vaporizebooter, vstresser and hardstresser (rows 1, 2, 6 and 7, respectively) have their first attacks dating months before the domain name registration of their website and the first passive DNS record observed by DNSBD. A possible explanation is that some booter owners started their booter websites based on the source code and the database of other previously hacked booters, without flushing the records in the database. Although these databases are very inconsistent, we still analyse them in the next section, for example, to investigate whether they share the exact same information. Another motivation to perform this type of investigation is that booters in rows 1 & 2 and 13 & 14 have the same date of the first attack, which for us is an odd coincidence.

There are three booters, namely legionbooter, notoriousbooter and superstresser (rows 9, 13 and 15), that have their first attacks dating many days after the domain name registration of their website, and the first passive DNS record observed by DNSBD. A possible explanation is that, although those booters were active for months or even years, the owners removed all records that preceded a specific date. Although we consider this finding suspicious, it is supported by the fact that booter owners remove records from their databases.

Regarding the domain registration date and the first passive DNS records (C-B) we observed that most of the booters are consistent because their date is very close. Although twbooter and vaporizebooter have more than four months of difference, it is still possible to happen. One explanation is that the vantage points of DNSDB do not cover the region where these booters were created and initially used.

Table 3.2: Dates related to booters for checking their database consistency (DD/MM/YY).

| # | Alias | A. First attack | B. Domain registr. | C. First seen | A-B | A-C | C-B |
|----|------------------|-----------------------|--------------------------|---------------------|------|------|-----|
| 1 | pandabooter | 05/09/11 | 09/05/12 | 08/05/12 | -247 | -246 | -1 |
| 2 | vaporizebooter | 05/09/11 | 22/05/12 | 29/09/12 | -260 | -390 | 130 |
| 3 | panicstresser | 30/07/12 | 12/07/12 | 11/07/12 | 18 | 19 | -1 |
| 4 | pokeboot | 10/12/12 | 17/10/12 | 16/10/12 | 54 | 55 | -1 |
| 5 | twbooter | 24/01/13 | 13/07/12 | 14/12/12 | 195 | 41 | 154 |
| 6 | vstresser | 01/02/13 | 06/05/13 | 07/05/13 | -93 | -94 | 1 |
| 7 | hazardstresser | 15/03/13 | 27/04/13 | 11/04/13 | -43 | -27 | -16 |
| 8 | xrshellbooter | 19/03/12 | 27/10/11 | 26/10/11 | 145 | 146 | -1 |
| 9 | legionbooter | 04/04/13 | 30/08/11 | 29/08/11 | 583 | 584 | -1 |
| 10 | flashstresser | 24/05/13 | 25/04/13 | 16/04/13 | 30 | 39 | -9 |
| 11 | 212booter | 04/07/13 | 24/04/13 | 24/04/13 | 72 | 72 | 0 |
| 12 | nationalstresser | 05/09/13 | 03/05/13 | 01/05/13 | 125 | 127 | -2 |
| 13 | notoriousbooter | 20/01/14 | 25/04/13 | 09/04/13 | 271 | 287 | -16 |
| 14 | nullboot | 20/01/14 | 18/11/13 | 20/11/13 | 64 | 62 | 2 |
| 15 | superstresser | 12/02/14 | 04/04/13 | 02/04/13 | 314 | 316 | -2 |

In general, booter databases are all very suspicious. The only way to guarantee the veracity of the records in booter databases is by comparing the timestamp of records (within the database) with third party measurements. For example, the measurements presented in the previous chapter could guarantee that some clients access the booter website. Another example: companies that suffered and (intentionally) measured booter attacks could guarantee the veracity of (at least) the records related to the suffered attacks. In the end of the next section, we present some cases where the information of booter database were sustained by third party measurements.

3.5 Automated Analysis

After applying the generic database schema (defined in section 3.2) and discussing their consistency (in section 3.4), in this section, we analyse the outcome of our automated analysis applied to 15 booter databases. Afterwards, we analyse and compare the content of the booter databases. The analysis covers three parts: client characteristics, attacks and the infrastructure used by booters to perform attacks. For each part we use information from different tables, for example, tables ‘clients’, ‘logins’ and ‘payments’ are used to analyse the

characteristics of booter clients; Table ‘attacks’ is used to analyse characteristics of attacks and to compare with the ‘infrastructure’ table. This latter comparison reveals the relationship between the attacks offered and the infrastructure used, which is the last part of our analysis. In addition to the automated analyses, we present a case study of the query interface that we proposed in our methodology.

3.5.1 Clients, Customers and Attackers

Booter clients can play three different roles: simple clients, customers and attackers. A simple client is a person that created an account on a booter website. A customer is a client who purchased services on booters and an attacker is a person that performed attacks. Table 3.3 shows, for each database, the number of clients, customers and attacks; and the total amount of money paid by booter customers. Five booter databases (lines 11–15) did not have payment records and, therefore, the number of customers was not investigated.

It was expected that clients that performed attacks were the ones that paid for it (customers). Therefore, we expected that attackers are contained in customers, which are contained in clients (*i.e.*, $\text{attackers} \subseteq \text{customers} \subseteq \text{clients}$). We observed in Table 3.3 that there are more clients than customers and attacks, meaning that many clients are just attracted to take a look at what booters offer. Only a few attackers and customers cannot be observed in the list of clients. However, except superstresser and panicstresser (lines 2 and 4), all the others booters have more attackers than customers. There are two hypotheses for this: either the payment records were removed or some clients had the allowance of booter owners to perform attacks for free. Although it is possible that some clients had the privilege to order attacks without paying, it is more likely that some payment records were removed. The best example that indicates this hypothesis was found in the superstresser database, where the payment table contains only records from middle 2013, but the attacks records start at the beginning of 2012.

Another observation that emphasizes the removal of records is the number of customers that launched attacks (column customer attacker), which is smaller than the overall number of customers. Although it is entirely possible that some customers never launched any attack, we believe this to be unlikely in such a large proportion. Therefore, we suspect that some clients that launched attacks had their payment records removed, preventing them to be traced.

We also observed that the number of clients and customers are not proportional to the amount of money profited. For example, bootertw had far fewer clients (312) and customers (80) than superstresser (2236 clients and 684 customers) but earned 1.6 more. The reasoning is that the clients of bootertw paid more amount of money than superstresser (showed in Figure 3.4).

Table 3.3: Booter clients and total amount of money paid.

| # | Alias | Client | Customer | Attacker | Customer attacker | Profits [\$US] |
|----|------------------|--------|----------|----------|-------------------|----------------|
| 1 | bootertw | 312 | 80 | 277 | 26 | 8127.00 |
| 2 | superstresser | 2236 | 684 | 163 | 135 | 4885.00 |
| 3 | pokeboot | 464 | 96 | 194 | 94 | 2181.00 |
| 4 | panicstresser | 235 | 57 | 25 | 11 | 615.00 |
| 5 | 212-booter | 140 | 28 | 57 | 24 | 509.00 |
| 6 | nationalstresser | 1892 | 46 | 81 | - | 497.00 |
| 7 | hazardstresser | 79 | 28 | 24 | - | 307.00 |
| 8 | flashstresser | 749 | 13 | 66 | 7 | 165.00 |
| 9 | notoriousbooter | 81 | 2 | 22 | - | 37.00 |
| 10 | nullboot | 118 | 6 | 26 | - | 31.00 |
| 11 | vaporizebooter | 17 | - | 13 | - | - |
| 12 | legionbooter | 23113 | - | 691 | - | - |
| 13 | xrshellbooter | 374 | - | 27 | - | - |
| 14 | vstresser | 10 | - | 6 | - | - |
| 15 | pandabooter | 33 | - | 15 | - | - |

Following, we present our findings on the frequency of payments performed by customers (Figure 3.3) and the amount of money paid (Figure 3.4).

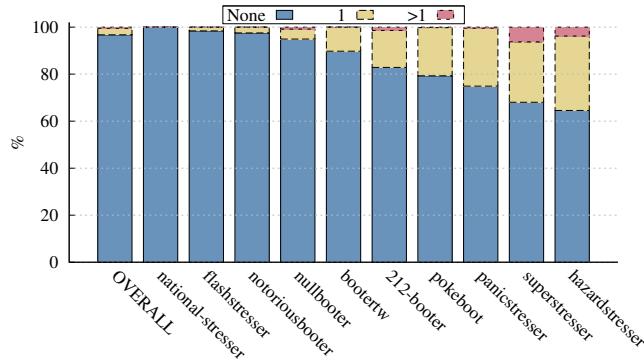


Figure 3.3: Payments per client.

Figure 3.3 shows, for each booter separately and on the overall of all surveyed databases, how many times clients purchased attack plans from booters. As already observed in Table 3.3, the number of clients that did not pay (simple clients) is far larger than the number of customers. We observed that the clients rarely paid more than once to perform attacks. Note that when a client pays

to a booter, the client is allowed to perform as many attacks as possible within a period, called as expiration date. Therefore, we can infer that either booter clients are satisfied to perform only a single set of attacks, or the payments records have been removed too often to observe clients buying again at a later moment.

In Figure 3.4, by analysing how much booter customers paid (instead of clients in general), we observed that, except bootertw, the majority of customers paid for the cheapest attack plan available. Overall, more than 50% of all customers paid \$US5.00 or less to perform attacks. Through this finding, we highlight that although booters offer several prices to perform attacks, the cheapest ones are the ones clients choose most. In this analysis, notoriousbooter and nullbooter results are less representative because their number of customers is very small, 2 and 6 respectively.

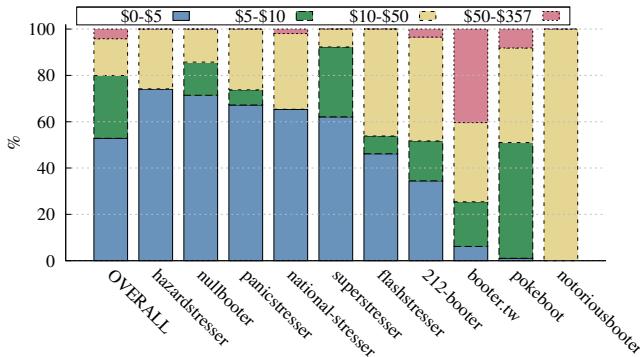


Figure 3.4: Amount of money paid.

Another observation in Figure 3.4 is that only four booters offer attacks that were purchased for more than \$US50. For those booters, we also found interesting customers outliers. For example, bootertw earned \$US3,000 in 1 day with only 2 customers. Our hypothesis to explain this is that it is most probably due to failure in the payments that lead to a record but that it is not a real benefit for the booter (two email addresses related to payment accounts repeatedly tried to pay the same amount during a short period). In the extreme, we found 6 clients in bootertw database that appeared in the payment table but in fact paid nothing (\$US0.00) to perform attacks. This booter has promotional campaigns allowing clients to perform attacks for free.

By looking at both graphs, Figure 3.3 and Figure 3.4, we observed that customers paid different amounts of money depending on the booter. Following, we investigate the IP addresses used by clients to access the booter website

(*e.g.*, TOR, VPN and proxy) and related this information to the number of attacks they performed.

3.5.2 IP addresses, TOR, VPN and proxy usage

We now analyse how clients access booter websites, even when these clients try to hide their activities behind TOR, VPN and proxy. This analysis was motivated by Karami and McCoy [39] that mentioned booter clients using VPN and proxies. Our methodology is divided into three steps. First, we observe the number of unique IP addresses used by each client, assuming that clients that log in using several IP addresses are likely using VPNs or proxies.

Second, we focus on analysing clients that use TOR, which is a software that helps clients to avoid traffic analysis and censorship. For this investigation, we correlate the IP addresses of client logins with the list of exit nodes of TOR. Exit nodes are the visible part of the TOR network to the Internet. TOR exports the list of exit nodes every hour since February 2010 and this list can change through time. In our last step, we analyse the countries resolved from the IP addresses that clients access booter websites. This analysis should emphasize the clients that are using an intermediary service to hide their actual IP address. Our assumption is that a client is unlikely to be in multiple countries around the world in a short period.

Table 3.4 summarizes the number of clients that access booters using a different number of IP addresses. We also show the total number of clients that have IP address records stored in the database to compare with the total number of clients (Table 3.3). Table 3.4 also shows the number of attacks performed by clients that accessed booters via a single IP address, two, three or more. At last, we show the total number of attacks related to IP addresses.

In general, the number of clients that access booters with a single IP address represent the largest fraction. For example, superstresser have almost 2.6 more clients that access booters with one IP address (379) than the ones that access with three (141). A possible reason could be that most clients (53%) accessed booters only once in the analysed datasets. Surprisingly clients that access Booters with three or more IP addresses generated far more attacks than the first group. For instance, clients that access superstresser with three IP addresses launched almost 10x more attacks than the clients that access them with a single IP address. It means that clients that perform more attacks are more likely to take precautions in hiding their real IP address.

The exception of having clients related to several IP addresses performing more attacks is legionbooter clients. Note that for this booter, the number of attacks related to one IP is far larger than related to three IPs. However, the number of clients missing is representative, calculated via the difference between

Table 3.4: Client IP address(es) and attacks.

| Alias | 1 IPs | 2 IPs | ≥ 3 IPs | Total clients related IPs | Attack related 1 IP | Attack related 2 IPs | Attack related ≥ 3 IPs | Total attacks |
|------------------|----------|----------|-----------------|------------------------------------|---------------------------|----------------------------|--------------------------------------|------------------|
| bootertw | 64 | 34 | 191 | 289 | 3,392 | 1,055 | 44,382 | 48,829 |
| legionbooter | 230 | 31 | 29 | 290 | 11,053 | 4,020 | 2,879 | 17,952 |
| pokeboot | 52 | 19 | 40 | 111 | 503 | 493 | 1,091 | 2,087 |
| superstresser | 379 | 79 | 141 | 599 | 494 | 832 | 4,005 | 5,331 |
| nationalstresser | 9 | 5 | 3 | 17 | 64 | 29 | 8 | 101 |
| 212booter | 99 | 21 | 33 | 153 | 364 | 802 | 815 | 1,981 |
| notoriousbooter | 2 | 0 | 2 | 4 | 29 | 0 | 0 | 29 |
| flashstresser | 522 | 66 | 114 | 702 | 136 | 89 | 346 | 571 |
| nullboot | 5 | 0 | 2 | 7 | 146 | 0 | 56 | 202 |
| hazardstresser | 14 | 7 | 8 | 29 | 21 | 0 | 13 | 34 |
| vaporizebooter | 17 | - | - | 17 | 660 | - | - | - |
| xrshellbooter | 1* | - | - | 1* | 619 | - | - | - |
| panicstresser | 230 | - | - | 230 | 0 | - | - | - |
| pandabooter | 28 | - | - | 28 | 76 | - | - | - |
| vstresser | - | - | - | - | - | - | - | - |

the total number of clients (23133) and the clients related to IP addresses (290) (Table 3.3). In addition, the difference between the total number of attacks, and the attacks related to IP addresses is also representative. It indicates that possibly a large number of records were removed. Slightly different from legionbooter, the booters bootertw, 212-booter and flashstresser are very consistent, meaning that (even if some records were removed) the number of clients and attacks matches with the relation of client IP addresses and the number of attacks related to those IP addresses.

After analysing the relation between attacks and ways that clients access booters, we analyse how many of them used TOR. Table 3.5 shows booter clients that used TOR to login in a booter website. Besides, the table indicates the number of logins made by those clients and how many among those logins were realized by using TOR. Finally, we describe the number of attacks launched by those clients and the ratio between attacks and number of clients.

Surprisingly, only four booters were found having clients that used TOR; and the number of clients that used TOR to access booters is insignificant (20) in comparison to the total number of clients. However, those clients performed far more attacks than the average of clients. For example, the clients that access bootertw via TOR (7) performed 513 attacks each on average (opposed to 31

Table 3.5: Details per Booter about clients using TOR.

| Alias | Total clients | TOR clients | Login TOR clients* | Login with TOR | Attack TOR clients | Attack /TOR clients |
|---------------|---------------|-------------|--------------------|----------------|--------------------|---------------------|
| superstresser | 2236 | 8 | 256 | 82 | 205 | 25,6 |
| bootertw | 312 | 7 | 9128 | 146 | 3595 | 513,6 |
| flashstresser | 749 | 4 | 255 | 13 | 24 | 6 |
| xrshellbooter | 374 | 1 | 1 | 1 | 26 | 26 |

attacks for all clients with login information). Note that for xrshellbooter we had only the last login IP address for all clients. However, the clients of this booter performed 26 attacks, which we consider a reasonable number of attacks.

Note that the number of clients that access booters with more than one IP address (Table 3.4) is far bigger than the number of clients that used TOR. This observation means that some clients are taking precautions to hide their actual IP address, but also that there are still clients using several IP addresses without using TOR. This fact does not exclude that these clients could also use others services, such as VPN or proxies.

We refined the analysis of counting of IP addresses per client by resolving the addresses to retrieve the Autonomous System (AS) and the corresponding country. Reasoning in term of countries allows solving the issue of a client accessing a booter from different legitimate locations (*e.g.*, home, school and work) and also the dynamic IP allocation from their ISP.

Table 3.6 shows our findings regarding geolocation of IP addresses per countries and attacks. As expected, the number of client IP addresses related to one country is bigger than the other two options. It happens because the number of logins related to one IP address (Table 3.4) is also the biggest one. However, most of the attacks are linked to a single country, not to ≥ 3 countries (except for booter.tw). It means that our assumption that clients that access booters with different IP addresses are using VPN and proxies is not completely true. This happens because a client can access from various locations where there is Internet connection (*e.g.*, home and school). However, it is still clear that some clients access Booters via VPN and proxies, because their access originated from multiple countries. Furthermore, the most important finding showed in Table 3.6 is that the proportion of attacks by these clients, logging in from several countries, is significantly higher than those logging in from a single one.

Note that the analysis on countries is much less significant than on the one IP addresses. If we consider that clients logging in from a single country are not using any VPN or proxy, then it could mean that many customers are ordering attacks without trying to hide their actual IP address. This confirms our

Table 3.6: Client countries related to attacks.

| Alias | Access from country | | | Attack from country | | |
|------------------|----------------------------|----------|----------------------------|----------------------------|----------|----------------------------|
| | 1 | 2 | ≥ 3 | 1 | 2 | ≥ 3 |
| bootertw | 172 | 57 | 60 | 14,131 | 6,336 | 28,362 |
| legionbooter | 272 | 14 | 4 | 15,923 | 1,171 | 858 |
| pokeboot | 81 | 18 | 12 | 1,280 | 510 | 297 |
| superstresser | 499 | 56 | 30 | 2,396 | 1,735 | 1,196 |
| nationalstresser | 13 | 2 | 2 | 93 | - | 8 |
| 212-booter | 146 | 7 | 0 | 1,475 | 506 | - |
| notoriousbooter | 3 | 0 | 1 | 29 | - | - |
| flashstresser | 631 | 49 | 22 | 338 | 150 | 83 |
| nullboot | 6 | 0 | 1 | 153 | - | 49 |
| hazardstresser | 20 | 6 | 3 | 22 | 12 | - |
| vaporizebooter | 17 | - | - | 660 | - | - |
| xrshellbooter | - | - | - | - | - | - |
| panicstresser | 230 | - | - | 0 | - | - |
| vstresser | - | - | - | - | - | - |
| pandabooter | 23 | - | - | 76 | - | - |

previous hypothesis that minorities of customers are performing more attacks and are taking precautions. Most importantly, this also shows that a significant number of customers are performing fewer attacks but apparently without taking any precautions. This observation means that their identity could be more easily discovered and, therefore, they could suffer legal actions.

3.5.3 Same clients in different booters

We also investigate whether booter clients emails are found in various booter databases. Table 3.7 show our findings. It is not surprising that some clients have accounts in different Booters. However, we did not expect to find more than one hundred clients between legionbooter and flashstresser, or between superstresser and legionbooter (again). In both cases, we exclude the possibility of the re-use (or copy) of another database because the timestamps when clients subscribed booters are different. Note that clients in both cases are distinct; otherwise, the intersection between flashstresser and superstresser would be more representative. Therefore, a possible explanation can be on the period of booter activity. While legionbooter is the oldest booter online (from the 15 booters that we analyse), since 2011, it shared clients with the booter flashstresser (active before superstresser). Then, more recently, superstresser became active, flashstresser went offline and legionbooter stayed online.

We extended the analysis of client emails in different booters to all types

Table 3.7: The same client email account in different booters.

| Alias | 212 | pok | boo | fla | haz | leg | not | nul | pan | sup | vap | xrs |
|-----------------|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|
| 212-booter | - | 1 | | 1 | | 31 | 1 | | | 2 | | |
| pokeboot | 1 | - | 2 | 6 | 4 | 24 | | | 4 | 12 | 1 | 1 |
| bootertw | | 2 | - | | | 2 | | | | 2 | | |
| flashstresser | 1 | 6 | | - | 2 | 102 | 1 | 1 | | 16 | 1 | |
| hazardstresser | | 4 | | 2 | - | 5 | 1 | | | 3 | | 1 |
| legionbooter | 32 | 24 | 2 | 113 | 5 | - | 5 | 8 | 8 | 167 | 1 | 3 |
| notoriousbooter | 1 | | | 1 | 1 | 5 | - | 1 | | 9 | | |
| nullboot | | | | 1 | | 8 | 1 | - | | 16 | | 1 |
| panicstresser | | 4 | | | | 8 | | | | - | 4 | 3 |
| superstresser | 1 | 11 | 1 | 16 | 2 | 150 | 9 | 17 | 4 | - | | 4 |
| vaporizebooter | | 1 | | 1 | | 1 | | | 3 | | - | |
| xrshellbooter | | 1 | | | 1 | 3 | | 1 | 2 | 4 | | - |

of records. Then, we observed that pandabooter and vaporizebooter share 90 records (with the same timestamps), consisting of 28 attacks and 62 records that are related to the infrastructure used to perform attacks. This sharing of data means that at least one of the two booters reused a database from the other. We also observed that xrshellbooter have identical records with pandabooter (34 records) and vaporizedbooter (4 records), which are related to the infrastructure used to perform attacks. This finding does not necessarily implies that xrshellbooter is a copy of the other two booters; instead, it highlights that booters generally share a small number of resources to perform attacks.

3.5.4 Overall attack records

In this subsection, we characterize attacks ordered by booter clients to provide an overview of what kind of attacks have been the most frequent one. First, we provide an overall number of attacks and the time span of the data (the difference between the first date and the last date). Second, we analyse the types of attacks that were recorded in the databases. Then, we analyse how attacks were performed over time (*i.e.*, in a sequential or parallel way). Lastly, we study the duration of attacks.

By analysing the information on Table 3.8, we observed that the dataset span is not proportional to the number of attacks. For example, bootertw (line 1) has a smaller dataset span than vaporizebooter (line 8); however, the number of attacks is the opposite. It could mean that bootertw potentially had more popularity than vaporizebooter. Another explanation is that although vaporizebooter has a wider data span, it does not mean that they had their infrastructure always able to perform attacks.

Table 3.8: Overall attack numbers and data span.

| # | Alias | Attacks | Data span |
|----|------------------|---------|-----------|
| 1 | bootertw | 48844 | 403 |
| 2 | legionbooter | 38248 | 134 |
| 3 | pokeboot | 6915 | 83 |
| 4 | superstresser | 5565 | 36 |
| 5 | nationalstresser | 2756 | 93 |
| 6 | 212booter | 1993 | 57 |
| 7 | notoriousbooter | 879 | 99 |
| 8 | vaporizebooter | 725 | 971 |
| 9 | xrshellbooter | 629 | 41 |
| 10 | flashstresser | 580 | 32 |
| 11 | nullboot | 343 | 65 |
| 12 | panicstresser | 209 | 0 |
| 13 | hazardstresser | 173 | 88 |
| 14 | vstresser | 157 | 423 |
| 15 | pandabooter.com | 104 | 258 |

3.5.5 Attack types

To investigate the types of attacks mostly chosen by clients, we analyse records from the database attacks table. Since the names of attacks vary from one booter to another, we clustered the types of attacks into three categories. First, UDP-based attack is a category that includes Distributed Reflection DoS attacks, such as attacks based on CharGen, DNS and NTP, but also what they advertise as “UDP”, which can be a simple UDP flood or any attacks relying on UDP. TCP-based attacks are the second category of attacks. It includes SYN flood, “TCP” and “TCPAMP” attacks. Both categories (UDP and TCP-based) rely on protocols from the transport and network layer to perform attacks. The last category is the Application-layer attacks, which includes RUDY, SLOWLORIS, ARME and HTTP-based attacks (HEAD/POST/GET). There are others types of attacks found in databases but ignored in our analysis because they were too vague terms to be classified in a category, such as “SMALL1”, “test” and “FULL-POWERED-ATTACK”.

The rationale behind clustering attacks in categories is that we are aware that booters advertise some types of attacks but perform a more specific attack. For example, Karami and McCoy [39] shows that although bootertw advertises attacks as “UDP”, this booter performs DRDoS attacks based on DNS and CharGen. Note that clients are not aware of what booters perform (specific attack types). This information about the attack performed is restricted to booter owners and to the target that suffered the attack.

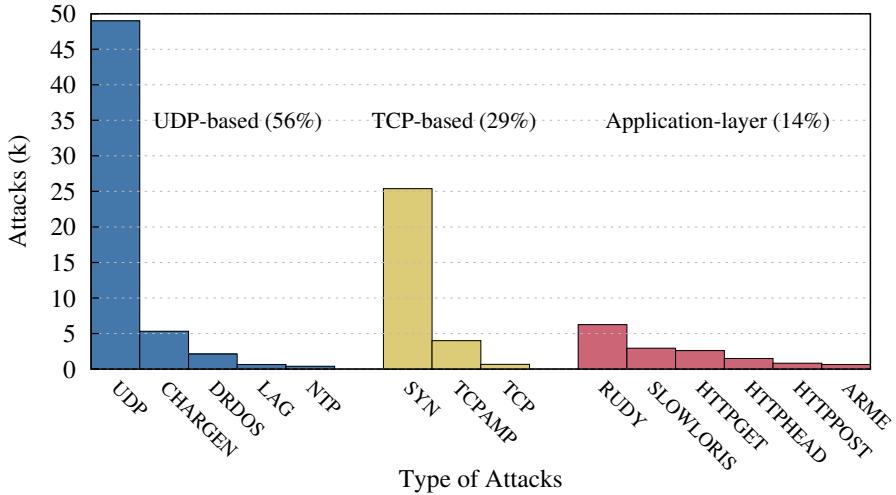


Figure 3.5: Attack types.

Figure 3.5 shows the sum of all attack types clustered per category. Note that the number of UDP-based attacks is almost double than others. It was not surprising that UDP and SYN flood attacks were the most popular among clients. We expect that booters follow the same trend of DDoS attacks reported by big network security companies [11], given that those companies blame booters for the majority of attacks observed.

3.5.6 Attacks usage

To understand how clients choose attacks and how often they perform them, we analyse the history of attacks for customers. Figure 3.6(a) and Figure 3.6(b) show the distributions of the overall and over time results, respectively. According to the overall analysis (Figure 3.6(a)), only 6% of customers performed a single attack. It means that clients do not buy a package of attacks to perform a single attack, but to keep attacking targets, alone the period of their package. Another finding is that 25% of clients perform more than 50 attacks, which is a representative number of attacks.

By analysing attacks over time (Figure 3.6(b)), we see that 38% of clients did not perform consecutive attacks. On average, clients performed only one attack per day. It is remarkable that 10% of clients that perform more than 13 attacks also performed consecutive attacks against the same target. We also

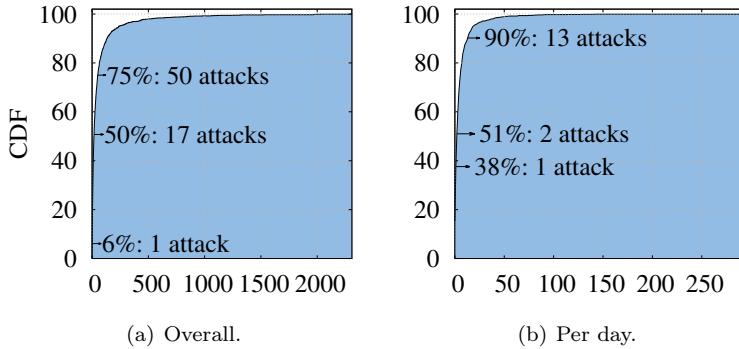


Figure 3.6: Attacks per client.

found some outliers, such as a single client that launched 2,308 attacks in ~ 8 days, which on average represents 294 attacks per day. One hypothesis is that this client shared his/her account with other clients to perform a campaign of attacks.

By analysing attacks on the same target by the same client (Figure 3.8(a)), we notice that 22% of the attacks targeted only once the same target by the same client. Consequently, it means that most of the clients performed two or more attacks against the same target. Note that 67% of the attacks have been launched at least 10 times on the same target by the same client.

If we analyse the probability of an attack to be re-launched on the same target less than 5 minutes after the end of the previous one, we can see that 58% of attacks have been at least repeated once more, as shown in Figure 3.7. The attacks seem to be chained to produce a longer one; 19% of all attacks are part of a DDoS campaign of at least 5 consecutive attacks. This behavior makes more sense when we analyse the duration of the attacks. As we can see on Figure 3.8(b), attacks are usually short. 70% of them last less than 10 minutes. An explanation for it is that the prices for short term attacks (less than 10 min) are lower than for longer attacks.

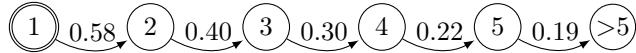


Figure 3.7: Probability for attacks to be relaunched less than 5 minutes later.

By analysing attacks we also notice that 32% of consecutive attacks have been launched in parallel. It indicates that new attacks against the same target started before the end of the current one. This observation means that most

of the time, customers are willing to deal as much damage as possible to their target and for a longer period.

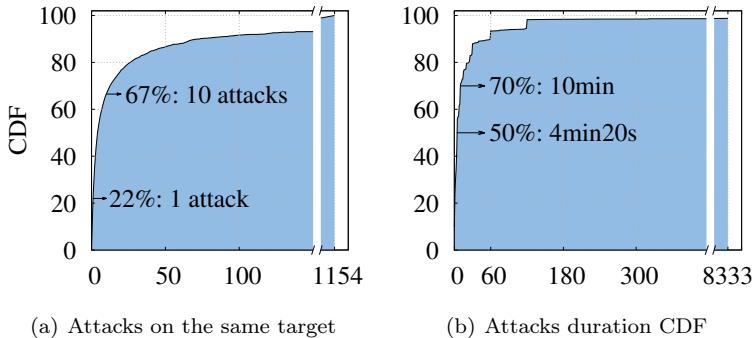


Figure 3.8: Cumulative distribution of attacks against a same target and the duration.

3.5.7 Attack infrastructure

Our last automate analysis on booter databases focus on to the infrastructures used to perform attacks. Although previous work describes the booter infrastructure as based on servers, by analysing records from the infrastructure database table we had an entirely different observation of that. Table 3.9 shows, for each booter, the number of web-shells and servers listed in the infrastructure database table, together with the types of attacks performed. Note that “UDP(+)” means that a booter performs UDP-based attacks including DRDoS.

Surprisingly, all Booters, except bootertw, had their infrastructure based on *Web-shells*, which is entirely different from what related works have concluded. Web-shells are scripts hosted on machines (compromised or not) that are accessed via HTTP/GET or HTTP/POST and expect parameters to launch attacks, such as the target’s IP address or URL, the duration of the attack, the destination port and (sometimes) the type of attack. For example, in <http://example.com/web-shell.php?host=yourwebsite.com&time=30&port=80> a Web-shell hosted in example.com will perform an attack against yourwebsite.com, during 30 seconds, on port 80.

Table 3.9: Booter infrastructure.

| Alias | Web-shell | | Servers | Attacks types |
|------------------|-----------|------|---------|------------------------|
| | GET | POST | | |
| bootertw | - | - | 15 | UDP, TCP, App-layer |
| superstresser | 2 | - | - | UDP(+), TCP, App-layer |
| notoriousbooter | 1 | - | - | UDP(+), TCP, App-layer |
| 212-booter | 1 | - | - | UDP(+), App-layer |
| flashstresser | 7 | - | - | UDP, TCP |
| panicstresser | 4 | - | - | - |
| hazardstresser | 2 | - | - | - |
| legionbooter | 16 | - | - | UDP(+), TCP, App-layer |
| vaporizebooter | 209 | - | - | - |
| pandabooter | 466 | 139 | - | - |
| xrshellbooter | 134 | 64 | - | - |
| pokeboot | - | - | - | UDP, TCP, App-layer |
| nullboot | - | - | - | UDP(+), TCP, App-layer |
| vstresser | - | - | - | UDP, TCP |
| nationalstresser | - | - | - | - |

By analysing the name of Web-shell scripts we notice that most of them are PHP scripts. An interesting observation, showed in Table 3.9, is that Web-shells (in theory) can cover all types of attacks (UDP, TCP and Application-layer attacks), such as stresser. However, according to Prolexic [71], DRDoS attacks cannot be covered by Web-shells (or at least no Web-shell has been found with these characteristics). That is not possible (yet) because to perform this type of attack, machines running Web-shells need to have a list of other services that will be mislead to perform attacks, such as DNS and NTP services. It means that only booters that have their infrastructure based on servers can be used to perform DRDoS attack. Consequently, it implies that notoriousbooter, stresser, 212-booter and legionbooter that advertise to perform DRDoS attacks should also have their infrastructure based on servers, but the URL to access it might be hard-coded, not in the database.

A last observation is that, although bootertw did not offer DRDoS attacks, it performed DRDoS attacks based on DNS and CharGen instead of pure UDP attacks. So, it makes even more sense bootertw infrastructure is based on servers, as presented by Karami and McCoy [39].

3.5.8 A use case of our querying interface

In addition to our automated analysis, after adapting and enriching the input booter database, our methodology creates an easy-to-use interface for querying

records. This interface has been set up to facilitate searching for specific records in the data. We used this interface to investigate whether we could find booter owners that were prosecuted. We used the nickname used by those booter owners on our querying interface. From this search, we found that all the booter owners launched hundreds of attacks against several target systems. This observation emphasizes that booter owners are among the clients that most performed attacks. It also stresses that booter databases contain valuable information to support legal action against people involved with booters.

3.6 Concluding Remarks

In this chapter, we investigated the clients' usage of booter websites. Our approach used for this investigation was based on the analysis of booter database information. We had a set of challenges for performing this investigation in an automated fashion, such as the difference of database schema used by different booters, the (in)consistency and (in)completeness of information in those databases and the lack of a comprehensive and relevant set of aspects to be automated. In section 3.2, we analyse 23 booter databases to define a generic database schema, which enables to adapt any booter database to be automatically analysed. In section 3.3, we surveyed existing investigations of booter databases to highlight that the combination of those analyses would be a suitable and relevant set of aspects to be automatically analysed. After that, in section 3.4, we evaluate the consistency of booter databases. Our findings, based on 15 booter databases, is that although booter owners removed many records, the remaining records could still be used to investigate the clients' use of booters. Although the dataset that we analysed is suspicious, we still consider extremely valuable for supporting legal action.

By overcoming the challenges to perform an automated analysis of booter databases, we apply our methodology to analyse 15 booters. In section 3.5 we compared our findings and highlighted our overall observation for all booters. The picture that emerges from the analyses is that booter clients have very distinct characteristics. On one side, we found the majority of clients accessing booters with a unique IP address, willing to pay less than \$US 10, performing attacks with less than 5 minutes duration and targeting a few URLs or IP addresses. On the other side, we also found a small number of harmful clients that hide their actual IP address via VPN and proxies, accessing booters using hundreds of IP addresses from dozens of countries, willing to pay several hundreds of Dollars to perform hundreds of attacks per day, often against the same target. We finally observed that booter owners were in this second group of clients that performed the majority of attacks.

Considering that most of the clients seem to take little precaution when accessing booters, we conclude that the usage of a list of booters, proposed in the previous chapter, would be a very effective measure to prevent users from accessing booters. This practice has been adopted by our partners at CERT SURFnet and the University of Twente and has successfully prevented several (potentially) clients from accessing booters. As a consequence of this practice, however, the number of clients that access booters via VPN or proxies (therefore from an unfiltered connection) might increase. From the same observation that most of clients are layman Internet users, we conclude that they can be easily identified (given they provide their email and their IP addresses), which enables law enforcement agencies to take actions against those booter clients. We have three considerations on this chapter:

- **On the ethical usage of leaked databases:** all booter databases analysed in this chapter were retrieved from public websites such as pastebin.com, leakforums.net and bitleak.net. However, it is known that those databases were hacked. To avoid any ethical issues on the analysis of those databases, we omitted all kind of personal information, such as email addresses and user names, even when these details were known (for example booter owners that faced prosecution). Besides of that, those databases served only as a ground truth to validate our generic database schema and the automated analysis proposed in this chapter.
- **On the completeness and consistency of booter databases:** our analysis indicates that booter databases are often incomplete and sometimes inconsistent and their content should, therefore, be used *cum grano salis*. For this reason, in several parts of our analysis, we were not able to provide a definitive explanation of our observations. Instead of it, we draw many hypotheses to encompass the uncertainty of those databases. However, we have also shown that booter databases are a valuable source of information about how booters are used in practice and offer valuable insights that can help to mitigate them.
- **On the comprehensiveness of our semi-automated analyses:** although the definition of a set of aspects to be automatically investigated restricts the potential of investigations, our set is composed of the most common and relevant aspects found in the literature. Besides, we made our methodology available at https://github.com/jjsantanna/booter_dbs_analyses to facilitate the addition of other investigations in a semi-automated fashion.

Our scientific contribution in this chapter is the definition of a generic database schema and the automated analysis of booter databases that is

enabled by the database schema. The outcome of our methodology reveals the characteristics of booter clients (the primary goal of this chapter) and also the infrastructure used by booters to perform those attacks. To enhance the impact of our methodology, we made publicly available, at https://github.com/jjsantanna/booter_dbs_analyses, the scripts used for our automated analysis currently applied to twenty-four booter databases (17-May-2017).

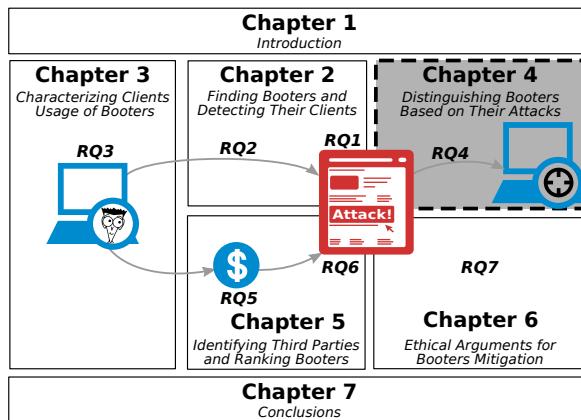
We wish to encourage law enforcement agencies to use our methodology and facilitate finding evidence to support legal actions against booter clients and owners.

“Do I not destroy my enemies when I make them my friends?”

—ABRAHAM LINCOLN
1855

Distinguishing Booters Based on Their Attacks

In the previous chapter, we investigated records in booter databases that indicate clients hiring attacks. To legally request access to these databases, enforcement agencies must first determine which booters were involved in attacks. To support enforcement agencies' actions, in this chapter we focus on answering the question of *do booters have distinct attack characteristics, and if so, what are these characteristics?* (RQ4). Our approach to addressing this question is based on measuring attacks performed by booters. We became a booter client and asked booters to perform attacks against a controlled environment. In addition to analysing whether booters can be differentiated by their attack characteristics, we also consider whether booters deliver what is advertised on their websites and reveal third-party services that (in)directly provide support to booter operations.



The organization of this chapter is as follows:

- In Section 4.1, we describe what is advertised in booter Websites;
- In Section 4.2, we present our methodology to measure booter attacks;
- In Section 4.3, we analyse the characteristics of booter attacks;
- In Section 4.4, we investigate third-party services that support booters operations;
- In Section 4.5, we provide our final considerations of this chapter.

4.1 What is Advertised on Booter Websites

In this section, before investigating the characteristics of booter attacks, we describe what is advertised on booter websites. This section is necessary to further compare what is offered and delivered by booters. In addition to using this information for comparison purposes, we provide details of information found on booter websites to identify possible further mitigation actions (*e.g.*, at a technical or economic level). Our approach to describing what is offered follows the order of users' observations while planning to purchase an attack.

There are four steps in hiring booter attacks. The first is to find a booter website. Popular search engines, such as Google and Bing, retrieve thousands of websites, when someone searches for the terms "booter" or "stresser". Not all websites retrieved by these search engines are actual booters, for example they may be blogs advertising or investigating booters (as described in chapter 2). However, the majority of websites on the first three pages of results (on search engines) are actual booters. This means finding a booter is straightforward for any Internet user.

After finding a booter, the second step in hiring an attack is creating a login account. The first page of any booter website (*i.e.*, the landing page) contains a login and a registration form for creating a new user account. Some booters also use the landing page to describe characteristics of the attacks they offer, for example the types of attacks and the maximum power they can provide. After login, the website usually shows some of the user's usage record (*e.g.*, previous attacks performed and the last IP addresses used to access the booter website) and some operational statistics for the booter (*e.g.*, the number of current attacks running and the number of users logged-in). In addition to the historical information and the booters' operation, there is often a menu panel with links to web pages such as "purchase", "hub" and "resolvers". In the following section, we describe what is provided on each of these webpages.

On the purchase page, booters present options for attack subscriptions. The synonyms used by booters for "attack subscription" are attack plans, attack package and membership plan. Each attack subscription is mainly composed of five pieces of information: (1) maximum duration of each single attack (*i.e.*, max boot time), (2) the maximum number of concurrent attacks, (3) the duration of the subscription (also called "length" and "expiration time"), (4) price and (5) the payment system that clients prefer to use to transfer the money for the subscription (*e.g.*, credit card, PayPal or Bitcoin). Less commonly, some booters provide the maximum data rate of the attacks (Mb/s) (although the data rate is not a parameter available for users to change). Choosing an attack subscription is the third step in hiring attacks. Clients are then redirected to the selected payment system website to pay. If Bitcoin or another cryptocurrency

system is chosen, instead of redirecting to another website the booter informs the user which cryptocurrency identifier (*i.e.*, wallet) the money should be transferred to.

After the booter owner receives confirmation from the payment system, the client is automatically allowed to hire attacks via the hub webpage accessed in the menu panel. In general, this allowance is automatic (via scripts). However, if the payment fails or does not work correctly, clients are invited to contact booter operators (owners) via a ticket system, chat box or Skype call. Then a booter operator will interact with the client and solve the problem.

The hub page of the booter website is where attacks are hired, which is the fourth and last step in hiring an attack. On the hub page, a web form is provided for the user to complete with the characteristics of the attack to be performed. This form is composed of four fields: (1) the IP address of the target; (2) the attack type (also called attack method); (3) the destination port of the target and (4) the attack duration. Booters provide an additional service accessible via the menu options, which is called “resolvers” and helps clients to identify the correct target IP address. The resolvers page provides services for finding the actual IP address of a target system. For example, the “Skype resolver” discovers the IP address related to a Skype account. There is also the “Steam resolver”, which finds the IP address of Steam user accounts. Steam is one of the biggest platforms for online gaming on the Internet. Booters are well known in the online gaming community, where players (DDoS) attack one another to gain an advantage in matches. There is also the “Cloudflare resolver”, which discovers the IP address hidden (for protection purposes) by Cloudflare, which is a company that offers DDoS protection services. In addition, the simplest resolver finds the IP address of a domain name.

Booters offer dozens of attack types, usually grouped based on network and transport layers and on the application layer. While attacks based on network and transport layers aim to overload the network capacity of the target system, attacks on the application layer aim to overload a specific limitation of a service (*e.g.*, the number of requests a web server can establish at the same time). Examples of attack types based on layers 4 and 5 are UDP, UDP-Lag, HOME, SYN, Spoofed SYN (SSYN), RSSYN, SSYN-FIN, XSYN, TCP amplification, ESSYN, Distributed and Reflected Denial of Service (DRDoS), CHARGEN, SSDP, DNS, NTP, SNMP, RIP, Dominate and Valve Source Engine (VSE). Examples of attacks based on layer 7 are: HTTP GET, HTTP POST, HTTP HEAD, R-U-Dead-Yet?(RUDY), Slowloris, Joomla Reflection, ARME and XMLRPC. Although it is technically possible for booters to have the ability to perform all these types of attack, our observations (in section 4.3) suggest that offering several types of attack attracts more attention from their clients.

In the remainder of this chapter, we focus on whether the advertised

characteristics of attacks are actually delivered. In addition, we investigate the infrastructure used by booters to perform attacks and discuss whether their attack infrastructures are distinct from each other. If this is the case, individual booters could be identified as responsible for attacks by looking at the attack characteristics.

4.2 Measuring Booter Attacks

To analyse the attack characteristics delivered by booters, and ensure that these attacks are actually launched by a particular booter (ground truth), our methodology relies on hiring and measuring booter attacks against a controlled environment (our network infrastructure). Although it is straightforward to find a booter website, hire its services and launch attacks (as described in the previous section), we found several challenges in measuring such attacks.

First, there are some ethical and legal issues. For example, to measure attacks we purchase booter services. By paying booters, we reinforce their market and operations, although this is not our intention. Another example of an ethical and legal challenge: there is a high chance that booters misuse infected systems to perform attacks (which is analysed in the next section). We therefore indirectly abuse these systems when taking our measurements. To be transparent about our work, we discussed our research intentions with operators from the Dutch NREN (SURFnet) and a public prosecutor. In response, supporting our research, SURFnet and the prosecutor advised us to add the following statement to any academic material that we publish: “we are aware that research of this nature may touch on, or cross, legal boundaries, but we are convinced that the results from this research will benefit future mitigation methods and thus help combat booters, both operationally as well as legally. In order to be transparent about our work, we have informed the office of the public prosecutor in the Netherlands about our intention to pursue this research.”(Appendix D)

Besides the ethical and legal challenges, we observed that booters do not provide an option to choose the attack strength (*i.e.*, data rate or the packet rate of the attack). This means that we cannot control this aspect of booter attacks and, consequently, we put the network infrastructure of third-parties (*e.g.*, networks connected to ours) at risk. Therefore, we must strategically place our measurement infrastructure to minimise the possibility of damaging third-party networks. To overcome this challenge we relied on the support of SURFnet, which facilitated measuring attacks directly at the Amsterdam Internet Exchange (AMS-IX). Using this approach, the network infrastructure is unlikely to be overloaded as AMS-IX is mainly a point where Internet Service

Providers (ISPs) connect to each other.

On 14th and 15th August 2013, the period of our measurements, there were 21 booter websites online (found using the methodology described in chapter 2). Among these booters, seven had a faulty payment system that did not allow us to subscribe to perform attacks. For each of the 14 remaining booters, presented in Table 4.1, we: (1) created an account, (2) purchased the cheapest attack subscription and (3) launched UDP-based DDoS attacks against our measurement infrastructure (at AMS-IX).

Table 4.1: Alias of 14 booters, their prices and their maximum attack rate.

| Alias | URL | Price[€] | Attack Rate [Gb/s] | Internet Archive |
|-------|---|----------|--------------------|------------------|
| B1 | http://destressbooter.com | 3.89 | 25 | ✓ |
| B2 | http://grimboot.com | 3.90 | 6 | ✓ |
| B3 | http://quantumbooter.net | 8.00 | 1.5 | ✓ |
| B4 | http://dejabooter.com | 3.89 | 10 | ✓ |
| B5 | http://booter.tw | 10.90 | - | ✓ |
| B6 | http://restricted-stresser.info | 1.95 | 5 | ✓ |
| B7 | http://anonymous-stresser.net | 3.12 | 5 | ✓ |
| B8 | http://rebel-security.com | 3.00 | 3 | ✓ |
| B9 | http://flashstresser.net | 3.89 | - | ✓ |
| B10 | http://olympusstresser.org | 4.90 | 3 | ✓ |
| B11 | http://ebooter.5gbfree.com | free | - | - |
| B12 | http://vdoss.net | 3.11 | - | ✓ |
| B13 | http://respawn.ca | 3.90 | 8 | ✓ |
| B14 | http://onionstresser.com | 3.90 | - | ✓ |

Table 4.1 presents the alias that we use to identify each booter in the remainder of this chapter. In total, we spent €58.35 subscribing to their services. At that time (August 2013), booters offered no more than 25 Gigabits per second (Gb/s) maximum traffic rate (booter B1), while nowadays they offer hundreds of Gb/s attacks. Only booter B3 is still available (checked on 6th July 2017). The other booters are offline and can still be observed at the Internet Archive (available at <https://archive.org/web>), which has more than twenty years of webpage history for the majority of pages on the Internet. The Internet Archive did not retrieve booter B11 because the URL to access their website was a subdomain of a web-hosting company (*i.e.*, .5gbfree.com).

Attacks from each booter were hired at different times, so that their respective attack traffic (rates) would not interfere with one another. Although booters offer several types of DDoS attacks, as described in the previous section, for our experiments we concentrated on volumetric attacks based on UDP,

because no service running on the target system is required (*e.g.*, web-server) and the only potential bottleneck is the network link capacity.

Placing our network equipment at AMS-IX, however, did not guarantee that our measurements were successful. Booters advertise dozens to hundreds of Gb/s of attack data, which is more than our measurement equipment can collect (originally 10 Gb/s). We must therefore find a way to compensate for loss in our measurements. In the following section, we provide more details about our measurement losses and describe an algorithm to overcome these losses.

4.2.1 Compensating DDoS attack traffic

SURFnet and their Computer Security Incident Response Teams (CSIRTs) were informed and actively collaborated by monitoring botter attacks (in addition to our measurements). When we compared the traffic collected by SURFnet with the traffic measured by our equipment (both located at the AMS-IX), the latter had a lower traffic rate, indicating that our equipment was overloaded during the measurement. By investigating artefacts in the data, we realised that both the PCIe-bus and RAID system of our monitoring system were overloaded during the attacks. Unfortunately, SURFnet only retains visual statistics rather than packet captures; this meant that we could not use their measurements to analyse the attacks in detail.

To be able to use our dataset, we developed an algorithm to compensate for lost traffic. Since we are looking solely at volumetric attacks, our algorithm assumes the attack traffic is sent in a streaming fashion, with short inter-arrival times between packets. A longer inter-arrival time then constitutes an indication that packets have been dropped. Figure 4.1 presents an example of the inter-arrival time distribution for one of the attacks, at millisecond resolution. The distribution shows the presence of larger gaps in the inter-arrival time, in this example, clustered around 102 ms.

Most of the attacks we measured had a similar behaviour pattern, as shown in Figure 4.1. To overcome the observed losses, in Algorithm 2, we compensate the time series of the raw (measured) traffic. For every packet, the algorithm determines whether it falls within the current or the next time bin of the time series (line 5). When the packet falls in the current bin, its size in bytes is counted and the time difference with the previous packet is calculated (line 6–7). We refer to a situation in which the time difference with the previous packet is larger than the pre-set threshold as a gap.

When a gap is detected, the gap duration is counted (line 8–9), which is needed for the compensation later on. If the considered packet falls in the next time bin (line 10), the actual compensation is performed (line 11) by dividing the total number of bytes within the bin by the compensated bin size (*i.e.*, the

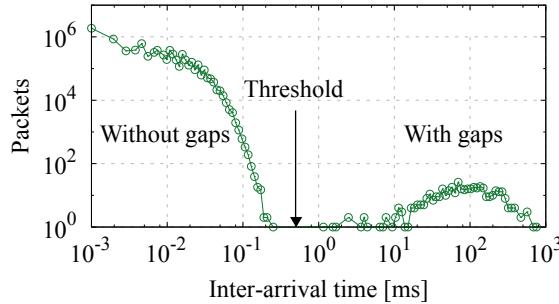


Figure 4.1: Example of inter-arrival time distribution.

Algorithm 2 Traffic compensation for sets of packets in a time bins, returning a compensated time-series.

```

input: pkts, threshold, bin_size
output: compensated_timeseries
1: bin_start_time ← pkts[0].time
2: bin ← 0; bin_data ← 0; gap_time ← 0

3: procedure COMPENSATINGPACKETLOSS(input,output)
4:   for i ∈ [1, pkts.length] do
5:     if (pkts[i].time - bin_start_time) ≤ bin_size then
6:       bin_data ← bin_data + pkts[i].size
7:       Δt ← pkts[i].time - pkts[i - 1].time
8:       if Δt > threshold then
9:         gap_time ← gap_time + Δt
10:      else
11:        compensated[bin] =  $\frac{\text{bin\_data}}{(\text{bin\_size}-\text{gap\_time})}$ 
12:        bin_start_time ← bin_start_time + bin_size
13:        bin ← bin + 1
14:        bin_data ← pkts[i].size
15:        gap_time ← 0
16:      end if
17:    end if
18:   end for
19: end procedure

```

duration of the bin in seconds, divided by the total duration of the gaps). Since the missing packets have not been accounted for in the total number of bytes per bin (as they are not available in the packet trace) but implicitly in the bin size, as we assume a constant number of packets per bin, only the bin size requires

compensation. Finally, after resetting the variables for the next run, the next packet is processed.

The value of the threshold was chosen based on the inter-arrival time distribution of each attack. In Figure 4.1, we have indicated the distribution of the traffic with and without gaps and the threshold must be chosen such that it discriminates between the left and right part of the distribution. In practice, the value of the threshold for all the attacks we measured is between 1 and 10ms. Our analysis shows that the algorithm is very accurate about what was measured at SURFnet. In the next section, we present the measured and compensated attacks.

4.3 Booter Attacks Analyses

Of the fourteen booters from which we purchased attacks, five did not perform the UDP-based attacks that we ordered: two of those did not send any traffic (booters B13 and B14), and three surprisingly generated a handful of TCP packets (booters B10–B12). The nine remaining booters performed attacks as requested and generated more than 250 GB of traffic in total. In the remainder of this section, we identify booters by numbers, from 1 to 9, to prevent legal and ethical issues around advertising booter services. All the collected traces are publicly available at https://www.simpleweb.org/wiki/index.php/Traces#Booters_-_An_analysis_of_DDoS-as-a-Service_Attacks.

Although there are several types of UDP-based attacks, our measurements only show seven DNS-based attacks and two attacks involving the CharGen protocol. This observation is in line with current trends, described in [73], which show DNS and CharGen as two of the most common types of UDP-based attacks. Both types of attack (DNS and CharGen) belong to the class reflection and amplification attacks. These attacks are based on the principle that an attacker sends a relatively small request to a server, crafted with the spoofed IP address of the intended target (reflection), and for which the response is much larger than the request (amplification). For example, in the case of a DNS-based attack, an attacker may send a relatively small DNS query (in the order of 40–60 bytes), which may be answered with a large response that can be 4 KB. For DNS, this amplification can be stronger when the extension mechanism for DNS (EDNS0) [22] is set, generating larger DNS responses (than the originally specified 512 bytes), with the most common maximum size set to 4 KB. In the case of CharGen [68], RFC 864 defines that requests to servers should be answered with a randomly-sized reply up to 512 bytes in size. In the next subsection, DNS-based attacks are analysed, followed by analysis of the CharGen-based attacks.

4.3.1 DNS-based attacks

Figures 4.2(a) and 4.2(b) show the volume of DNS-based attacks measured and compensated, respectively. By analysing both figures, it is clear that packets have been dropped for attacks with a rate higher than 400 Mbps. On the one hand, the rates of booters B2, B3, B5 and B7, for which the traffic rate is below 400 Mbps, are barely affected by the compensation algorithm. Booters B1, B4 and B6, on the other hand, show significant gaps for which the algorithm compensates. Our compensated results for all attacks, shown in Figure 4.2(b), are completely in line with what was measured by SURFnet.

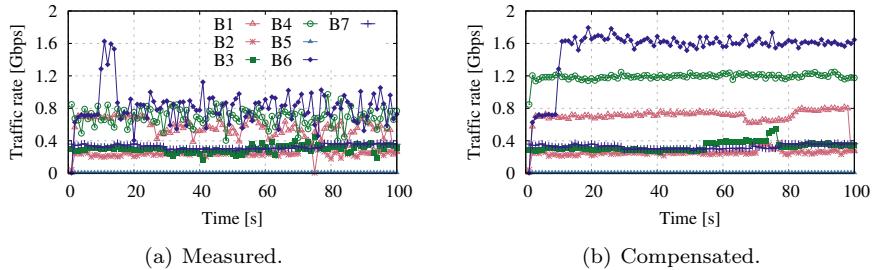


Figure 4.2: Traffic rate of DNS-based attacks.

Based on SURFnet’s experience, it is not a surprise that some booters (*e.g.*, B4 and B6) generated attacks with rates of more than 1 Gb/s. SURFnet have observed many episodes in which schools in the Netherlands have been taken offline by booters hired by students. More worrying is that all booters, except B5, generated rates high enough to saturate typical ADSL, ADSL2+ and DOCSIS connections, which are used by a large proportion of home Internet users and are also commonly used by small and medium enterprises [3].

Figure 4.3 shows the distribution of packet sizes in the attacks. The range in packet sizes for all booters significantly exceeds 512 bytes, the default maximum response size for regular DNS. The CDF even shows that for B4 75% of the distribution is concentrated around values as high as 4 KB. The size of DNS responses is an important factor in an amplification attack, since a large amplification factor—*i.e.*, the ratio between the size of the response and that of the request—will lead to an attack that requires fewer resources on the side of the attacker. By inspecting the packets captured at our measurement point, we found that all attacks make use of EDNS0, which allows for responses typically as large as 4 KB. Besides, it should be noted that all the attacks we saw use ANY queries to achieve maximum amplification. The DNS ANY query is used for

retrieving all resource records available for a given domain name.

Figure 4.3 shows the distribution of packet sizes in the attacks. The range in packet sizes for all booters significantly exceeds 512 bytes, the default maximum response size for regular DNS. The CDF even shows that, for B4, 75% of the distribution is concentrated around values as high as 4 KB. The size of DNS responses is an important factor in an amplification attack, since a large amplification factor *i.e.*, the ratio between the size of the response and that of the request will lead to an attack that requires fewer resources on the side of the attacker. By inspecting the packets captured at our measurement point, we found that all attacks make use of EDNS0, which allows for responses as large as 4 KB. It should also be noted that all the attacks we saw used ANY queries to achieve maximum amplification. The DNS ANY query is used for retrieving all resource records available for a given domain name.

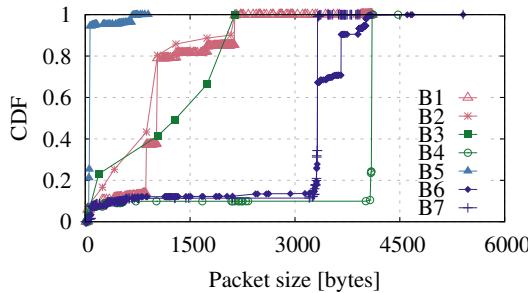


Figure 4.3: Packet size distribution (DNS).

Particularly noteworthy is that all seven DNS amplification attacks used identical values for certain query parameters. We therefore suspect that the script or program used to generate the attacks is the same or based on a common root source in all seven cases. Since knowledge of these particular parameters may help greatly in mitigating this type of attack, we will not disclose the specifics, to avoid helping attackers to improve their attacks.

Figure 4.3 also shows that booters that use the same DNS query (see Table 4.2) have a very similar distribution in packet length, such as booters B1 and B2, and booters B6 and B7. Bearing in mind that the query has almost the same size for all booters, we conclude that the amplification factor of the former group of booters is lower than that of the latter. What stands out is that B5 generates the shortest responses.

Table 4.2 shows the average rate of each attack, the number of systems involved (misused DNS resolvers) in performing the attacks, the average number of packets per system and the DNS query used for attacks. The most surprising

finding in Table 4.2 is that although B5 has the largest set of misused systems (8281 DNS resolvers), the rate of attack was the lowest (6.11 Mbps) of all the booters, since the involved resolvers on average sent only 261.8 packets each and the packet size distribution ranges from 20–900 bytes. This indicates that the number of hosts involved in an attack is not necessarily an adequate indicator of the attack strength. In fact, the volume of an attack is a function of the number of systems involved, the number of packets each system sends and the amplification factor. For example, although B3 relied on a set of misused systems more than one hundred times smaller than B7, this booter generated almost the same volume of traffic as B7. This was possible because the number of packets sent by B3 was 88 times larger than the number sent by B7.

Table 4.2: Details of DNS-based attacks.

| Alias | Average rate | Misused systems | Average packets per system | DNS query |
|-------|--------------|-----------------|----------------------------|------------------|
| B1 | 0.70 | 4486 | 2886.1 | root-server.net |
| B2 | 0.25 | 78 | 116082.5 | root-server.net |
| B3 | 0.33 | 54 | 245169.2 | root-server.net |
| B4 | 1.19 | 2970 | 12327.9 | packetdevil.com |
| B5 | 0.006 | 8281 | 261.8 | ddostheinter.net |
| B6 | 0.15 | 7379 | 1329.2 | anonsc.com |
| B7 | 0.32 | 6075 | 2756.7 | anonsc.com |

4.3.2 CharGen-based attacks

According to several reports [74, 72, 73], DDoS attacks based on CharGen barely appear before 2013, but since then their use has increased significantly. For example, from September to December 2013 Prolexic [73] reports an increase of 92.31%. Figures 4.4(a) and 4.4(b) show the rate of CharGen-based attacks measured and compensated, respectively. The traffic rate generated by B9 exceeded our expectations, with peaks around 7.0 Gb/s, almost four times higher than the largest DNS-based attack (B6).

Surprisingly, we noticed a significant discrepancy between the maximum allowed packet size in the CharGen protocol specification in RFC 864 [68] (512 bytes) and what we measured. As shown in Figure 4.5, for both booters B8 and B9, the size of packets is linearly distributed in the range of [0, 6956] bytes. We therefore suspected that the systems involved in the attacks were running a non-RFC-compliant implementation of the CharGen protocol. To verify this,

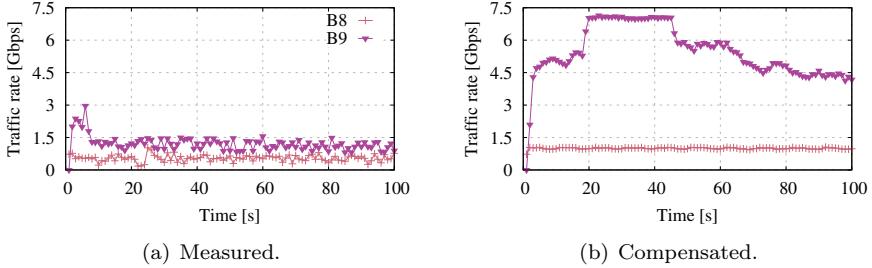


Figure 4.4: Traffic rate of CharGen-based attacks.

we examined the misused systems using nmap (<http://nmap.org>) and observed that the majority of these systems were running Microsoft Windows.

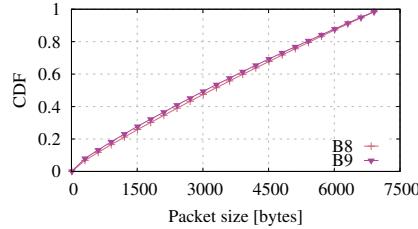


Figure 4.5: Packet size distribution (CharGen).

To verify whether the observed CharGen implementation is specific to Microsoft Windows systems, we installed several recent versions of Microsoft Windows, as well as the reference implementation of the xinetd (<http://www.xinetd.org>) daemon on Linux (which includes CharGen). When we tested the protocol in our lab environment, our results confirmed those of the live attacks for the implementations on systems running Microsoft Windows. The maximum CharGen packet sizes measured in our lab environment are remarkable: all Microsoft Windows versions from XP up returned messages with a random size of $[0, 6956]$ bytes. This observation confirms that Windows implementations are non-RFC-compliant. Also, since CharGen is installed as part of the Simple TCP/IP Services, Windows systems may become a powerful base for this type of amplification attack if these services are enabled.

Our tests also show that the xinetd implementation of CharGen is non-RFC-compliant, although in this case the maximum obtained response size is limited to 1024 bytes, on average 3.4 times smaller than for Microsoft

Table 4.3: Details of CharGen-based attacks.

| Booter | Average rate [Gb/s] | Misused systems | Average packets per system |
|--------|---------------------|-----------------|----------------------------|
| B8 | 0.99 | 281 | 20491.1 |
| B9 | 5.48 | 3779 | 3514.4 |

Windows.

Similar to DNS-based attacks, the traffic rates of the CharGen-based attacks depend on the number of systems involved, the number of packets sent per system and the implementation of the service on abused systems. In the case of booters B8 and B9, the attacks show a remarkable similarity in the packet size distribution (Figure 4.5), indicating that both booters abuse the same type of systems. From Table 4.3, we see that despite the systems controlled by B8 being more aggressive (20491.1 packets/system), B9 has activated a larger set of hosts, which resulted in a larger attack volume.

4.3.3 Geographical distribution of misused systems

We also examined the geographical distribution of the servers abused in the attacks. Since the attacks are reflection-based, the measured source IP addresses are the legitimate addresses of the misused systems and therefore geolocation provides meaningful results. Figure 4.6(a) and 4.6(b) show the geographical distributions of the servers involved in the DNS and CharGen attacks, respectively, cumulated for all the measured attacks. The Figures also show the top ten most active countries by number of misused servers. In the case of DNS, the top ten is dominated by the US (with more than 5.8k hosts) followed by Japan and Germany. This result is not surprising, since these countries are among the countries with the highest Internet penetration [3].

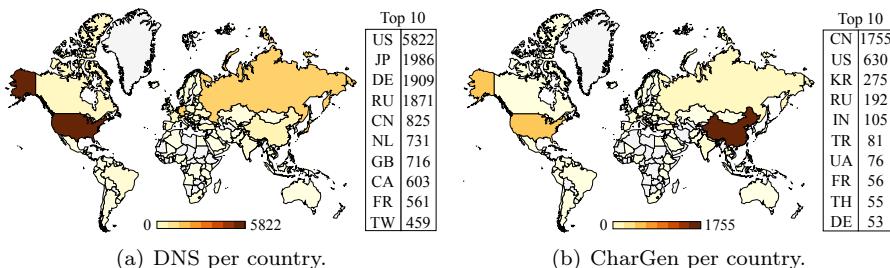


Figure 4.6: Geographical distribution of misused servers.

More surprising is the distribution of hosts abused for carrying out CharGen-based attacks. In this case, China dominates the top ten, while the US follows with only about a third of the number of hosts in China. It is currently unclear why China dominates the top ten for CharGen attacks. The predominance of China, however, was already observed in an earlier report by [73].

Finally, we investigated the geographical distribution of individual hosts involved in each attack. Figure 4.7 shows the continent breakdown of hosts misused by each of the nine tested booters. For the majority of the booters, except for B8 and B9 that generated the aforementioned CharGen-based attacks, the majority of misused hosts are located in North America (22–33%) and Europe (31–61%).

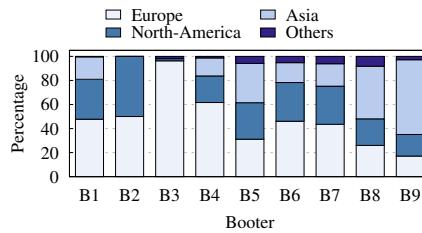


Figure 4.7: Continent breakdown per booter.

4.3.4 Differentiating booters based on their attacks

One of the main goals of this chapter is to investigate whether booters share the same infrastructure for performing attacks. We looked at this issue by comparing the set of misused systems in each of the observed attacks. Since online services can be misused to perform reflection and amplification attacks, and these services can be found by simply (port) scanning, we expected to find a significant intersection between the sets of misused systems.

Table 4.4 shows the pairwise intersection between the set of misused systems, calculated as the overlap fraction between the sets of misused host for booters of B_X and B_Y , expressed as a percentage: $\frac{|B_X \cap B_Y|}{|B_X|} \times 100$. Table 4.4 is not a triangular matrix, given that the numbers of misused systems are different in size. Contrary to our expectations, there is not much (or even no) infrastructure overlap among booters. This lack of shared infrastructure indicates that the set of misused hosts in an attack is part of the booter business model, and we speculate that booters may employ more advanced techniques in choosing their infrastructure than just harvesting by scanning. From Table 4.4, we also observe

that there are some exceptions to this observation. For example, the booter B6 shares 98.65% of its infrastructure with B7. By looking at the economic aspects, we found that the most likely reason for the large intersection is that B6 and B7 are both linked to the same PayPal account, indicating that these booters share the same owner. This finding also seems to indicate that booter owners offer different products with different prices to attract different customers.

Table 4.4: Intersection between sets of misused systems by the tested booters.

| \cap | B1 | B2 | B3 | B4 | B5 | B6 | B7 | B8 | B9 |
|-----------|-----------|-----------|-----------|-----------|-----------|--------------|-----------|-----------|-----------|
| B1 | – | 0.20 | 0.20 | 3.88 | 0.02 | 1.07 | 0.73 | 0 | 0 |
| B2 | 11.54 | – | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| B3 | 16.67 | 0 | – | 0 | 0 | 1.85 | 1.85 | 0 | 0 |
| B4 | 5.86 | 0 | 0 | – | 0.20 | 4.11 | 1.04 | 0 | 0 |
| B5 | 0.01 | 0 | 0 | 0.07 | – | 8.38 | 7.99 | 0 | 0.08 |
| B6 | 0.65 | 0 | 0.01 | 1.65 | 9.42 | – | 81.33 | 0 | 0.07 |
| B7 | 0.54 | 0 | 0.02 | 0.51 | 10.90 | 98.65 | – | 0 | 0.08 |
| B8 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | – | 43.06 |
| B9 | 0 | 0 | 0 | 0 | 0.18 | 0.13 | 0.13 | 3.20 | – |

Lastly, Table 4.4 also shows a small intersection between the hosts used to perform DNS-based and CharGen-based attacks (*e.g.*, B9 correlates with B5, B6 and B7). This same intersection pattern is also described in [62], without a clear conclusion. Perhaps systems being setup with the intention of being involved in attacks would be a possible explanation.

Table 4.4 also clearly indicates that booters have a high potential for future attacks. Booters can easily increase their firepower by using each other’s infrastructure. In the case of DNS-based attacks, our measurement includes 29,321 single misused IP addresses. This could indicate an increase in firepower between 3.5 (B5) and 542 times (B3). For example, if booter B8, which uses 281 systems, uses the 3,779 systems of booter B9, then B8 could generate an attack up to 13 times stronger than what we measured, possibly reaching 13 Gb/s.

The potential firepower of booter attacks may be even worse. Work by Marc et al. [62] shows that, as of early 2014, there were at least 89,000 CharGen amplifiers on the Internet. If, for instance, booter B9 abused all of these, it could increase its firepower by over 23 times, potentially achieving peak attack volumes over 160 Gb/s. Marc et al. [62] describe measurements over a three month period in late 2013 and early 2014, showing a pool of open DNS resolvers

between 23 and 25.5 million hosts in size. Assuming the lower boundary and that, for example, booter B6 abuses all available open resolvers, it could reach well over 3,000 times the attack volume it achieved in our measurements.

Four years after our measurements (2017), the number of amplifiers has reduced. For example, Shadow Server (available at shadowserver.org) reported (in July 2017) that the number of DNS servers is around 9 million, while for CharGen it is 28,652. Although the number of amplifiers decreased significantly, the power and frequency of attacks increased. This is mostly explained by two things. First, our predictions are right and booters began misusing a larger set of devices. Secondly, in early 2017, booters started using large botnets to send traffic without the need for amplifiers to generate Tbytes of data per second attacks.

Overall, the most important takeaway from our observations in this section is that botter attack infrastructures are distinct. Therefore, an analysis of a set of source IP addresses found in a generic attack can be used to indicate which (owners of) booters should be held responsible for the attack.

4.4 Booters Behind DDoS Protection Services

After concluding that booters can be differentiated by their attack characteristics, a follow-up question is which companies help to host botter websites. The goal of this investigation is to identify web-hosting providers that could act to mitigate booters. Therefore, as well as purchasing attacks from 14 booters, we also tracked where websites for a larger list of 102 booters were hosted. To track where botter sites were hosted, we used a passive DNS data source called DNSDB and provided by Farsight Security (<https://www.dnsdb.info>). This data source holds information from the DNS over time, mapping host names (in our case, botter hostnames) to IP addresses. Based on these IP addresses we can determine where botter websites were hosted over time.

Surprisingly, the results of our analysis did not reveal the web-hosting providers that support botter operations but network security companies that provide DDoS Protection Services (DPses). These services act as proxies for their customers, combining the concept of Content Delivery Networks (CDNs) for improving the availability of their clients with advanced traffic filtering to neutralise DDoS attack traffic. In the botter market, where DDoS attacks are the main product, the best way to beat the competition is to isolate them from the Internet. As presented in the previous chapter, booters attack one another for competition purposes. A possible mitigation approach for booters, therefore, would be to use DPses themselves, as a countermeasure against attacks from

competitors.

Figure 4.8 shows the percentage of time each of the 102 booters was protected by a DPS. In Figure 4.8, booters are sorted by the year in which their domain name was registered. Although there are several DPSes available on the Internet, such as Versign, Akamai and Prolexic, only Incapsula and CloudFlare were found to be used by booters (in our dataset). A possible explanation for this penchant for Incapsula (<http://incapsula.com>) and CloudFlare (<http://cloudflare.com>) may be that only these two offer the option of a free subscription. However, we are also aware of the fact that the DDoS attack mitigation mechanisms of these DPSes are not included in the free subscription, leading us to conclude that booters may subscribe to more advanced protection plans.

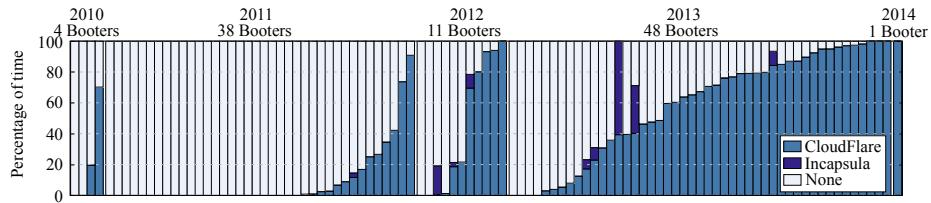


Figure 4.8: Percentage of time that 102 Booters are protected, sorted by the year they started to be accessed.

Figure 4.8 also shows that 89% (53 out of 59) of booters that started their activities between 2012 and 2014 were protected by DPSes for part of their lifetime. On average, booters spend an increasingly large fraction of their lifetime protected by DPSes, as indicated in Table 4.5. New booters that started operations in 2014 are all protected by a DPS. The more long-lived booters, like the ones that became active in 2010 and 2011, also make use of DPSes. However, we found that these booters mostly started subscribing to protection services from 2011. This finding lead us to believe that the trend of using DPSes in the botter market started in 2011.

Table 4.5: Average fraction of time in DPSes, per year.

| Year | CloudFlare | Incapsula | None |
|------|------------|-----------|-------|
| 2010 | 22.41 | 0.09 | 77.51 |
| 2011 | 9.07 | 0.08 | 90.85 |
| 2012 | 43.55 | 2.75 | 53.70 |
| 2013 | 58.18 | 2.44 | 39.37 |
| 2014 | 100 | 0 | 0 |

An in-depth analysis of the functionality of DPSes is outside the scope of this thesis. We note here, however, that DPSes could play a major role in mitigating booters. By acting as a proxy, DPSes can access information specific to booters, such as the real IP address of booter websites, in addition to information about customers accessing the service or the booter owners. Attack parameters such as the target IP address could also be used to pre-empt attacks. More research is needed to understand which types of information these services can access and which types of mechanisms they offer against different types of DDoS attacks. In the next chapter, we update the findings on booters subscribing to DPSes and extend our analysis to investigate which other third-party services (in)directly support booter operations.

4.5 Concluding Remarks

In this chapter, we had two main goals. The first was to answer the question of whether booters deliver what they promise, while the second was to address whether their attack infrastructures can be used to differentiate between booters. To answer our questions, we used a measurement-based approach in which we hired and measured booter attacks in a controlled environment. We also discussed some ethical and legal considerations of our measurements, which were supported by a Dutch public prosecutor.

In the second half of 2013, we subscribed to attacks from 14 booters to attack our infrastructure and analyse the traffic generated by these attacks. We measured primarily UDP-based reflection and amplification attacks, using DNS and CharGen. While DNS amplification attacks are well-known, CharGen attacks are relatively new and, since 2013, have been rapidly gaining popularity. The DNS-based attacks, for which we had to pay only a few dollars, showed traffic peaks of up to 1.6 Gbps, whereas the CharGen attacks showed peaks around 7.0 Gbps. Although those attack rates are sufficient to overload any ADSL Internet connection (which represents the majority of Internet users and the majority of small and medium size online businesses), it is still far less than what is advertised as the maximum power of booters.

A surprising observation regarding the attack sources is that booters do not use the same hosts to amplify their attacks. This observation means that by looking at the source of attacks we can determine which booter is responsible for the attack (given a ground truth). The fact that booters do not share their attack infrastructure also means that attacks might become much stronger once a single booter can exploit all systems currently used for amplification. Although the number of amplifiers has decreased significantly (in 2017 compared to 2013), the power and frequency of attacks has increased. This is explained by booters

using a larger set of systems (compared to 2013) and also them starting to use large size botnets to send attack traffic without the need for amplifiers.

By looking at the characteristics of attacks (CharGen and DNS-based), we conclude that CharGen-based attacks are the easiest to mitigate right away. CharGen uses the fixed UDP port 19, which can simply be filtered at the network border router, or, if this would not be sufficient, by an upstream network provider. It remains questionable in our opinion, however, whether CharGen should be installed on end-systems at all. CharGen seems to be mostly abused for DDoS attacks and has limited benign applications nowadays.

It is much harder to take countermeasures against DNS-based attacks. Blocking DNS traffic, for instance, is impossible as it would prevent end users from properly using the Internet. DNS-based attacks benefit from a large number of open DNS resolvers, often installed by default in Customer Premises Equipment (CPE), such as home routers. An obvious step towards mitigation is therefore to disable such services. Another approach is to rate limit DNS traffic on network edge routers, since DNS traffic should rarely be more than a fraction of overall traffic. Note that this may be detrimental to legitimate DNS servers running inside the network that performs the rate limiting, so it is no catch-all solution.

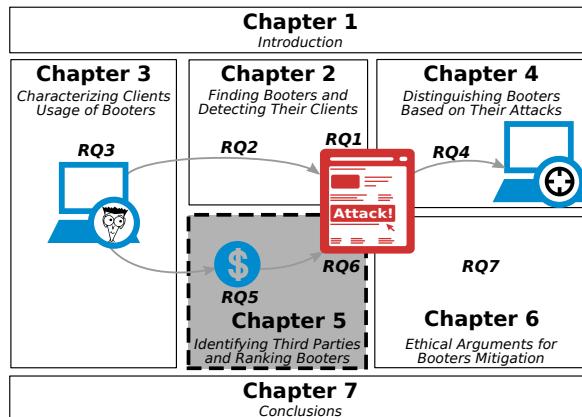
After we had observed that booters could be identified via their attack characteristics, we tried to identify which web-hosting providers were related to booters. For this additional investigation, we analysed a total of 102 booters, using (historical) DNS data from North America, covering a period since 2010. Based on this analysis, we could not identify the hosting providers but identified companies that offer DDoS Protection Services (DPSes) to booters. Since 2010, the number of booters that protect themselves against the same types of attack that they sell, by making use of DPSes, has grown dramatically and every booter that we know to be active in early 2014 was behind a DPS. This fact does open up avenues for future work in combating booters; if DPSes can be compelled to collaborate, characterising who runs booters and what their internal infrastructure looks like becomes a lot easier.

"Have you noticed how many people who walk in the shade curse the Sun?"

—IDRIES SHAH,
IN: REFLECTIONS, 1968

Identifying Third-Parties and Ranking Booters

In the previous chapter, when we tried to discover which web-hosting providers were the most used by booters, we identified booters protected by cloud-based DDoS protection organisations. In this chapter, we extend that investigation to answer the question of **what third-party companies are used by booters?**(RQ5). After answering this question, we also discuss the effect that mitigation actions performed by these third-party organisations would have on the booter ecosystem. In addition to this discussion, we propose a ranking methodology to find **which booters are most dangerous?**(RQ6). Our methodology in this chapter is based on combining measurement datasets that we collected ourselves with those retrieved from public sources.



The organization of this chapter is as follows:

- In section 5.1, we introduce the goals of the chapter;
- In section 5.2, we identify the organizations involved with booters that can also play a role on mitigating booters;
- In section 5.3, we propose a ranking methodology to highlight booters that should face mitigation actions at first;
- In section 5.4, we present our concluding remarks.

5.1 Introduction

As described in Chapter 2, booters can easily be found on the public Web through search engines, using Google for example. Distributed Denial of Service (DDoS) attacks performed by booters can be hired for a couple of US\$. Booters also offer multiple ways of paying for their “services” (*e.g.*, PayPal, Bitcoin or credit card), while offering various types of attacks (*e.g.*, SYN flood, DNS-based reflection or application layer attacks). Karami et al. [41] showed that several factors have contributed to the increasing occurrence of DDoS attacks: a significant number of active booters; the ease with which booters can be found and the ease with which they can be hired. Their observations have proved to be correct as since 2015 a majority of attacks, including the most powerful DDoS attacks, have been launched by booters (at a data rate higher than 100 Gbit/s), as reported by Akamai [7].

Among others reasons for the increase in the number of active booters and attacks by them, are the existence of third-party organisations that (in)directly support booters’ operation. In the previous chapter, while investigating where booter websites were hosted and located, we discovered network security companies that protect booters against DDoS attacks. In this chapter, this finding leads us to an in-depth analysis of third-party organisations that (in)directly support booter operations. We also discuss what mitigation actions could be instigated by those third-party organisations to inhibit or even dismantle booter operations.

After identifying third-party organisations that (in)directly support booters, we propose an empirical methodology to rank booters. Our goal with this methodology is to highlight booters backed up by third-parties that should face mitigation. There are hundreds of booter websites available and performing mitigation actions against all at once is unrealistic. Only a few booters, those involved in massive attacks, have experienced mitigation actions. Booters that have been the target of investigations, mitigations or prosecutions are those that successfully disrupted the operation of popular services, such as Xbox Network, PlayStation Network, Instagram and Tinder [50]. The ultimate goal of our ranking methodology is to reveal booters offering powerful attacks for a low price that have previously never been the target of legal action.

Our methodology involves both investigating the third-party organisations and developing our ranking heuristic and is based on combining measurement datasets that we both collect ourselves and retrieve from public sources. The ground truth used in this chapter is a list of 435 Booter domain names from the Booter Blacklist initiative (<http://booterblacklist.com>, available at Appendix B, and retrieved on 27-July-2017). Our dataset and associated scripts for data analysis are publicly available at http://jairsantanna.com/booter_

`ecosystem_analysis.`

5.2 Identifying Third-Parties

To identify organisations (in)directly involved in booter activities, we review the elements already discussed in previous chapters. Figure 5.1 shows the ecosystem of booters, *i.e.*, elements involved in booter activities. To hire an attack, the client must first access the booter website and create an account. As described in the previous chapter, access to a booter website usually happens via a third-party Cloud-Based Security Provider (CBSP), also called as DDoS Protection Service (DPS), transparent to the client. The payment for a hired attack (or an attack plan - sets of attacks that can be performed within a given period) is made via a third-party payment system. After selecting a “service” and paying for the plan, clients can launch attacks at any time and against any target on the Internet.

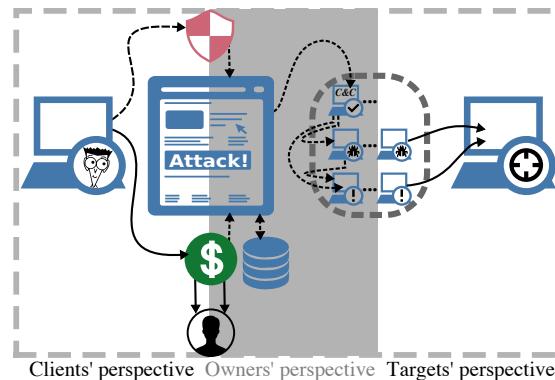


Figure 5.1: Booter ecosystem elements.

To perform DDoS attacks, booters use a back-end infrastructure that consists of three types of machines: command and control (C&C) machines, infected machines (computers with bugs in Figure 5.1) and misused public services (computers with exclamation marks in Figure 5.1). Booters are unlikely to send attack traffic directly from their C&C machines. Instead, infected machines can be used as part of a botnet able to perform various types of attacks. Misused public services are only used for reflection and amplification attacks (*e.g.*, DNS-based and NTP-based attacks). The final element in the booter ecosystem is the booter operational database, where all information about clients and hired attacks is stored.

In addition to CBSPs and payment systems, five other organisations are also (in)directly involved in the booter ecosystem: (1) web-hosting companies that host the content of booter websites; (2) Top-Level Domain (TLD) operators; (3) domain registrars that provide the means for the registration of booter domain names; (4) web indexing and searching companies that facilitate the finding of booter websites and (5) local DNS resolvers that resolve booter domain names to IP addresses. In the following section, we discuss to what extent these third-parties are involved with booters and discuss potential actions they could perform to support the mitigation of booter operations. The starting point of the analysis presented in this section is the list of 435 booter domain names provided by booterblacklist.com (retrieved on 17th July 2017 and available in Appendix B).

5.2.1 TLDs Operators, Domain Registrars, Web Hosting Companies, and CBSPs

These four types of organisation (*i.e.*, TLDs operators, domain registrars, web-hosting companies and CBSPs) are analysed together for the following two reasons. First, they are linked to booters mainly via domain names. Second, the same company may provide different types of services. Examples of such organisations include SIDN (<https://sidn.nl>), which is both a TLD operator (of .nl) and a domain registrar; GoDaddy (<http://godaddy.com>), which is both a domain registrar and a web-hosting company and CloudFlare (<https://cloudflare.com>), which is both a web-hosting company and a CBSP.

We use distinct methodologies to analyse each of these four types of organisation: for TLDs, we look into the composition of booter domain names; for domain registrars, we rely on Whois information and for web-hosting and CBSPs, we use their respective IP address and Autonomous System (AS) information (<http://www.team-cymru.org/IP-ASN-mapping.html>).

As shown in Figure 5.2, by analysing the composition of domain names we see that more than 68% of all 435 booters are registered within the .com and .net TLDs. Other TLDs account for less than 5% of registrations each. For example, .nl accounts for around 1% of registrations. We also see that 74% of booter domain names contain the terms stresser, booter or ddos. Information on the composition of domain names could be used by TLD operators and registrars to, for example, take down existing domains or prevent the registration of new (suspect) ones. An enabler to check booter domain names was proposed in chapter 2. Preventing the registration of new domains could, however, have an impact on the registration of valid domain names that could mistakenly be classified as suspect.

We analysed the impact of domain names that have the terms stresser, booter

or ddos in their composition, and are registered within .com, using a large-scale active DNS measurement dataset [95]. We found that of all 2,721 domain names in .com containing one of the three terms, only 61 domain names (less than 3%) are not related to booters. That is, in filtering registrations based on these three terms a very small percentage of legitimate registrations would be affected. However, booter owners could overcome these actions by adopting alternative terminologies, such as shutdowner and blackouter. Therefore, it is important to emphasise that this solution is not definitive and it is crucial to keep up-to-date with variations that booter owners can use on keywords to overcome mitigation strategies.

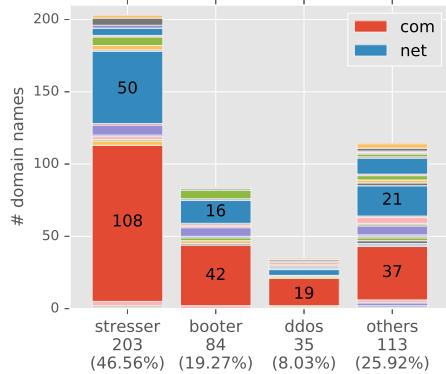


Figure 5.2: Domain word composition and TLDs distribution (.com and .net highlighted).

By analysing booters' Whois information, we observe that almost 70% of all booters are within the top-ten registrars, as can be seen in Figure 5.3. If only Enom, GoDaddy and Namecheap decided to act against booters, around 50% of all booters would be affected.

When looking at the IP addresses and ASs related to 202 (online) booter domain names, we also noticed that some companies would have a higher impact if they took mitigation action. For example, CloudFlare is involved with at least 76 booters (37%). The fraction of booters behind CloudFlare dropped significantly compared to the results in the previous chapter from (88%), of the 52 CloudFlare-involved booters in that chapter, 49 booters are seen in the current analysis. Given that booters typically attack each other [80], competing for market share or even to simply show off their attack power, we believe that if CloudFlare (and other CBSPs) stopped protecting booters, they would eventually take each other offline or at least compromise each other's

reachability. However, this action would only have an impact if all CBSPs decide to get involved, leaving no options for booters to be part of a DDoS protection service.

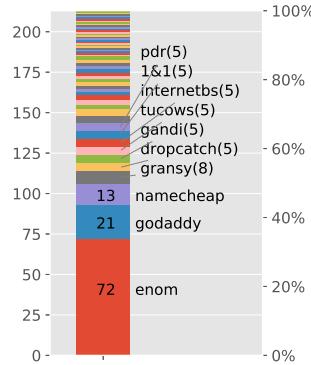


Figure 5.3: Registrars analysis based on Whois information (absolute numbers in y-axis).

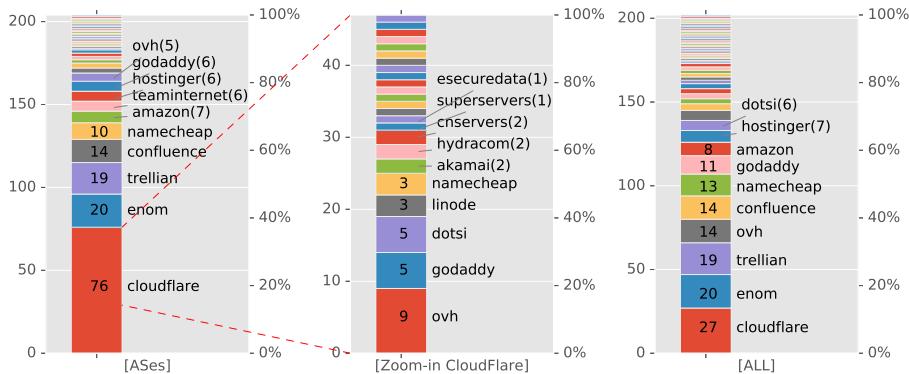


Figure 5.4: Web hosting analysis based on ASes (left), with zoom-in on the ASes hidden by CloudFlare (middle), and the overall merged results (right).

As we discovered in the previous chapter, booters behind CBSPs require a more refined investigation to determine which web-hosting company actually hosts their websites. To determine the web-hosting companies specifically obscured by CBSPs, we used the CloudPiercer initiative (<http://cloudpiercer.org>) [98], which applies several metrics to lookup actual (or

historical) IP addresses. Using this methodology, we found that 24 web-hosting companies host 47 booters (out of 76 in CloudFlare), as depicted in the middle (zoom-in) graph of Figure 5.4. The other 29 booters are also likely to be protected and hosted by CloudFlare.

Merging web-hosting companies in Figure 5.4 (ASs) with the previously hidden ASs, we observe that the top-ten web-hosting companies remains the same (though their ranks change). For example, comparing the left and right graphs in Figure 5.4, it can be observed that OVH and GoDaddy gain six and two positions respectively. The main takeaway message from this analysis is that if the top web-hosting companies engaged in effective mitigation actions, *e.g.*, simply stop hosting alleged booters, a high percentage of booters would go offline. However, booters could adapt to such an action by moving to other hosting companies.

5.2.2 Payment Systems

Payment systems are one of the main elements of the booter ecosystem. Karami et al. [41] reported a joint effort by PayPal and the FBI, in which Paypal accounts allegedly belonging to booter owners were deactivated. This operation was very successful, momentarily reducing the number of payments and attacks by booters. However, booters have partially overcome this mitigation action. For example, only one booter among those listed in Figure 5.5 still offers PayPal as a payment option.

The majority of booters now use various crypto-currencies, such as Bitcoin, Litecoin and Dogecoin. This change in the payment system makes it harder to trace booter owners by following the money they earn. The action by PayPal had a positive impact on damaging the booter ecosystem, given that only a small number of booter clients have Bitcoin wallets. Also, based on the profile of a typical booter client (revealed in chapter 3), we believe that not many of them would be willing to create a Bitcoin wallet to perform attacks.

5.2.3 Web Searching Companies

It is extremely easy to find booter websites through public search engines, such as Google, Bing and Yahoo. To prevent users from interacting with booters, search engines could notify the users that hiring or even accessing booter “services” may have legal implications. This action is similar to one currently performed for “unsafe sites” in Google Chrome [30], and could reduce the number of accesses to booters and, ultimately, the number of attacks launched by booters.

5.2.4 DNS Resolver Operators

A straightforward way to prevent users from accessing booters is blacklisting booter domain names at DNS resolvers. When this is done, when an IP address resolution is needed for a blacklisted domain name, the resolution is refused. Booter websites would still be reachable via alternative DNS resolvers that do not block the resolution or via VPNs or Tor browser. However, considering that booters under CBSPs (by default) block access from VPNs and Tor nodes, this action by DNS resolver operators could ultimately result in a significant reduction in the number of attacks from booters.

It is very important to highlight that the mitigation actions described in this section might require a court order before they are put in place. For example, the CloudFlare CEO stated that “it is tricky when private organizations act as law enforcement” and that “they comply with any court order” [53]. Although the legitimacy of services offered by booters is still debatable, in chapter 6 we demonstrate that it is unlikely that booters provide legal and ethical services.

5.3 Ranking Booters

Mitigation and prosecution actions performed against the booter ecosystem (*i.e.*, websites, owners, clients, attack infrastructure and third-party organisations) have mostly targeted those booters that have launched powerful attacks towards large organisations. However, there are still hundreds of booters, such as those revealed by the booter blacklist initiative (<http://booterblacklist.com>), which are somehow “under the radar” of security initiatives and therefore rarely the target of mitigation actions. Obviously not all booters could be mitigated at once, but a priority order would be welcomed. The first booters to be mitigated should preferably be the ones that perform the most powerful attacks. Identifying these booters, however, is a task mostly restricted to large network security companies, which have the ability to classify the most dangerous attacks targeting their clients. In this section, we describe a heuristic to prioritise the mitigation of a (second) set of booters. Our heuristic relies on the following three premises:

- Booters’ services are not likely to be ethical or legal. It is debatable whether an illegal booter can be a legitimate stress tester. One of the main arguments for this, presented in the next chapter, is that the attack infrastructure used by booters mostly consists of compromised/misused machines (*e.g.*, botnets and amplifier services). Others have attested to this argument by hiring attacks from booters and testing them against controlled environments [39, 17, 81];

- The ratio between the number of accesses to their websites and the number of launched attacks is similar between all booters. This premise leads us to conclude that the most accessed booters are those likely to launch more attacks;
- The attack power advertised by booters can be accurate. It has been observed that booters, in general, deliver far less attack power than they promise to their clients (presented in chapter 4). However, booters that have caught the attention of the media have performed attacks stronger than those they advertise on their websites. For example, lizardstresser.su attacked the PlayStation and Xbox networks during Christmas 2013 with 300 Gbit/s attack power, while their website (as of 2013) only advertised attacks of up to 125 Gbit/s. To further support our premise, we argue that it is quite easy to find amplifiers for reflection attacks and to compromise a large number of systems (*e.g.*, Internet of Things devices, such as the ones used by Mirai botnet [52]). Therefore, skilled hackers and owners of booters can easily scale up their attack power (presented in chapter 4).

Based on our premises, our heuristic to highlight booters consists of four steps. First, we identify the most accessed booters using the website ranking provided by Alexa (<http://alexa.com>). For each of the 435 booter domain names on <http://booterblacklist.com> (available at Appendix B), we scraped the Alexa rankings 1st November 2016 to 1st February 2017. Our analysis only considers those booter domain names that ranked up to 3M-th in Alexa, which represents around 1% of the total number of registered domain names on the entire Internet [96]. We then scraped these top-ranked booter domain names to reveal their highest advertised attack rate (*i.e.*, the most powerful attack) and their price range. Finally, we investigated the dates of creation and expiration of their domain names based on Whois information. This final step shows how long the top-ranked booters have been offering (and probably delivering) attacks, supposedly without facing any mitigation action.

Figure 5.5 summarises our findings. From the ground-truth list of 435 booter domain names, 33 ranked among the top-1% most accessed domain names on the Internet (Figure 5.5a). In addition to their position in Alexa's rank, we observed that eight booters offer attacks with a rate of 100 Gbit/s or higher (Figure 5.5c); these are the booters ranked in the following positions of the Alexa rank: 4, 7, 8, 13, 14, 18, 25 and 32. Attacks of 100 Gbit/s are powerful enough to bring most systems offline on the Internet, especially those that are not protected by large security companies. Figure 5.5b shows that some of these eight booters (namely, booters ranked at 4, 14, 18 and 32) charge at most US\$100, while their cheapest service plan is US\$10 or less. Such a range of prices is surprising as the cost for recovering from a DDoS attack is on average US\$53K for small and

medium companies and US\$417K for large companies [42].

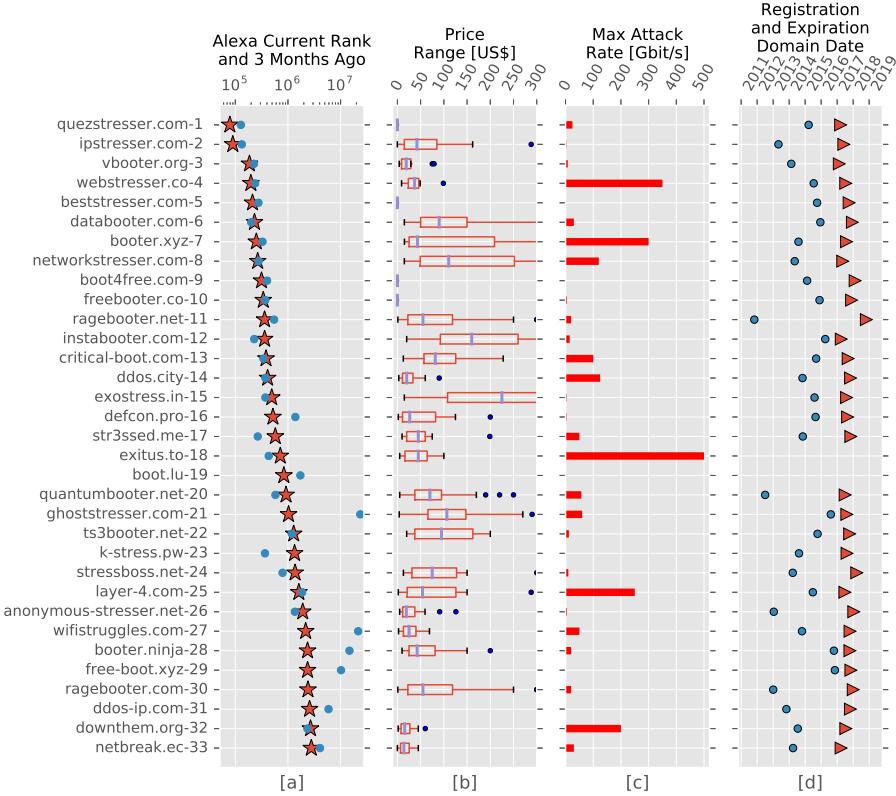


Figure 5.5: (a) Top ranked booter domain names, up to 3M-th position in Alexa—red star is the current rank; blue circle is the rank of 3 month ago. (b) Distribution of price ranges for each booter, including outliers. (c) Maximum advertised attack rate in Gbit/s. (d) Blue circles: registration of domain names; and red arrow-heads their respective expiration dates.

Based on our premises and findings, it can therefore be deduced that these four booters (ranked at 4, 14, 18 and 32) are the ones that, with the lowest cost for their clients, can do the most damage to the target of an attack. Furthermore, we observed that four other booters offer attacks for free (booters ranked 1, 5, 9 and 10). However, upon closer examination of these booters, we discovered that, except for the booter ranked 9th, they all promote services from other (paid) booters. We believe that these “free-service” booters are used to increase

the popularity of paid booters.

From those booters listed in Figure 5.5, three domain names are currently for sale: ranked 19, 29 and 31. These domains pointed to websites of booters which were active in the past, as confirmed by the Internet Archive initiative (<https://archive.org>). The Internet Archive has dozens of historical snapshots of these specific domain names. These are still highly ranked domains in Alexa because users still try to reach them or DNS resolvers keep trying to refresh their cache (due to bad implementation). This assumption is supported by the DNSDB initiative (<https://www.dnsdb.info>), one of the largest collections of DNS records worldwide. We found, in DNSDB records, that each of these three domains have received thousands of DNS requests (likely interpreted as web access) in the last two years.

Finally, we observed that two booter domain names (ranked 11 and 30) point to the same booter website. This booter has the oldest domain creation date among the top-ranked sites: it was registered in 2011. Although it was reported in 2013 by the security specialist Brian Krebs [47], we are unaware of any mitigation or prosecution action against it. A possible explanation is that this booter is actually an “FBI backdoor”, as described by its owner [53]. In a speech, the CEO of CloudFlare said that: “sometimes we have court orders not to take (web)sites down” [53]. Whether this is true or not, the concrete fact is that this booter is still online.

Our heuristic clearly provides means to highlight booters “under the radar” of security companies, which should be the first to undergo mitigation actions. In the next section, we present our concluding remarks.

5.4 Concluding Remarks

In this chapter, we had two goals. The first goal was to identify third-party organisations that are (in)directly used by booters and could act to mitigate booters. Our second goal was to determine booters “under the radar” of security actions that should face mitigation in a priority order.

Concerning the first goal, we learned that dismantling the entire booter ecosystem is very challenging. None of the mitigation actions mentioned in section 5.2 could, on a standalone basis, eliminate the booter phenomenon. However, if some of them were deployed we would certainly see a decrease in booter operations, similar to that after PayPal’s operation against booters in 2015. This decrease would be mostly in layman users (*i.e.*, booter clients) who would not be able to overcome the challenges imposed by the mitigation actions. While technically skilled users would still find a way to use booter services, they remain a minority.

To achieve the second goal, we proposed a heuristic based on (1) website popularity, (2) maximum attack rate, (3) price range and (4) domain creation and expiration. Using this heuristic and a set of premises, we identified 33 booter domain names that should face higher priority mitigation and provided arguments to justify the need for such mitigation actions. We showed that booters “under the radar” pose a potential risk and, as such, we consider proactive mitigation to be the best course of action.

Booter owners are likely to find ways to overcome any mitigation action. Booters can profit from relatively safe business when they do not attract too much attention from society and security specialists. To date, legal action against both booter owners and clients has only been taken in cases where large corporations were targeted by DDoS attacks. This chapter raises awareness about the hundreds of silent booters, safely operating “under the radar” of security actions, which could at any point in time cause substantial damage to any system on the Internet. We believe that our findings can foster further discussions and effective actions against booters.

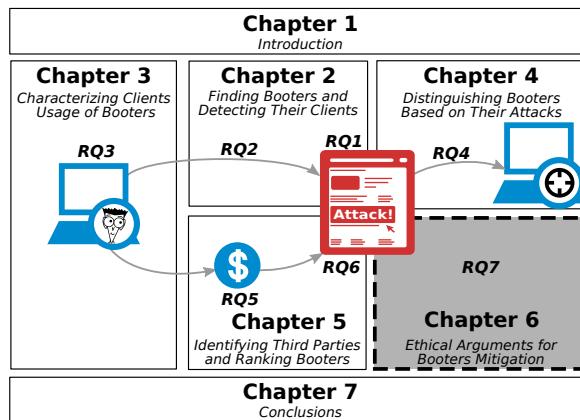
"You can't connect the dots looking forward; you can only connect them looking backwards. So you have to trust that the dots will somehow connect in your future."

—STEVE JOBS
STANFORD COMMENCEMENT SPEECH, 2005

CHAPTER 6

Ethical Arguments for Booters Mitigation

In the previous chapters, we investigated booters, their clients, their attack characteristics and the third party organisations used by them. In this chapter, we used the findings from these previous chapters to answer to question of which ethical arguments can be used to support mitigation actions against booters?(RQ7). First, we discuss two arguments commonly used by booters to justify their operation and usage, i.e., that they provide useful tools for testing the network capacity and that DDoS attacks are morally acceptable as a form of civil disobedience. Then, we refute these two arguments based on the findings from previous chapters. By refuting their arguments, we disqualify booters from being morally justifiable and, ultimately, support legal mitigation actions against booter operators and their users.



The organization of this chapter is as follows:

- In section 6.1, we identify cases in which DDoS attacks could be morally acceptable;
- In section 6.2, we discuss the justifications for using and operating booters attacks;
- In section 6.3, we present our concluding remarks.

6.1 Introduction

DDoS attacks are not a new phenomenon and are explicitly illegal in most jurisdictions. For example, it is illegal in the United States as described at the Computer Fraud and Abuse Act, and in the United Kingdom as described at the Computer Misuse Act. Despite this illegality, booter owners claim that their services have legitimate uses, such as testing a system or networks ability to handle heavy network traffic. Booter owners sometimes refer to their services as *network stressers* to emphasise this purpose.

Plausible justifications have also been presented for regarding some DDoS attacks as examples of public protest or civil disobedience in the Internet [63, 85]. This chapter will not address the question of the overall legitimacy or otherwise of DDoS attacks as a form of protest. Such arguments can be found elsewhere [85, 45]. Instead, we will focus specifically on how the particular characteristics of booters (found in the previous chapters) affect these justifications.

We claim that booters differ from previous methods of launching DDoS attacks due to two characteristics: the ease with which any person can launch a powerful DDoS attack, and the involvement of a third party, the booter operator, who provides the infrastructure necessary to perform a DDoS attack as a ‘for-hire’ service available to anyone. In this chapter, we argue that these differences undermine what we consider to be the most plausible argument in favour of performing DDoS attacks: *that they are a legitimate form of civil disobedience*.

In this chapter, we argue against the use of booters as a means of network stressing and claim that only in a limited set of circumstances there is a plausible moral justification for operating or using booters. We begin by describing the main characteristics of booters, including the infrastructure they use, the components that perform the attack itself, and the various companies connected to operate them. We then turn to the question of whether any DDoS attack can be morally justified, and if so, whether the characteristics of booters affect such a justification. First, we consider the use of booters as network stressers by examining the roles played by the various agents associated with booters and discuss whether any of them can offer a reasonable moral justification for their actions. After that, we present our argument that while some DDoS attacks may be classified as acts of civil disobedience, consider whether the particular characteristics of booters undermines the legitimacy of this justification.

6.2 Revisiting Booter Characteristics

In this section, we revisit concepts and findings from previous chapters aiming to highlight observations that will be used in the next section (against booters)

owners and clients). DDoS attacks employ large numbers of coordinated attackers to make their target inaccessible. The software performing the attack may be explicitly launched and controlled by the owners of the systems running them, or are running secretly on systems without the owner’s knowledge (such as in a botnet). This is often the result of a system’s security being compromised by malware or by a third party exploiting security flaws to allow unauthorised software to run on it.

What distinguishes booters from other methods of performing DDoS attacks is the ecosystem that makes such attacks available as a service for clients. In Figure 6.1, we present an extension of Figure 1.4 (from Chapter 1), which depicts the elements involved with booter websites. The extension is based on findings from previous chapters. In Figure 6.1, the elements in red colour are controlled by booter operators, while blue elements are third parties.

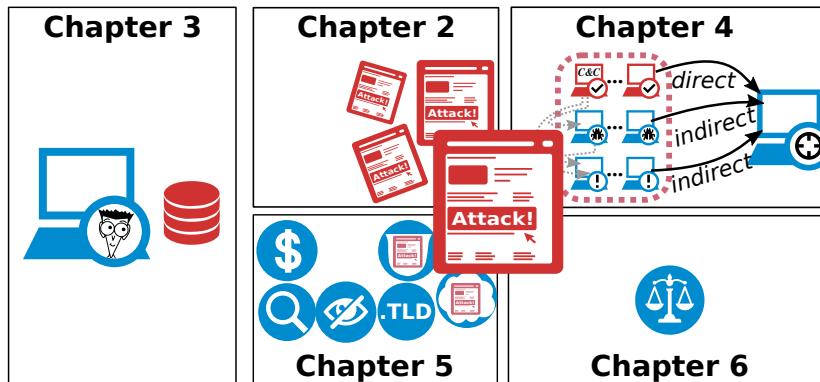


Figure 6.1: Booter ecosystem (extended from Figure 1.4).

In Chapter 2, we present a comprehensive set of booter websites, which are a public frontend containing information about the services offered by the booter and their prices and an interface for launching attacks once clients have been paid for. Then, in Chapter 3, we focus on the information of clients within booter databases. Client create accounts via the website, allowing the operator to record the target of an attack and whether the client has paid for it.

Then, in Chapter 4, we show that the attack infrastructure may contain up to three groups of machines. The first group is necessary to perform an attack, while the other two are optional. Each group may consist of dozens, hundreds, or even thousands of machines. When an attack is launched, the first machines contacted by a booter website belong to the first group. This group is the most important for the booter infrastructure, because these machines

orchestrate attacks by either performing them themselves (*i.e.*, direct attacks), or by contacting the second group of machines (*i.e.*, acting as the command and control of a botnet). If the second group of machines is contacted, the attack is indirect as the booter operator is using others computers to perform the DDoS attack.

The second group of machines consists of systems that send network traffic to the target (*i.e.*, perform attacks themselves), or contact a third group of machines. This third group of machines run ordinary Internet services (*e.g.*, DNS and NTP servers) that are misused to reflect and amplify attacks. The reflection occurs because these ordinary services do not validate the source of requests. By pretending to be the target system, attackers send traffic directly to the target. Amplification is another characteristic of the third group of machines, which attackers exploit the larger size of responses in services such as the DNS. This characteristic makes the reflected attacks stronger in volume of traffic.

Booters may offer several types of DDoS attacks, and the backend infrastructure may vary according the type of attack. For example, a booter offering reflection and amplification attacks has a backend infrastructure with machines belonging to groups A and C. In these attacks, group A machines send spoofed messages, and group C machines reflect and amplify attacks on the target.

In Chapter 5, we identify third party companies (in)directly involved with booters: payment systems, searching engines, companies that offer Whois protection information, Top Level Domain registers, Cloud-based Security Protection companies, and Web hosting providers.

6.3 Justifications for Using Booters

Booters have targeted the Internet sites of schools, personal websites, game servers, and government websites. Ironically, booter operators themselves are frequently targets for other booter clients (Chapter 3). The variety of targets suggest a range of motives for using booters, but it most appears to be some form of self-interest or malice against the target.

While booter clients do not ordinarily declare their motives for hiring DDoS attacks, it is possible to draw reasonable conclusions from the targets they attack. This task is helped considerably by the leak of several client databases from booter operators, which list the targets and frequency of attacks paid for by clients. For example, Karami and McCoy [39] conclude, in their analysis of the leaked TwBooter client database, that the most popular targets were game servers and forums. Such uses of booters deliberately interfere with how others

communicate and interact for personal gain. Malicious self-interest in limiting others' ability to communicate and play is the apparent motive in these cases.

Despite of that, there are two possibilities for using booters that do not fall into the category of malicious self-interest. The first is that booters offer a way of testing how well a service handles heavy network traffic. This is the use-case promoted by calling booters as “network stressers”. In this case, the DDoS attack is targeted at the user’s own server or to a server that the user has permission to test. The other potential justification for using a booter is based on the argument that DDoS attacks are defensible acts of civil disobedience. It is possible that the TwBooter attacks on government websites that Karami and McCoy [39] describe (two sites belonging the Indian government and the website of the Los Angeles Police Department (LAPD)) are political protests. Courts have already recognized some DDoS attacks as forms of activism. For example, the Higher Regional Court of Frankfurt in Germany recognized Andreas-Thomas Vogel’s DDoS attack on the Lufthansa website as a method of influencing public opinion [85]. Before considering this possibility, we will examine the justification for booters as a tool for network stress testing.

6.3.1 Justifying Booters as Network Stress Testing Tools

Portraying booters as a means of testing a system’s ability to handle heavy network traffic depicts them as a neutral service that can be used for both moral and immoral purposes. A network stress test performed on one’s own systems could be defended as a self-regarded action, following John Stuart Mill’s arguments for the liberty of the individual over actions that concern only herself and the *maxim volenti non fit injuria*: no-one is unfairly harmed by an action to which she has consented [14].

These justifications, that the network stress test does not harm others or that anyone else’s systems, that are used in the attack have given consent for them to be used in this way. We will return to the question of moral legitimacy later in this chapter. The separate question of whether operating a booter is legally legitimate is more straightforward: DDoS attacks are illegal in most jurisdictions, so their intended and advertised purpose is illegal [46, 85].

What if the booter operator is in a jurisdiction where DDoS attacks are legal? For the sake of the argument let us suppose that (1) there are legal uses for a booter; and (2) that it is possible to distinguish between legal and illegal uses before the attack occurs (*i.e.*, when the client requests and attempts to pay for an attack). Under these assumptions, the operator is justified in allowing attacks that meet these two criteria and the operator confirms that the client is requesting a legal use. This is the ideal “network stresser” case that many booter operators advertise to legitimize their services.

Despite the claims of booter operators, the ideal “network stresser” case does not justify operating or using their services. There are alternative methods of testing the resilience of servers and networks that do not require sending overwhelming amount of network traffic across public networks generated by booters and DDoS attacks. For instance, the network gateway connecting a server or local network to the Internet itself could generate the traffic needed to simulate a DDoS attack. While this is of course a simulation of a DDoS attack, it does not involve the abuse and exploitation of the Internet services, devices, and traffic of others (through reflection and amplification attacks) that actual DDoS attacks involve.

For the sake of the argument, let us assume that (1) a “network stresser” is necessary for testing the resilience of a server or network and (2) that there is no alternative means of performing this test. If assumptions 1 and 2 hold in this case, it would be acceptable to operate a booter. What about cases where both 1 and 2 hold, but the operator does not confirm that the request is legal? The operator might reply that it is not his/her responsibility to confirm an attack’s legality, and that this responsibility falls to the client. The booter (and by extension, the operator) is portrayed here as being morally neutral.

However, this neutrality is compromised by the second assumption that legal and illegal uses of the booter can be distinguished before it performs a DDoS attack. Given that the only morally permissible use of a booter is as a network stress test under a specific set of conditions, the booter’s frontend could be designed so that it will only accept attack requests if those specific conditions are met. For example, after receiving a request for a DDoS attack, the operator could contact the target and confirm that the request is legitimate and that the target has actually agreed to it. If the request is confirmed, then the operator could proceed with the attack and seek payment from the client. Otherwise, the operator would reject the request.

The legal illegitimacy of booters creates difficulties for the other agents that are associated with booters. In addition to the clients wishing to launch DDoS attacks and the operator maintaining the frontend and backend systems necessary to attract clients and perform attacks, there is the payment system that transfers money between the clients and the operator and the Cloud-based Protection Services (CBPS) protecting the booter from similar attacks. The payment system and the CBPS offer support services necessary for the booter to function, either directly or indirectly. The payment system is necessary for the booter to operate as a paid-for service, so the transfer of money between the clients and the operator directly supports the booter’s operation. The CBPS’s service only indirectly supports the booter’s operation as it is not a necessity for a booter to operate: while it might be a practical necessity given the frequency of booters themselves being DDoS targets, the concept of a booter

as a “DDoS-for-hire” service does not in theory entail using a CBPS.

Both the payment system and CBPS can claim that they are neutral between the uses of their services. Both have a stronger case for such neutrality than the booter operator, since both services have many clear legitimate uses. Both the payment system and the CBPS should only be involved if the two assumptions mentioned earlier hold; otherwise, any use of a booter is illegal. This suggests that both the payment system and the CBPS have a *prima facie* duty to end their relationship with a booter once they are aware of the illegality of its business. Payment system operators and CBPSs usually have acceptable use policies that prevent their services from being used for illegal purposes. These policies give a straightforward means for such services to sever their relationship with a booter operator. A booter might continue to operate without the support of a payment system or a CBPS, but it would be a less effective tool for allowing anyone to perform DDoS attacks.

Before moving on, we will briefly mention a potential conflict of interest that CBPS operators might have in this context. DDoS protection is only necessary if DDoS attacks occur. DDoS attacks will be easier to perform if booters are active and available, and demand for DDoS protection will increase accordingly. The increasing use of CBPS requires more powerful booters to effectively perform DDoS attacks. Given that booters are often targets of DDoS attacks themselves, if CBPS services did not protect booters there would potentially be fewer sources of DDoS attacks. While it is not a symbiotic relationship, since the need for CBPS services would remain even if booters disappeared, there is a potential incentive for a particularly wily booter operator to operate a CBPS as well and vice versa.

6.3.2 Justifying Booter Attacks as Civil Disobedience

We have now established that operating booters does not have any legal legitimacy, given that performing DDoS attacks is illegal. But what can be said about any potential moral legitimacy that such attacks may have? As mentioned earlier, DDoS attacks have been recognised as a form of civil disobedience. Before discussing whether booters may be used for this purpose, we need to explore the concept of civil disobedience itself.

Identifying Civil Disobedience

Civil disobedience is a contested concept, so relying on any single definition to determine whether an act is legitimate civil disobedience is problematic. However, it is possible to identify general features that acts of dissent should have if they are to be regarded as civil disobedience. Brownlee [16] notes two

reasons for resorting to illegal protest as an effective means of conveying moral or political dissent: (1) it encourages media coverage of the issue that may otherwise not occur, and (2) it allows protesters to demonstrate the strength of their convictions by showing the public their willingness to face the legal consequences for their actions.

These two reasons may be called the conscientiousness (in the sense of seriousness and sincerity of belief) and communicative features necessary for an illegal act to be a form of civil disobedience [15]. The conscientiousness of civil disobedience is the seriousness and sincerity of belief in the cause motivating the protest. The desire to act, break the law, and to face the consequences of doing so, serve as evidence of the dissident's beliefs about the injustice he/she is protesting. Illegal acts lacking this serious consideration and sincere motivation are likely to be dismissed as criminal behaviour without any political objective.

Civil disobedience is not alone in exhibiting conscientiousness. Participants in radical protest who are indifferent to damaging property and using force against persons, and who frequently attempt to avoid legal accountability for their actions, may also share this sincerity and seriousness of belief. (Radical protest without such conscientiousness is merely violent crime.) The communicative feature of illegal protest is what distinguishes civil disobedience from radical protest.

Communication is necessary to provoke the political or social response necessary to end the injustice that motivates the protest. Without effectively communicating the motivation to an audience able to bring down change, civil disobedience cannot serve its intended purpose. How this communication distinguishes civil disobedience from radical protest is that civil disobedience suggests that change is still possible within the existing social and political structure. It recognises that the existing political and legal system is capable of bringing about the desired response, even if it requires changes to these systems themselves.

Radical protest suggests that the cause motivating the protest is impossible to achieve within the current social and political structure, and seeks to replace or undermine them through coercion. It rejects the possibility that the desired response is possible through the existing political and legal systems, even if they are reformed through the existing means of making such changes. The communicative aspect of the protest is why non-violence is frequently (but not always) given as a requirement for legitimate civil disobedience, as violence and coercion may compromise the communicative features of the protest. Acts of violence may discredit the cause, motivating them by making the public reject the views of those who commit them. Civil disobedience aims to provoke debate and engage both the public and the government in considering and hopefully accepting the protesters' view; while radical protest seeks to end debate and

compel others to accept the protesters' view.

Finally, a useful distinction may be drawn between direct and indirect acts of civil disobedience. Direct acts of civil disobedience violate the law or policy that is the motive for dissent, while indirect acts do not themselves violate the laws or policies that they are protesting against [76]. Obstructing the entrances of public buildings as a protest against a state's military activities overseas is indirect civil disobedience, while obstructing the entrances to a forest to protest the logging of that forest is direct civil disobedience.

Using Booters to Perform Acts of Civil Disobedience

Based on the account presented above, a DDoS attack may be classified as an act of civil obedience if it is an illegal action that has the features of conscientious motivation and the aim of communicating dissent to the public. Unless it is protesting the illegality of DDoS attacks themselves, these actions will be indirect forms of civil disobedience [85].

We can also distinguish between protest and network integrity perspectives on DDoS attacks as civil disobedience. The protest prospective sees DDoS attacks as legitimate forms of hacktivism: political and social activism that operates primarily via the Internet. Such "mass action" hacktivism regards itself as comparable to a physical sit-in protest [37]. The protesters "create a space" for their message to be heard by interrupting or interfering with Internet traffic. Sauter [85] presents an in-depth defence of this view. In contrast, the network integrity perspective sees DDoS attacks as damaging to both the network and to freedom of expression. Rather than creating a space for protesters to convey their message, the interruption caused by a DDoS attack silences the victim. Ruffin [78] expresses this view.

The network integrity view rejects almost any usage of booters. The only possible usage for such services is as a voluntary self-directed network stress test. As mentioned earlier, current booter services fail this justification. Most booters use indirect methods to perform DDoS attacks, either through compromised systems (such as botnets) or by abusing publicly accessible services (such as DNS resolvers and NTP servers). The compromised systems operate without the knowledge or consent of their operators. It is possible for a booter to perform direct attacks if the participating systems either belong to the booter operator or the system owners choose to contribute to the attack. However, the difficulty of identifying the backend controlling the attacks (and thus identifying who is controlling the attack) makes it difficult to distinguish potentially "legitimate" direct attacks from "illegitimate" indirect ones. Using direct attacks instead of indirect ones would also be worth advertising to potential clients wishing to avoid legal concerns about using compromised computers in performing network

stress tests of their own systems. The fact that booter operators do not advertise this is revealing in itself.

Let us return to the protest perspective of DDoS attacks. As the protest perspective accepts the use of DDoS attacks as legitimate acts of civil disobedience, we need to examine whether using a booter to perform the attack compromises the conscientiousness and communication features that would otherwise make it an act of civil disobedience.

Sauter [85] qualifies her justification of DDoS attacks as “electronic civil disobedience” by emphasising the symbolic value of launching such attacks rather than the actual disruption they cause. She refers to the Starbucks website as an example, explaining that this site serves more as a poster representing the company rather than the means through which it conducts its business of selling coffee. Many government websites may also be thought of as being similar to this. The primarily symbolic nature of such attacks highlights their communicative nature. It appears straightforward to attribute this characteristic to booter-launched DDoS attacks as well.

Despite of that, a civil disobedience justification for using booters faces the problem that DDoS attacks performed through a booter are mass actions controlled by one person rather than mass actions performed by distinct individuals motivated by a common goal. Booter attacks lack the “democratic accountability” of DDoS attacks performed by a mass of individuals working together. In defending their DDoS attacks against the Internet sites of the World Trade Organization (WTO) in November and December 1999, the *electrohippies collective* stated that “[our] method has built within it the guarantee of democratic accountability. If people don’t vote with their modems (rather than voting with their feet) the action would be an abject failure.” [24] Relying on the number of individual participants for the effectiveness of a DDoS attack means that it can only succeed if large numbers of people (and thus, a large number of direct attackers) voluntarily participate. This democratic accountability becomes part of the communicative feature of the attack: it demonstrates that a large number of people was motivated to contribute to the protest.

Whether civil disobedience requires a group to perform it or whether it can describe the actions of an individual is controversial. Arendt [12] argues that civil disobedience must be the work of a group that is a minority within society. This perspective would reduce the justification for using a booter as it allows a single individual to perform a DDoS attack: it could still be a protest, but it would not be civil disobedience as such. On the other hand, the term civil disobedience itself comes from Henry David Thoreau’s (an American poet, philosopher, abolitionist, naturalist, tax resister, development critic, surveyor and historian) justification of his individual protest against the US government’s actions in Mexico by refusing to pay his taxes. For the sake of the argument,

we will leave open the possibility that an individual may perform an act of civil disobedience.

Group civil disobedience increases the public spectacle and enhances the communicative aspect of the dissent. Of course, this only holds if the group members are acting on their own belief in the cause and are sincere in their motivation. A lone protester hiring a large number of actors to participate in a sit-in may succeed in creating a spectacle that nevertheless does not indicate that he/she is the only one there motivated to act upon his/her cause. This significantly weakens the sincerity of the act: the actors inflating the size of the protest are being paid to do so, regardless of how sincere the protester herself might be in promoting her cause. With this in mind, it should be clear how using a booter compromises the conscientiousness aspect of performing a DDoS attack as a form of civil disobedience.

The additional participants of a DDoS attack that a booter provides are not motivated by a sincere conviction in the cause but by the instructions of the booter operator, who is being paid by the client to perform the attack. *Booter DDoS attacks do not exhibit the necessary sincerity and seriousness required for illegal acts to be considered acts of civil disobedience.*

Booters also obscure the attribution of the protest act, which affects the communicative aspect of using DDoS attacks as a form of protest. The client is paying the booter operator to perform the protest, and this transaction obscures the communication of the motivation for performing a civil obedience. Consider the DDoS attacks performed by the group Anonymous that began in 2010 under the name “Operation Avenge Assange”. These attacks were directed at organizations and groups that disrupted the operation of the WikiLeaks website [85, 21]. The Anonymous DDoS attacks, despite the relative anonymity of the participants, are attributable to the group Anonymous itself. The group publicized and claimed responsibility for the actions, and presented its justifications for doing so.

Those who chose to download Anonymous’ DDoS tool (*i.e.*, Low Orbit Ion Cannon—LOIC) and participated in the protest, performed direct attacks, even if the software itself was remotely controlled. The *electrohippies* went further, publicly explaining their actions and posting links to arguments for and against the WTO on their website so that potential participants could make an informed decision about joining the DDoS attack [24]. To ensure that the action effectively communicates the motivation that makes the act civil disobedience rather than vandalism or other criminal activity, the booter client or operator would have to claim responsibility for the DDoS attack and explain their reasons for doing so.

The *electrohippies* and Anonymous publicly justified their actions by appealing to political principles: anti-globalization and freedom of expression

and association, respectively. Their actions were intended to influence political policy by publicly expressing dissent with political actions. Lizard Squad, however, presented their DDoS attacks as an advertisement for their LizardStresser booter. Their motivation is self-interest, and there was no desire to influence political policy or social institutions to correct injustice.

In response to these arguments for DDoS attacks as civil disobedience, it may be said that the relatively low threshold for participating in the Anonymous DDoS attacks reduces its legitimacy as an expression of popular dissent. The argument portrays contributing to these attacks as little more than “slacktivism”, irrelevant and ineffective displays of political opinion on the Internet that express support without making a meaningful contribution or risking themselves in working towards achieving the political goal [63].

This objection appeals to the conscientiousness feature of civil disobedience: someone who has carefully considered the costs of being caught performing illegal activity and still does so based on political or moral conviction displays the seriousness of their motivation [15]. If legal consequences are unlikely and the effort required to participate is minimal, a protester may be less serious in her commitment to the belief motivating the protest.

This objection can be challenged by arguing that it appeals to a conception of civil disobedience that is too beholden to historical cases [85]. Participating in a DDoS attack may have legal consequences, as the later arrest of some of those involved in the Anonymous DDoS protests demonstrates [21]. However, even if we accept the charge of *slacktivism* it still grants the Anonymous protests a degree of legitimacy that booter attacks do not share because the consent of the computer users involved. The difference between a user knowingly participating in a DDoS attack and a user’s system participating in a DDoS attack without his/her knowledge and consent is enough to undermine the legitimacy of a booter-controlled indirect DDoS attack as an act of civil disobedience.

Returning to the distinction between direct and indirect DDoS attacks will further clarify this point. The most basic direct DDoS attack is a large number of people connecting to one website and continually trying to reload it. Websites that are unprepared for a link from a highly visible and active Internet community often suffer this unintended side effect of such exposure. All the participants in such attacks, intentional or otherwise, are direct attackers as they control the computers attempting to connect to the target. A single user or group (whom we will call the “DDoS attacker”) can control a direct attack if they have been voluntarily granted access to other computers, such as by others voluntarily installing software that gives the DDoS attacker such access to their computer and are aware of the purpose that it will be used for. This is what those who downloaded and installed the LOIC software to contribute to the Anonymous protests did: they volunteered computer time and

capability to the purpose of the protest. In this sense, they are little different from people who donate money or resources to activist groups to support their cause. Participating in an active DDoS attack therefore is more than mere “slacktivism”, unless the term should also cover those who make donations to activist groups without otherwise participating in their activities.

Indirect attacks are a different matter. Here the software giving the DDoS attacker access to a computer has been installed without the user’s knowledge or agreement, and he/she is unaware that her computer is contributing to the attack. The DDoS attacker has appropriated the user’s computer time and capability without his/her permission. A user volunteering her computer for use by a DDoS attacker can withdraw her system by refusing to participate or by removing the software if she disagrees with the purposes of the attack. Unwitting participants in indirect attacks have no such option unless they discover the software running on their computers themselves and remove it (and even then, they are unlikely to know for what purpose their computer has been used for). Utilizing indirect attacks also fails the claim of democratic accountability, as those whose systems are used are unaware of their participation and their contributions to the DDoS attacks cannot be interpreted as protests on their behalf.

Sauter [85] rejects using indirect attacks in DDoS protests for similar reasons: “The use of someone’s technological resources without their consent in a political action, particularly one that carries high legal risk, is a grossly unethical action.” Sauter [85] also rightly argues that using indirect attacks damages the legitimacy of voluntary contributors to DDoS attacks by making it easier to disregard the protest as merely a criminal act rather than a political one. It obscures the communicative features of the protest by violating the rights of the users of compromised computers to control the usage of their own computers, exposing unwitting participants to legal risks (and raising fears that one may have unwittingly contributed to the protest), and by calling in question the voluntariness of the contributions to the protest.

There is a final possible argument in defence of using a booter for civil disobedience. Direct DDoS attacks performed manually by activists working in concert are increasingly unlikely to disrupt well-resourced websites [85]. It can be said that using a booter is necessary to achieve the scale of DDoS attacks necessary for the action to be noticeable, especially given the inequality of resources that governments or multinational corporations have compared to activists. A group of activists might use a booter (or several booters) to perform a DDoS attack that would be more effective against a well-resourced target than if they relied solely on direct DoS attacks from their own computers, like the *electrohippies* protest against the WTO. This partially addresses the objection of democratic accountability, as it is the action of multiple individuals rather

than just one. It is still weaker in its communicative aspect than a direct DDoS attack, however. The effect of the protest is multiplied by using a booter, and so does not convey the same breadth of concern and sincerity among the public that a direct DDoS attack of the same scale that requires a significantly greater number of participants would do.

Using a booter to ensure the effectiveness of their attack as a protest is also compromised by the fact that the booter operator is being paid for her role in performing the attack, which weakens the claim that those involved all share a sincere political motivation. Such an action might be more effective, but it would only be legitimate if the booters all used direct attacks (so that no computers are used without permission) and both the clients and booter operators publicly announced the motivation and claim responsibility for the action. The operators' public announcement of attribution and support is necessary to show that they are acting from mixed motives: both the political motive in support of the client's actions and the business motive of charging for a service. If the booter operator allowed the protester(s) performed DDoS attacks on a specified target for free, it would prevent the operator's profit motive from diluting the communicative aspects of the DDoS attack. The operator would effectively be acting as a fellow protester, or at least as a supporter by donating resources to his/her cause.

This argument helps to clarify that the protesters are not trying to hide the fact that the effect of their protest is greater than it would have been if they had only used individual direct attacks. The requirement of sincere political motivation is further strengthened if the booter operators donate the use of their service to the clients, making them contributors and supporters of the protest. In effect, donating their infrastructure for the protesters' use makes the booter operators protesters as well. This case offers the possibility for booters to be legitimately used for civil disobedience, provided that it employs only direct attacks that use computers and systems that the booter operator has legitimate control over. Otherwise, the objections to using indirect attacks in DDoS protests and the difficulties they raise for the communicative and conscientiousness features of civil disobedience still hold.

However, the need to use booters to ensure the necessary scale to effectively disrupt a well-resourced target is itself questionable. The attack's effectiveness is secondary to the attack's visibility, including both the publicity surrounding it and that someone (or some group) was motivated to perform it. As civil disobedience intends to draw attention to injustice, publicizing the act and explaining the motivation behind it is vital. Without raising public awareness of the injustice motivating the act, acts of civil disobedience cannot prompt the social or political change they aim to achieve.

In terms of DDoS attacks as civil disobedience, the success of the attack

itself is often secondary to the publicity generated for the motivation behind it [85]. Performing a massive DDoS attack, such as that possible by utilizing several booters to attack a target simultaneously, might be counter-productive as the publicity gained by the impact of the attack (including the use of others' computers without permission, if indirect attacks are involved) might overwhelm the publicity for the motivation behind it.

A parallel can be drawn here with violent protest (radical or otherwise). When violence (planned or otherwise) occurs during a protest, public reporting of the protest often portrays the violence as undermining the legitimacy of the protesters' motivation. As Sauter [85] rightly notes, any public protest faces the challenge of drawing public and media attention to the motivation for the protest, rather than the protest itself.

A DDoS attack that makes a major Internet service unavailable must be careful in announcing and promoting its motivation to ensure that it is not overshadowed by reports and interest in the disruption it causes. While the democratic accountability of direct DDoS attacks launched by enough motivated individuals to impact a major Internet service may mitigate this (since the large number of people motivated to join the attack is itself noteworthy, and the motive for doing so is likely to be publicized by the individuals themselves), an attack of similar size that relies on booters lacks the noteworthiness of a large number of people attacking in concert for a political or social goal.

6.4 Concluding Remarks

Booters are a serious threat to any system connected to the Internet. We describe, however, one argument in which booters could be legally and morally justifiable: when a booter is an ideal "network stresser". In this case, attacks are used against a target that has given permission for a DDoS attack to be performed against it, and the booter only performs direct DDoS attacks (in contrast to indirect attacks exploiting third-party Internet services or using compromised systems). Nonetheless, based on observations of current known booters and their attacks (from Chapter 4), we conclude that this case is unlikely to happen.

In terms of a moral justification for using a booter, there is little to say in defence of using booters out of self-interest. A DDoS attack, by definition, attempts to disrupt the target's ability to communicate with others. It deliberately prevents the target from interacting with others via the Internet without the legitimate authority to do so. Civil disobedience offers a possible justification, provided that the DDoS attack demonstrates the seriousness and sincerity of the protester's motivation and attempts to communicate the political

or social change that motivates his/her.

However, it is important that the systems used in a booter attack that is intended as civil disobedience are direct attackers. Using indirect attacks, either by abusing publicly accessible servers or by gaining control by compromising systems, attenuates the communicative aspect of the attack. Then there are problems of conscientiousness in motivation and attribution, as the clients are paying the booter operator to perform the attack on their behalf. Using a booter to perform civil disobedience is morally justifiable only if the booter performs direct attacks, and the clients and operators publicly announce the attack and their motivations.

With the discussions and conclusions in this chapter, we expect to foster legal actions against booter operators and clients.

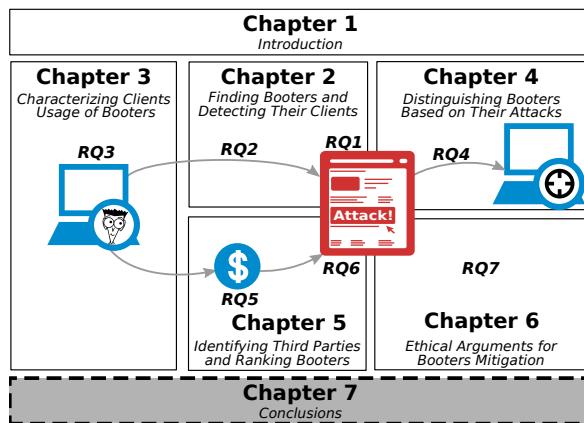
"The present is never our goal. The past and present are our means. The future alone is our goal."

—BLAISE PASCAL
IN: PENSÉES, 1960

CHAPTER 7

Conclusions

In this last chapter, we reflect the main findings and contributions of our research in this thesis. In addition to that, we discuss future directions for the investigation of booters and DDoS attacks.



The organization of this chapter is as follows:

- In section 7.1, we summarize the research presented in this thesis;
- In section 7.2, we revisit our research questions to highlight details of our findings;
- In section 7.3, we discuss the future of booters and DDoS attacks.

7.1 Summary

In Chapter 1, we discussed the damage that DDoS attacks have caused to our online society since the beginning of the Internet. After that, we presented the evolution of DDoS attacks over five periods (1982–2000, 2000–2003, 2003–2009, 2009–2012 and 2012–2017). Then we highlighted the characteristics of the period between 2012 and 2017, called the booters period. This thesis is about this period.

As we outlined in Chapter 1, the main difference between the booters period and the other two periods is related to the ability to identify attackers and attacks. In contrast to the other two periods, we observed that booters expose their operations not only to potential clients but also to anyone on the Internet, such as the research community or law enforcement agencies. The reason for this exposure is that booters *publicly* offer and facilitate the launching of DDoS attacks. Because of booters' public presence, we believe there is a window of opportunity to identify the stakeholders involved with booters. We defined our goal as being to support mitigation actions by understanding booter websites, their clients, the infrastructure used to perform attacks and third-party companies (in)directly involved with booters.

To address this goal, we defined seven Research Questions (RQ). Before we could begin investigating clients, attack infrastructure and third-party companies, in RQ1 (*how to find booters?*), we focussed on finding the booters. In RQ2 (*how to detect clients accessing booters?*) and RQ3 (*how do clients use booter services?*) we focussed on the clients who access and launch attacks using booter websites. In RQ4 (*do booters have distinct attack characteristics, and if so, what are these characteristics?*), we focussed on the attacks, while in RQ5 (*what third-party companies are used by booters?*) we focussed on the third-party companies that are used by booters to maintain their DDoS attack businesses. Then, in (*which booters are most dangerous?*), we focussed on highlighting booters that had not previously been affected by mitigation action but pose serious threats to online systems. Finally, in RQ7 (*which ethical arguments can be used to support mitigation actions against booters?*), we use legal and ethical arguments together with findings from previous RQs to support legal action against booter operators and clients.

To answer the research questions, we used measurement-based approaches. These approaches were composed of *methods* for automatic (or semi-automatic) collection and analysis of datasets both produced by us and retrieved from private and public sources.

7.2 Revisiting Research Questions

To provide a more detailed view of the methods and findings in this thesis, this section revisits each of the research questions defined in Chapter 1.

RQ1: *how to find booters?*

In Chapter 2, entitled Finding Booters and Detecting Their Clients, we addressed RQ1. We answered this research question by using a three-step approach that enabled us to find a total of 435 botter websites. In the first step, based on the observation that booters tend to offer their services on the public Internet (to attract their main clients), we used Google's search engine to find any websites suspected of being a botter. The search terms used were botter, stresser, DDoSer, DDoS-for-hire, and DDoS-as-a-service. After step one, we could not be certain that all the websites we found and suspect of being booters were actually booters. Therefore, we refined our results in the second and third steps.

In the second step, for each suspected URL we collected the characteristics most frequently used in website classification approaches. By performing a systematic analysis (based on odds ratios), we found that only 15 characteristics are actually relevant in differentiating booters from non-booters. Among these 15 characteristics, the top-three were (1) the number of (sub)pages that a website had, (2) the number of links to external websites and (3) the age of the domain name.

In the final step, we classified the suspected botter websites, using the 15 characteristics, and compared them to a training data set. We then evaluated seven classification methods (*i.e.*, Euclidean distance, Sq. Euclidean distance, Manhattan distance, Cosine distance, Fractional distance, k-Nearest Neighbours and Naive Bayes) to define which was the best method to classify botter websites. Finally, we used a machine learning algorithm to improve the results of the best classification method (Cosine distance). In the end, we achieved 95.5% confidence in the classification of botter websites given a set of suspected URLs.

From observations by us and others [51, 41], we know that booters are quite dynamic, that is new ones frequently appear and others frequently cease to exist. The Internet community is under constant threat from new booters. Thus, we decided to create an initiative called Booter Blacklist (available at <http://booterblacklist.com>) that frequently updates and openly shares the most comprehensive list of booters (collected using the approach outlined above) to aid the community in defending itself. The botter list has three key usages: (1) identify booters, (2) monitor users accessing botter Websites and (3) monitor

the booter market.

RQ2: *how to detect clients accessing booters?*

In Chapter 2, we monitored network traffic using a list of booters (obtained from RQ1). We showed that traditional network monitoring, based on observing users accessing booters' IP addresses, is not possible. The reason is that the majority of booter websites (more than 90% in 2016) use the IP of Cloud-Based Security Providers (such as CloudFlare). Thus one IP address points to several websites and not just to the booter. We therefore, in collaboration with the Dutch NREN (SURFnet), used passive DNS monitoring, in which we collected DNS requests from clients to a booter within our list. Using this approach we found hundreds of users accessing booters.

This approach, based on passive DNS, does not detect booter clients who used third-party DNS resolvers (instead of SURFnet's resolver). However, we assumed that only a minority of booter clients have the technical knowledge to change this type of configuration on their devices. This assumption was confirmed while answering RQ3.

RQ3: *how do clients use booter services?*

After collecting a list of booters and identifying clients accessing them, in Chapter 3, we focused on understanding the behaviour of their clients. We answered RQ3 by analysing (leaked and publicly available) booter databases that contain client information. We developed a semi-automated analysis method and analysed 15 booter databases.

Based on our analyses, we observed that the majority of clients paid less than US\$10, performed attacks of less than five minutes duration, targeted a handful of URLs or IP addresses and accessed booters using single IP addresses. We also observed that the majority of clients do not obscure their access via a Virtual Private Network (VPN) or The Onion Router (TOR) network. In addition, we observed that the email addresses clients registered on the booter websites were often found on social networks and online hacker forums. Based on these observations, we conclude that the majority of booter clients can be identified by their registration email address combined with the IP address(es) from which they accessed booter websites.

While analysing booter databases, we observed that all databases have records of clients attacking a target located in the same country from which they were accessing the booter. We concluded that booter databases that contain records of clients can facilitate legal action from law enforcement agencies

(within a particular country), which can immediately lead to legal action against the owner and clients of booters.

We also observed a small set of very active clients who performed a significant number of attacks (outliers). We also noted that booter owners who have previously been prosecuted are among this small set of clients. For example, the owners of vDOS booter were the ones that performed most attacks (revealed by their database) [49]. Therefore, we conclude that in addition to financial gain, booter owners have a level of self-interest in running booters for attacking third-parties.

RQ4: do booters have distinct attack characteristics, and if so, what are these characteristics?

In Chapter 4, entitled Distinguishing Booters Based on Their Attacks, we addressed RQ4. Based on the observation (from Chapter 1) that anyone can become either a user or a target of booters, we subscribed to the most popular booters (in 2013) and launched attacks against our (controlled) network infrastructure. Our analysis showed that the set of attack sources used by different booters combined with some specific characteristics of attacks (*e.g.*, DNS requests used for achieving more attack power) was sufficient for distinguishing between booters with different owners.

Besides distinguishing booters based on their attack characteristics, we observed that booters delivered less power in their attacks than they advertised on their websites. We strongly believe booters are likely to have the attack power they advertise. The main reasons are: (1) users share the attack resources of booters, (2) users can only see their target's status (online or offline), thus for a booter it is meaningless to deliver more power than enough to make a target go offline and (3) we speculate that booters want to 'stay under the radar' by not generating very large attack volumes (*i.e.*, 100 Gb/s and up), as these are much more likely to attract the attention of network operators, security specialists and law enforcement agencies.

In addition to these observations, we noticed that (1) booters (mis)use far fewer attack sources than would be easily possible if they used all available Internet resources (*e.g.*, UDP services and vulnerable devices), (2) there is little overlap between the attack sources used by different booters and (3) the number of vulnerable Internet systems is growing rapidly. These observations lead us to conclude that booter attacks have the potential to cause much more damage than they currently deliver, for example by combining their sources or adding more misused systems.

All the hundreds of attacks traces that we collected during our research are publicly shared at <http://ddosdb.org> (DDoSDB). We hope that publicly

sharing booter attack traces will help the security community to gain a deeper understanding of actual attacks and test and methods of preventing booter attacks, such as detection and mitigation solutions.

RQ5: what third-party companies are used by booters?

While investigating which hosting providers were most used by booters (in Chapter 4), we found that booters subscribed to services from Cloud-Based Security Providers (CBSP). These CBSPs hide the IP addresses of their customers (such as booters) through a proxy-like solution. Therefore, in Chapter 4, we were not able to reveal booters' hosting providers because we were not able to discover booter IP addresses. Then, in Chapter 5, we discovered the actual booter IP addresses and analysed not only their hosting providers but also third-party companies (in)directly used by booters (RQ5). We answered RQ5 by analysing third-party companies identified via the IP addresses or the domain name information related to booter websites.

Using this method we observed that it is not only web-hosting companies and Cloud-Based Security Providers that are (in)directly used by booters, but also Top Level Domains, Domain Registrars, web-hosting companies, Payment Systems and search engines. Besides identifying companies used by booters, we also discussed mitigation actions that these third-party companies could take against booters. For example, we observed that if only the companies Enom, GoDaddy, and Namecheap decided to act against booters, around 50% of all booters would be affected.

We analysed not only the mitigation actions these third-party companies could take against booters but also the implications of these actions. We concluded that no action by a single third-party can solve the entire problem of booters. However, if some of the mitigation actions identified by us were deployed, we would certainly see a decrease in booters' operations. For example, a co-ordinated action by PayPal against dozens of booter owners resulted in a (temporary) reduction in attacks.

RQ6: which booters are most dangerous?

After identifying the third-party companies and mitigation actions they could take against booters, we focused on identifying which booters (among the hundreds found from RQ1) we need to prioritise to be taken down. Considering that booters are likely to have the attack power they advertise on their websites (from RQ4), we proposed five metrics to determine the danger level of booters: (1) the popularity level of booter websites, (2) the price charged, (3) the max attack power advertised, (4) the date when the domain name was created and

(5) the date when the domain name will expire.

We applied these metrics over a three-month period (between 1st November 2016 and 1st February 2017), and found 33 booter websites to focus on. These booters ranked among the top-1% of most accessed domain names on the entire Internet; Then from these 33 we identified eight booters that pose the greatest threat to online systems. These eight booters offered attacks with a rate of 100 Gbit/s (and up) and their prices started from US\$10 or less.

Although it is not 100% clear which part of the booter problem would be solved if these eight or thirty-three booters were taken down, we observed, however, that legal actions against the most popular booters have had a considerable impact on the overall picture of the problem. For example, the prosecution of owners and clients of three large booters (Twbooter, Panda Booter, and vDOS) showed a temporary clear reduction of attacks.

RQ7: which ethical arguments can be used to support mitigation actions against booters?

We observed that third-party companies on which booters rely (identified while answering RQ6) all require some sort of order from law enforcement agencies before they will take action against booters (these companies include CloudFlare and PayPal). However, up to early 2017 (the conclusion of this thesis), questions remain over the legality of booter services and booter users launching attacks. Therefore, in Chapter 6, entitled Ethical Arguments for Booters Mitigation, we collaborated with experts from the area of ethics to answer RQ7.

Together with the Ethics department at the University of Twente we performed an extensive analysis of the ethical arguments for and against performing and providing DDoS attacks. We concluded that there is only one scenario in which booter attacks can be classified as ethically acceptable. This hypothetical scenario consists of booters being a conceptual “stress tester” and the attacks being claimed as an act of civil disobedience.

However, based on the observations from Chapters 2, 3, 4 and 5, this hypothetical scenario is not likely to occur, given that: their attack infrastructure is composed of misused machines; the attacks are performed without the consent of the target and the attacks are not declared as an act of civil disobedience (before or during the attack). Therefore, there is no ethical justification for operating or using booters. We hope that this conclusion fosters legal action against booter clients and owners in the coming years.

7.3 Moving Forward from Findings

In this thesis, we studied the booter ecosystem in its entirety, learned their characteristics and used these characteristics to propose mitigation actions. The natural next step is to employ the knowledge provided in this thesis towards legal actions against booter owners and users, which would eventually reduce the number of DDoS attacks. Akamai [9] reported, in the first quarter of 2017, a decrease of 17% in the total number of attacks and a reduction of 83% in attacks greater than 100Gb/s. In line with our view of what the next steps should be, Akamai stated that the reason for this decrease is the following:

“There is one factor that seems to be affecting the DDoS landscape as a whole: law enforcement. Early attacks by the Mirai botnets appear to have been triggered by the announcement of the arrests of two teens in Israel who were responsible for the vDos botnet—a DDoS-for-hire tool [booter] that netted them hundreds of thousands of dollars. More recently, Europol coordinated the arrest of 34 individuals across 13 countries as part of an effort called Operation Tarpit. Operations like Tarpit target the largest [booter] services [and their clients] responsible for DDoS attacks directed at banks, gaming companies, and retailers. This can have a significant effect on reducing the number of attacks on these organizations.”(Akamai [9, pp 2])

More law enforcement action would force booters to change, for example by moving their business from the public Internet to the underground market and towards utilisation of crypto-currencies. As a consequence it is likely that booters would lose their main clients (layman users). Therefore, a follow-up step that should be investigated is how to trace attack perpetrators in the coming DDoS attack phase (back to the underground market using crypto-currencies).

We agree with Jonathan Polnay [93], the UK prosecutor of a booter owner, who said in court: “where there are computers, there are attacks”. In our view, it is clear that DDoS attacks will not end! Our dependency on online services will only increase over time. Consequently, the financial and social damage caused by DDoS attacks will also increase. Where there is demand, there will be someone to offer a service. On top of that, we believe that attack sophistication will increase, mainly in the form of hiding attack perpetrators and increasing the power of attacks. We believe that attackers will exploit more daily devices connected to the Internet (also known as Internet of Things devices, such as TVs, fridges and vacuum cleaners) and also mobile devices. We also believe that more attacks will be based on Internet Protocol version 6 (IPv6), which will make it even harder to trace the source of attacks. We also believe that cloud services

will be exploited more, because they facilitate access to on-demand network infrastructure resources.

Finally, we would like to highlight three others directions for future work. First, more support for enforcement of Best Current Practice (BCP) as proposed by technical groups from the Internet Engineering Task Force (IETF), which regularly identifies problems and discusses the most suitable ways to implement modifications. For example, the BCP38 [27] on Network Ingress Filtering is an efficient practice to defeat DDoS attacks which employ IP source address spoofing. This BCP motivated the Center for Applied Internet Data Analysis (CAIDA) to create the Spoofed project (<https://www.caida.org/projects/spoofed>), an open-source software to assess and report on the deployment of BCP38.

A second direction for future work is to improve the production and sharing of DDoS attack intelligence, such as attack fingerprints applied to Network Intrusion Detection Systems (NIDS). The DDoSDB initiative, proposed in this thesis, is a promising candidate to facilitate open sharing of this DDoS attack intelligence.

A third direction for future work is to improve DDoS mitigation solutions, focussing on redundancy and resilience of services. DNS and IP anycast are promising candidate technologies that can be used to redirect attacks and balance the traffic load of a target system under attack.

 APPENDIX A

List of URLs Containing Booter Databases Dumps

In this appendix we present the URLs from which we found Booter databases, used in chapter 3.

Table A.1: URLs in which we found Booter databases dumps.

| # | Alias | URL |
|----|------------------|---|
| 1 | 212booter | http://4lz5rmnk6f63tmm.onion/db/212booter.sql |
| 2 | superstresser | http://4lz5rmnk6f63tmm.onion/db/superstresser_db.txt |
| 3 | nullbooter | http://pastebin.com/L0miVqSB |
| 4 | stealthstresser | http://4lz5rmnk6f63tmm.onion/db/stealth_stresser.sql |
| 5 | flashstresser | http://4lz5rmnk6f63tmm.onion/db/flashstresser.sql |
| 6 | notoriousbooter | http://pastebin.com/L0miVqSB |
| 7 | hazardstresser | http://www.bitleak.net/Thread-Hazard-Stresser-Database-Dump |
| 8 | stealthstresser2 | http://4lz5rmnk6f63tmm.onion/db/stealthstresser.sql |
| 9 | pokeboot | http://ge.tt/7Qsb4ZU/v/0 |
| 10 | nationalstresser | http://pastebin.com/EpwVqbbh |
| 11 | vaporizedbooter | http://pastebin.com/ccfdEF2p |
| 12 | pandabooter | http://pastebin.com/WrB63sbA |
| 13 | pandabooter2 | http://pastebin.com/0Vddej44 |
| 14 | legionbooter2 | http://pastebin.com/fQb2UwH0 |
| 15 | galaxybooter | http://4lz5rmnk6f63tmm.onion/db/galaxybooter.sql |
| 16 | xrhostbooter | http://4lz5rmnk6f63tmm.onion/db/xr_hostbooter.sql |
| 17 | bootertw | http://krebsonsecurity.com/wp-content/uploads/2013/03/booter.7z |
| 18 | legionbooter | http://4lz5rmnk6f63tmm.onion/db/legion_2013_08_16.sql |
| 19 | vstresser | http://4lz5rmnk6f63tmm.onion/db/vStress.sql |
| 20 | urgentbooter | http://4lz5rmnk6f63tmm.onion/db/urgentbooter.sql |
| 21 | panicstresser | http://4lz5rmnk6f63tmm.onion/db/panicstresser.sql |
| 22 | jaysbooter | http://4lz5rmnk6f63tmm.onion/db/jaysbooter.sql |
| 23 | vddos | https://t.co/gfK3VdR0zn |

APPENDIX B

List of URLs From the Booter Blacklist Initiative

In this appendix we present the list of URLs retrieved from <http://booterblacklist.com> on 16-July-2017.

Table B.1: List of booter URLs retrieved from booterblacklist.com.

| | | |
|-----------------------------|------------------------|-------------------------|
| 1606-stresser.net | bestresser.net | carnagestresser.com |
| 9yrbryfyd.esy.es | beststresser.com | celerystresser.com |
| absolut-stresser.net | bezobooter.webs.com | centexbooter.com |
| acidstresser.net | b-h.us | chargeon.cf |
| agonyproducts.com | bigbangbooter.com | cmdbooter.ml |
| alien-stresser.com | bogabooter.org | cnstresser.com |
| alphastress.com | boot4free.com | cobra-api.com |
| ambushproducts.com | booterddos.890m.com | connectionstresser.com |
| ambushstresser.info | booter.eu | cpubooter.com |
| america-stresser.us | booterfull.com | crazyamp.me |
| animebooter.net | booter.in | critical-boot.com |
| annonytstresser.com | booter.io | critical-stresser.com |
| anonclan-stresser.net | booter.ninja | cryptostresser.com |
| anonmafiastresser.com | booter.org | crystalstresser.com |
| anonymousbooter.com | booter-sales.hourb.com | cstress.net |
| anonymoussddos.eu | booter.tw | cyber-sst.com |
| anonymous-stresser.com | booter.xyz | cyberstresser.org |
| anonymous-stresser.net | boot.lu | darealbooter.net |
| anxy-stresser.com | boot.ml | darkbooter.com |
| apocalypse-solutions.com | bootr.org | darkbooter.fr |
| apocalypsestresser.webs.com | bootyou.xyz | darkbooter.org |
| asylumstresser.com | bullstresser.com | darkmethods.info |
| aurastresser.com | bullstresser.ovh | darkstresser.info |
| avanzatostresser.com | buybooters.com | darkstresser.net |
| avengestresser.com | buuddos.com | darkstresser.nl |
| baby-booter.com | buz.bugs3.com | darkstresser.org |
| battle.pw | buzzbooter.fr | darkstresser.weebly.com |
| bemybooter.eu | buzzbooter.info | darkunion-booter.net |
| bestresser.com | campingwithkiddos.com | databooter.com |

LIST OF URLs FROM THE Booter BLACKLIST INITIATIVE

| | | |
|---|--|---|
| ddosapi.co.uk ddos-block.com ddosbouncer.com ddosbreak.com ddos.city ddos.click ddosclub.com ddoscover.com ddoseminc.com ddoser.pw ddoser.xyz ddos-fighter.com ddos-him.com ddos-ip.com ddosit.net ddosit.us ddos.kr ddos-monitor.ru ddosnow.com ddospower.com ddos-service.so ddossite.com ddos.space ddostest.me ddostheworld.com ddos.tools deadlyboot.net dedicatedstresser.net defcon.pro dejabooter.com deluxestresser.com demonstresser.eu denial-stresser.com destressbooter.com destressnetworks.com devicestresser.net devilstresser.net diablastresser.info diamond-dstresser.com diamond-dstresser.net diamond-dstresser.pw diebooter.com diebooter.net divinestresser.com divinestresser.info dmbooter.net dns-ddos.net downboot.xyz downloaddosgamesfree.com down-stresser.com down-stresser.us downthem.org | dream-stresser.com dreamstresser.com dstresser.net ebolastresser.com ebooter.5gbfree.com elite-booter.net elyxa-stresser.net emaizstresser.net emo-stresser.com epic-stresser.com equinoxstresser.net equivalent-stresser.net eraservices.co erast.pw eternal-stresser.com eternalstresser.pw evilbooter.net exclusive-stresser.com exclusivestresser.net exhilebooter.net exile-stresser.net exitus.to exostress.in exotic-stresser.com expedientstresser.com exploitstresser.org exresolver.jouwweb.nl fagstresser.net fatal-stresser.com fbi-stresser.eu flashstresser.net foreverinfamous.com formalitystresser.com fpsfuture.info frankgijsgang.nl freebooter4.me freebooter.co free-boot.xyz freestresser.net freestresser.xyz freezystresser.nl frozenstresser.net getsmack.de ghoststresser.com gigabooter.com globalstresser.net grimboot.com grimbooter.com h4x-stresser.us hazebooter.com heavystresser.com heddos.net | hexstresser.net hoodstresser.xyz horizon-stresser.eu hornystress.me hydrostress.com hydrostress.net hyperstresser.com iceberg-stresser.com iddos.net illuminati-products.net imbeingddosed.com imsocool.info inboot.me infectedstresser.com infectedstresser.net instabooter.com instinctproducts.com ionbooter.com ipstresser.co ipstresser.com ipstressers.net ipstressertest.com iridiumstresser.net isitdownyet.com jedistresser.com jetbooter.com jitterstresser.com kappastresser.nl kenkastresser.com kidstresser.com kryptonite.pw kryptonstresser.com k-stress.pw kth-stress.tk kushbooter.org layer-4.com layer7.pw legendboots.tk legionboot.com legionbooter.info lexsk-stresser.fr lifetimeboot.com lifetimes.pw logicstresser.com logicstresser.net luckybooteronline.altervista.org lunarstresser.com mafiastresser.com magmastreeter.com masterboot.net masterstresser.com maximumstresser.com |
|---|--|---|

| | | |
|---|--|---|
| maxstresser.com mercilessstresser.com metro-stresser.ga microstresser.net minecraftstresser.com mini-booter.com most-booter.info mystresser.com myutro-stresser.com national-stresser.com national-stresser.net nemesisbooter.com neptunestresser.com netbooter.info net-boot.net netbreak.ec netspoof.com netspoof.net netstress.net network.apocalypse-solutions.com networkstresser.com network-stresser.net networkstresser.net networkstresser.org network-stressing.net neverddos.com nfuze.cf nightlystresser.ml nightstress.net nismitstresser.net nitrousstresser.com notoriousbooter.com nuclearipstresser.com nuke.pe.hu nullednetwork.com obeystresser.com ocstresser.com olympusstresser.org omegastresser.com omega-stresser.us onestress.com onestresser.net onionbooter.com onionstresser.com onlinebooter.net opaquebooter.weebly.com optimusstresser.com orcahub.com parabooter.com payperboot.net pbooter.com phoenixstresser.com | pineapple-stresser.com piratestresser.com pokent.com powerapi.pw powerdos.co.uk powerstress.com powerstresser.com prevail.pw primestresser.pw privateroot.fr private-stresser.com pulsebooter.net purestress.net quantumbooter.net quantumstresser.com quantumstresser.net quezstresser.com ragebooter.com ragebooter.net raidstresser.com rav3nstresser.net rawlayer.com reafstresser.ga realystresser.com rebel-security.com rebornstresser.net red-stresser.com refinedstresser.net rebitch.com renegade-products.net respawn.ca restartstresser.com restricted-stresser.info riotstresser.com routerslap.com royalbooter.de safestresser.com securitystresser.net sexybooter.net shamar.hol.es sharkstresser.com signalstresser.com silence-stresser.com silverstresser.com skidbooter.info skidstresser.net skypebooter.com skypestresser.com skystresser.com smokestresser.com snow-services.com snowstresser.com | snowstresser.net spboot.net speed-stresser.com spoofedboot.com stagetresser.com starkstresser.com stealthstresser.info stormstresser.co stormstresser.net str3ssed.me str3sser.com stressboss.net stressed.pw stressem-networks.pw stressem.ninja stresserbooter.com stresser.cc stresser.club stresserddos.com stresser.in stresser.info stresserit.com stresser.network stresser.org stresser.ru stress-me.fr stress-me.io stress-me.net stress.so strong-stresser.com stuxstresser.com superstresser.com superstresser.net synstress.net talkshitgethit.comli.com terminalstresser.net teslabooter.com thebooter.co thestresser.com time-stresser.pw tingboot.info titanbooter.net titaniumbooter.comuv.com titaniumbooter.net titaniumstresser.net topstressers.com ts3booter.net ufa-booters-tools.com umbstresser.net unob.ninja unseenbooter.com ustress.co |
|---|--|---|

LIST OF URLs FROM THE Booter BLACKLIST INITIATIVE

| | | |
|-----------------------|--------------------|--------------------|
| v3rmillionbooter.info | webstresser.co | xplodestresser.pw |
| vastresser.ru | westsidebooter.com | xr8edstresser.com |
| vbooter.com | wickedstresser.com | xrshellbooter.com |
| vbooter.org | wifistruggles.com | xrstresser.net |
| vdos-s.com | wifistruggles.net | xtremebooter.com |
| vdoss.net | wifistruggles.org | xtreme.cc |
| vegastresser.net | wifistruggles.pw | yakuzastresser.com |
| vengeancestresser.com | wifistruggles.us | youboot.net |
| vex-stresser.net | wnddos.com | z7inc.com |
| vietbooter.com | wrtu-stresser.com | zenstresser.net |
| vps-booter.com | xbloutlawz.info | zstress.info |
| vpstresser.com | xboot.net | zynenstresser.us |
| webbooter.com | xenon-stresser.com | |

APPENDIX C

Open Dataset Management

In this appendix, we provide information to access the datasets and the source-code used in this thesis. We follow best practices for releasing open access data [13]. This means that we provide information (in academic papers) of each dataset, how the data was collected, the period over which data was collected and the vantage point from which the data was collected. We release datasets under a Creative Commons License that allows others to use the data for non-commercial purposes, allows them to share the data under the same conditions and requires attribution. Furthermore, we clearly state attribution requirements for when the datasets are used; typically this requires citing the paper that the datasets were collected for.

When collecting datasets, it is important to preserve the integrity of scientific results and to facilitate reproducibility of results. Careful curation of results allows other researchers to independently validate the results of research. We have strived to release all data related to this thesis as open access. All measurement data collected for the research in this thesis is publicly available in open access repositories to facilitate reproducibility. In addition to this, we have made the papers that form the basis of this thesis available in the institutional open access repository¹. Finally, the source-code used to perform measurements and analyses for this thesis have been released as open source software.

Table C.1 lists the datasets per chapter of this thesis. The first column lists the chapter, the second presents a brief description of the dataset, the third and the fourth present the URLs to the dataset and to the source-code that produces or analysed the dataset, respectively. Only one dataset used in our thesis (Chapter 2.5) is not made public. For privacy reasons, SURFnet requested to do not make the data public. In case of researching interest, SURFnet is willing to provide access to the data upon request. On Chapter 3 and 5 the dataset is available in the same URL where the source-code of the collection and analysis is presented.

¹<https://doc.utwente.nl/>

Table C.1: Per-chapter datasets and source-code URLs.

| Chapter | Brief Description | Dataset | Source-code |
|-------------|---|---|---|
| Chapter 2 | List of 435 booter Websites automatically collected using Google search engine and classified using several metrics | http://booterblacklist.com | https://github.com/jjsantanna/Booter-black-List |
| Chapter 2.5 | More than 130k DNS records collected by SURFnet and analysed to demonstrate clients accessing booters | — | https://github.com/jjsantanna/booterblacklist_use_cases |
| Chapter 3 | 24 publicly leaked booter databases analysed to demonstrate characteristics of booter clients | | https://github.com/jjsantanna/booter_dbs_analyses |
| Chapter 4 | 15 booter attacks packet-based traces and a list of 248 booter attacks characteristics | https://www.simpleweb.org/wiki/index.php/Traces#Booters_-_An_analysis_of_DDoS-as-a-Service_Attacks and http://ddosdb.org | — |
| Chapter 5 | List of 435 booter Websites enriched with information about their IP address, their Alexa Web Rank, Whois information and Autonomous System | | https://github.com/jjsantanna/booters_ecosystem_analysis |

APPENDIX D

SURFnet and Dutch Prosecutor's Recommendation

In this appendix, we present the advise by the SURFnet responsible (Roland van Rijswijk) for interacting with a Dutch public prosecutor (Danielle Laheij) about the research performed in this thesis.

From: Roland.vanRijswijk@surfnet.nl
Subject: Citing contact with public prosecutor in paper
Date: 18 December 2014 at 17:16
To: Jair Santanna j.j.santanna@utwente.nl
Cc: Anna Sperotto a.sperotto@utwente.nl

Hi Jair,

You can refer to our contact with the public prosecutor in any papers resulting from the research as follows:

"We are aware that research of this nature may touch on, or cross, legal boundaries, but we are convinced that the results from this research will benefit future mitigation methods and thus help combat Booters, both operationally as well as legally. In order to be transparent about our work, we have informed the office of the public prosecutor in the Netherlands about our intention to pursue this research."

This formulation is approved by them.

Cheers,

Roland

--
-- Roland M. van Rijswijk - Deij
-- SURFnet bv
-- w: <http://www.surf.nl/en/about-surf/subsidiaries/surfnet>
-- e: roland.vanrijswijk@surfnet.nl

Figure D.1: Advise by SURFnet regarding the transparency of our research, in accordance with a Dutch public prosecutor.

Bibliography

*Note: all URLs were accessed on the 1st November 2017.

- [1] C. C. Aggarwal, A. Hinneburg, and D. A. Keim. On the surprising behavior of Distance Metrics in high dimensional space. *Database Theory ICDT*, 1973, 2001.
- [2] Akamai. State of the Internet/Security (Q4/2013), 2013. URL <https://www.akamai.com/us/en/multimedia/documents/content/akamai-quarterly-global-attack-report-q4-2013-white-paper.pdf>.
- [3] Akamai. State of the Internet/Security (Q4/2014), 2014. URL https://media.scmagazine.com/documents/104/akamai_q4_ddos_report_25766.pdf.
- [4] Akamai. State of the Internet/Security (Q3/2015), 2015. URL <https://www.stateoftheinternet.com/downloads/pdfs/2015-cloud-security-report-q3.pdf>.
- [5] Akamai. State of the Internet/Security (Q4/2015), 2015. URL <https://www.akamai.com/us/en/multimedia/documents/report/q4-2015-state-of-the-internet-security-report.pdf>.
- [6] Akamai. State of the Internet/Security (Q1/2016), 2016. URL <https://www.akamai.com/us/en/multimedia/documents/state-of-the-internet/akamai-q1-2016-state-of-the-internet-security-report.pdf>.
- [7] Akamai. State of the Internet/Security (Q2/2016), 2016. URL <https://www.akamai.com/us/en/multimedia/documents/state-of-the-internet/akamai-q2-2016-state-of-the-internet-security-report.pdf>.
- [8] Akamai. State of the Internet/Security (Q3/2016), 2016. URL <https://www.akamai.com/us/en/multimedia/documents/state-of-the-internet/q3-2016-state-of-the-internet-security-report.pdf>.
- [9] Akamai. State of the Internet/Security (Q1/2017), 2017. URL <https://www.akamai.com/us/en/multimedia/documents/state-of-the-internet/q1-2017-state-of-the-internet-security-report.pdf>.

- [10] M. Aldwairi and R. Alsalmam. MALURLS: A Lightweight Malicious Website Classification Based on URL Features. *Journal of Emerging Technologies in Web Intelligence*, may 2012.
- [11] Arbor Networks and Arbor Networks. Arbor Networks 9th Annual Worldwide Infrastructure Security Report, 2014. URL <https://pages.arbornetworks.com/rs/arbor/images/WISR2014.pdf>.
- [12] H. Arendt. *Crises of the republic: Lying in politics, civil disobedience on violence, thoughts on politics, and revolution*, volume 219. Houghton Mifflin Harcourt, 1972.
- [13] V. Bajpai, A. W. Berger, P. Eardley, J. Ott, and J. Schönwälder. Global Measurements: Practice and Experience (Report on Dagstuhl Seminar 16012). *Computer Communication Review*, 46(2):32–39, 2016.
- [14] D. Brink. Mill’s Moral and Political Philosophy, 2016. URL <https://plato.stanford.edu/archives/win2016/entries/mill-moral-political/>.
- [15] K. Brownlee. Features of a paradigm case of civil disobedience. *Res Publica*, 10(4):337–351, 2004.
- [16] K. Brownlee. The communicative aspects of civil disobedience and lawful punishment. *Criminal Law and Philosophy*, 1(2):179–192, 2007.
- [17] V. Bukac, V. Stavova, L. Nemec, Z. Riha, and V. Matyas. Service in Denial - Clouds Going with the Winds. In *Network and System Security (NSS)*, 2015.
- [18] T. Chang and C.-C. J. Kuo. Texture analysis and classification with tree-structured wavelet transform. *IEEE Transactions on Image Processing*, 2(4):429–441, 1993.
- [19] A. Chitu. Google Search REST API, 2008. URL <http://googlesystem.blogspot.nl/2008/04/google-search-rest-api.html>.
- [20] J. J. Chromik, J. J. Santanna, A. Sperotto, and A. Pras. Booter websites characterization: Towards a list of threats. In *Brazilian Symposium on Computer Networks and Distributed Systems (SBRC)*, 2015.
- [21] G. Coleman. *Hacker, Hoaxer, Whistleblower, Spy: the Story of Anonymous*. Verso Books, London and New York, 2014.
- [22] J. Damas, M. Graff, and P. Vixie. Extension Mechanisms for DNS (EDNS(0)). RFC 6891, 2013. URL <https://tools.ietf.org/html/rfc6891>.
- [23] D. Dittrich. The DoS Project’s ‘trinoo’distributed denial of service attack tool, 1999. URL <https://staff.washington.edu/dittrich/misc/trinoo.analysis.txt>.

- [24] DJNZ and The Action Tool Development Group of the Electrohippies Collective and DJNZ and The Action Tool Development Group of the Electrohippies Collective. Client-side Distributed Denial-of-Service: Valid campaign tactic or terrorist act?, 2000.
- [25] D. Douglas, J. J. Santanna, R. de O. Schmidt, L. Z. Granville, and A. Pras. Booters: Can Anything Justify Distributed Denial-of-Service (DDoS) Attacks for Hire? *Journal of Information, Communication and Ethics in Society (JICES)*, 15(1), 2017.
- [26] Europol. Cyber Crime vs Cyber Security: what will you choose?, 2016. URL <https://www.europol.europa.eu/activities-services/public-awareness-and-prevention-guides/cyber-crime-vs-cyber-security-what-will-you-choose>.
- [27] P. Ferguson and D. Senie. BCP38: Network Ingress Filtering, 2000. URL <https://tools.ietf.org/pdf/bcp38.pdf>.
- [28] S. Gallagher. Details on the denial of service attack that targeted Ars Technica, 2013. URL <http://arstechnica.com/security/2013/03/details-on-the-denial-of-service-attack-that-targeted-ars-technica/>.
- [29] S. Garera, N. Provos, M. Chew, and A. D. Rubin. A framework for detection and measurement of phising attacks. In *ACM Workshop on Recurring Malcode*, 2007.
- [30] Google Chrome. Manage warnings about unsafe sites, 2017. URL <https://support.google.com/chrome/answer/99020>.
- [31] M. Hammami, Y. Chahir, and L. Chen. WebGuard: A web filtering engine combining Textual, Structural, and Visual content-based analysis. *IEEE Transactions on Knowledge & Data Engineering*, 18, 2006.
- [32] S. Hilton. Dyn Analysis Summary Of Friday October 21 Attack, 2016. URL <https://dyn.com/blog/dyn-analysis-summary-of-friday-october-21-attack/>.
- [33] A. Hinneburg, C. C. Aggarwal, and D. A. Keim. What is the nearest neighbor in high dimensional spaces? In *International Conference on Very Large Data Bases (VLDB)*, 2000.
- [34] P. Howarth and S. Rüger. Fractional Distance Measures for Content-Based Image Retrieval. In *European Conference on IR Research (ECIR)*, 2005.
- [35] Internet Live Stats. Internet Users, 2016. URL <http://www.internetlivestats.com/internet-users>.
- [36] T. Joachims. Text categorization with support vector machines: learning with many relevant features. In *10th ECML*, 1998.

- [37] T. Jordan. *Information Politics: Liberation and Exploitation in the Digital Society*. Pluto Press, London, 2015.
- [38] S. Kaplantzis and N. Mani. A study on classification techniques for network intrusion detection. In *Conference on Networks and Communication Systems (NCS)*, 2006.
- [39] M. Karami and D. McCoy. Understanding the Emerging Threat of DDoS-as-a-Service. In *6th USENIX Workshop on Large-Scale Exploits and Emergent Threats*, 2013.
- [40] M. Karami and D. McCoy. Rent to Pwn: Analyzing Commodity Booter DDoS Services. ; *login:: the magazine of USENIX & SAGE*, 38(6):20—23, 2013.
- [41] M. Karami, P. Youngsam, and D. McCoy. Stress Testing the Booters: Understanding and Undermining the Business of DDoS Services. In *International Conference on World Wide Web (WWW)*, 2016.
- [42] Kaspersky Lab. Denial of Service: How Businesses Evaluate the Threat of DDoS Attacks, 2015. URL https://kasperskycontenthub.com/presscenter/files/2015/09/IT_Risks_Survey_Report_Threat_of_DDoS_Attacks.pdf.
- [43] F. Kausar, B. Al-Otaibi, A. Al-Qadi, and N. Al-Dossari. Hybrid client side phishing websites detection approach. *International Journal of Advanced Computer Science and Applications (IJACSA)*, 5, 2014.
- [44] M. Kerkers, J. J. Santanna, and A. Sperotto. Characterisation of the Kelihos.B Botnet. In *International Conference on Autonomous Infrastructure, Management and Security (AIMS)*, 2014.
- [45] M. Klang. Civil disobedience online. *Journal of Information, Communication and Ethics in Society*, 2(2):75–83, 2004.
- [46] D. Kostadinov. Legality of DDoS: Criminal Deed vs. Act of Civil Disobedience, 2013. URL <http://resources.infosecinstitute.com/legality-ddos-criminal-deed-vs-act-civil-disobedience/>.
- [47] B. Krebs. Ragebooter: Legit DDoS Service, or Fed Backdoor, 2013.
- [48] B. Krebs. The Obscurest Epoch is Today, 2013. URL <http://krebsonsecurity.com/2013/03/the-obscurest-epoch-is-today>.
- [49] B. Krebs. Alleged vDOS Proprietors Arrested in Israel, 2016. URL <https://krebsonsecurity.com/2016/09/alleged-vdos-proprietors-arrested-in-israel/>.
- [50] B. Krebs. Booters (list of posts), 2016. URL <http://krebsonsecurity.com/?s=booter>.

- [51] B. Krebs. Hackforums Shutters Booter Service Bazaar, 2016. URL <https://krebsonsecurity.com/2016/10/hackforums-shutters-booter-service-bazaar/>.
- [52] B. Krebs. Source Code for IoT Botnet ‘Mirai’ Released, 2016. URL <https://krebsonsecurity.com/2016/10/source-code-for-iot-botnet-mirai-released/>.
- [53] B. Krebs and L. James. Black Hat USA 2013 - Spy-Jacking the Booters, 2013. URL https://www.youtube.com/watch?v=wW5vJyI_HcU.
- [54] H.-P. Kriegel and M. Schubert. Classification of Websites as Sets of Feature Vectors. In *International Conference Databases and Applications (IASTED)*, 2004.
- [55] P. Krumins. Python Library for Google Search, 2009. URL <http://www.catonmat.net/blog/python-library-for-google-search/>.
- [56] C. Lindemann and L. Littig. Coarse-grained Classification of web sites by their structural properties. In *International Workshop on Web Information and Data Management*. ACM, 2006.
- [57] C. Lindemann and L. Littig. Classifying web sites. In *International Conference on World Wide Web*. ACM, 2007.
- [58] H. F. Lipson. Tracking and Tracing Cyber-Attacks: Technical Challenges and Global Policy Issues, 2002.
- [59] C. Ludl, S. McAllister, E. Kirda, and C. Kruegel. On the effectiveness of techniques to detect phishing sites. In *Detection of Intrusions and Malware, and Vulnerability Assessment (DIMVA)*, 2007.
- [60] D. Maan, J. J. Santanna, A. Sperotto, and P.-T. de Boer. Towards Validation of the Internet Census 2012. In *EUNICE/IFIP International Workshop*, 2014.
- [61] L. M. Manevitz and M. Yousef. One-class svms for document classification. *The Journal of Machine Learning Research*, pages 139–154, 2002.
- [62] K. Marc, T. Hupperich, C. Rossow, and T. Holz. Exit from Hell? Reducing the Impact of Amp DDoS Attacks. *USENIX Security*, 2014.
- [63] E. Morozov. *The Net Delusion: How Not to Liberate the World*. Penguin Books, London, 2011.
- [64] National Crime Agency. Operation Vulcanalia targets users of netspoof website attack tool, 2015. URL <http://www.nationalcrimeagency.gov.uk/news/974-operation-vulcanalia-targets-users-of-netspoof-website-attack-tool>.

- [65] National Crime Agency. Operation Vivarium targets users of Lizard Squad's website attack tool, 2015. URL <http://www.nationalcrimeagency.gov.uk/news/691-operation-vivarium-targets-users-of-lizard-squad-s-website-attack-tool>.
- [66] Neustar. Worldwide DDoS Attacks & Cyber Insights Research Report, 2017. URL https://ns-cdn.neustar.biz/creative_services/biz/neustar/www/resources/whitepapers/it-security/ddos/neustar-2017-worldwide-ddos-attacks-cyber-insights-research-report.pdf.
- [67] Politie.nl. Ga naar content Politie komt 15 personen op het spoor na gebruik booter- of cryperservice, 2016. URL <https://www.politie.nl/nieuws/2016/december/12/11-politie-komt-15-personen-op-het-spoor-na-gebruik-booter--of-cryperservice.html>.
- [68] J. Postel. RFC864: Character Generator Protocol. RFC 689, 1983. URL <https://tools.ietf.org/html/rfc864>.
- [69] A. Pras, A. Sperotto, G. Moura, I. Drago, R. Barbosa, R. Sadre, R. Schmidt, and R. Hofstede. Attacks by “Anonymous” WikiLeaks Proponents not Anonymous. Technical report, University of Twente, 2010.
- [70] A. Pras, J. J. Santanna, J. Steinberger, and A. Sperotto. DDoS 3.0 - How Terrorists Bring Down the Internet. In *International German Informatics Society (GI) and Technology-Enabled Trading Solutions (ITG) Conference*, 2016.
- [71] Prolexic. Prolexic Threat Advisory - Threat: DDoS Booter Shell Scripts, 2012. URL http://ww1.prweb.com/prfiles/2012/04/24/9438887/ProlexicThreatAdvisoryDDoS_BooterScripts_041912.pdf.
- [72] Prolexic. Multiplayer video gaming attacks, 2013. URL <http://www.prolexic.com/knowledge-center-white-paper-gaming-reflection-attacks-drDos-ddos-infographic.html>.
- [73] Prolexic. Quarterly global DDoS attack report Q3, 2013. URL <http://www.prolexic.com/knowledge-center-ddos-attack-report-2013-q3.html>.
- [74] Prolexic. An Analysis of DrDoS and DDoS Attacks Involving the Multiplayer Video Gaming Community, sep 2013. URL <http://www.stateoftheinternet.com/resources-web-security-white-paper-2013-drDos-multiplayer-video-gaming-attacks.html>.
- [75] Radware. DDoS survival handbook, 2013. URL https://security.radware.com/uploadedFiles/Resources_and_Content/DDoS_Handbook/DDoS_Handbook.pdf.
- [76] J. Rawls. *A Theory of Justice*. Harvard University Press, Cambridge, Massachusetts, 1971.

- [77] Rever Security. Analysis of the Booter.tw Database, 2013. URL <http://www.reversesecurity.com/2013/03/analysis-of-bootertw.html>.
- [78] O. Ruffin. Hacktivismo, 2000. URL <http://w3.cultdeadcow.com/cms/2000/07/hacktivismo.html>.
- [79] J. J. Santanna and A. Sperotto. Characterizing and mitigating the DDoS-as-a-service phenomenon. In *IFIP International Conference on Autonomous Infrastructure, Management and Security (AIMS)*, 2014.
- [80] J. J. Santanna, R. Durban, A. Sperotto, and A. Pras. Inside Booters: An Analysis on Operational Databases. In *IFIP/IEEE International Symposium on Integrated Network Management (IM)*, 2015.
- [81] J. J. Santanna, R. van Rijswijk-Deij, A. Sperotto, R. Hofstede, M. Wierbosch, L. Zambenedetti Granville, and A. Pras. Booters-An analysis of DDoS-as-a-Service Attacks. In *IFIP/IEEE Symposium on Integrated Network and Service Management (IM)*, 2015.
- [82] J. J. Santanna, R. de O. Schmidt, D. Tuncer, J. de Vries, L. Granville, and A. Pras. Booter Blacklist: Unveiling DDoS-for-hire Websites. In *International Conference on Network and Service Management (CNSM)*, 2016.
- [83] J. J. Santanna, R. de O. Schmidt, D. Tuncer, J. de Vries, L. Zambenedetti Granville, and A. Pras. Booter List Generation: The Basis for Investigating DDoS-for-hire Websites. *International Journal on Network Management (IJNM)*, 2017.
- [84] J. J. Santanna, R. de O. Schmidt, D. Tuncer, A. Sperotto, L. Z. Granville, and A. Pras. Quite Dogs Can Bite: What Booters We Should Go After? and Which Are Our Mitigation Options? *IEEE Communications Magazine*, 55(7):50–56, 2017.
- [85] M. Sauter. *The Coming Swarm: DDoS Actions, Hacktivism, and Civil Disobedience on the Internet*. Bloomsbury, New York & London, 2014.
- [86] D. Schwarz. Digging Through an “Administrative Network Stressor” Provider’s Database, 2013. URL <http://www.arbornetworks.com/asert/2013/03/digging-through-an-administrative-network-stressor-providers-database/>.
- [87] Statistica.com. Internet of Things (IoT): number of connected devices worldwide, 2016. URL <https://www.statista.com/statistics/471264/iot-number-of-connected-devices-worldwide/>.
- [88] J. Steinberger, J. J. Santanna, E. Spatharas, H. Amler, N. Breuer, B. Kuhnert, U. Piontek, A. Sperotto, H. Baier, and A. Pras. “Ludo”: Kids Playing Distributed Denial of Service. In *TERENA Networking Conference (TNC)*, 2016.

- [89] C. Sun, N. Rampalli, F. Yang, and A. Doan. Chimera: large-scale classification using Machine Learning, Rules and Crowdsourcing. *VLDB Endowment*, 7(13):1529–1540, 2014.
- [90] P. Tassi. Lizard Squad Hacker Who Shut Down PSN, Xbox Live, And An Airplane Will Face No Jail Time, 2015. URL <http://www.forbes.com/sites/insertcoin/2015/07/09/lizard-squad-hacker-who-shut-down-psn-xbox-live-and-an-airplane-will-face-no-jail-time/>.
- [91] Tech Help Canada Staff. Google, Bing, Yahoo Comparison and Review, 2015. URL <http://www.techhelp.ca/blog/reviews/google-bing-yahoo-compare/>.
- [92] The Guardian. North Korea launched cyber attacks, says south, 2009. URL <https://www.theguardian.com/world/2009/jul/11/south-korea-blames-north-korea-cyber-attacks>.
- [93] The Guardian. Teenage hacker jailed for masterminding attacks on Sony and Microsoft, 2017. URL <https://www.theguardian.com/technology/2017/apr/25/teenage-hacker-adam-mudd-jailed-masterminding-attacks-sony-microsoft>.
- [94] W. Turton. Lizard Squad’s Xbox Live, PSN attacks were a ’marketing scheme’ for new DDoS service, 2014. URL <http://www.dailydot.com/crime/lizard-squad-lizard-stresser-ddos-service-psn-xbox-live-sony-microsoft/>.
- [95] R. van Rijswijk-Deij, M. Jonker, A. Sperotto, and A. Pras. A High-Performance, Scalable Infrastructure for Large-Scale Active DNS Measurements. *IEEE Journal on Selected Areas in Communications (JSAC)*, 34(7), 2016.
- [96] Verisign. Distributed Denial of Service Trends Report (Q2 2016), 2016.
- [97] T. Vijayj. Analysis of DDoS Service Database used to attack Brian Kreb’s Website, 2013. URL <http://vijayjt.blogspot.nl/2013/04/analysis-of-ddos-service-database-used.html>.
- [98] T. Vissers, T. Van Goethem, W. Joosen, and N. Nikiforakis. Maneuvering around clouds: Bypassing cloud-based security providers. In *Conference on Computer and Communications Security (CCS)*, 2015.
- [99] W. Vries, J. J. Santanna, A. Sperotto, and A. Pras. How Asymmetric Is the Internet? A Study to Support the Use of Traceroute (Best Paper Award). In *International Conference on Autonomous Infrastructure, Management and Security (AIMS)*, 2015.
- [100] S. Wain. Googolplex Page, 2004. URL <http://googolplex.sourceforge.net/>.
- [101] K. Q. Weinberger, J. Blitzer, and L. K. Saul. Distance metric learning for large margin nearest neighbor classification. In *Advances in Neural Information Processing Systems*, pages 1473–1480, 2005.

- [102] Wikipedia. Timeline of events associated with Anonymous, 2016. URL https://en.wikipedia.org/wiki/Timeline_of_events_associated_with_Anonymous.
- [103] R. Windrem. Timeline: Ten Years of Russian Cyber Attacks on Other Nations, 2016. URL <https://www.nbcnews.com/news/us-news/timeline-ten-years-russian-cyber-attacks-other-nations-n697111>.
- [104] J.-b. Zhang, Z.-m. Xu, K.-l. Xiu, and Q.-s. Pan. A Web Site Classification Approach Based On Its Topological Structure. *International Journal on Asian Language Processing*, 20, 2010.

About the author



by Mattijs Jonker

I was born in Belém, Pará, Brazil, on October 11th, 1987. I received my Bachelor of Science (B.Sc.) degree in Computer Engineer in 2010 from the Federal University of Pará, Brazil. I was awarded as the third best student of the Computer Engineer in 2010. I got my enthusiasm on talking to people by teaching the Catholic church rules to dozens of kids, teenagers, and adults, at a parish (CAJU) in my home town.

I received my Master of Science (M.Sc.) degree in Computer Science in 2012 from the Federal University of Rio Grande do Sul, Brazil. The solution developed during my master thesis (parts of the MEICAN project) is still in use by the Brazilian NREN (RNP). In 2012, I had a great pleasure to be a lecture at the University of Santa Cruz do Sul teaching hundreds of students.

In the period between 2013 and 2017, I was a Ph.D. candidate at the University of Twente, the Netherlands. A number of solutions proposed in my Ph.D. thesis were deployed by network operators worldwide and some methodologies are used by the Dutch High Tech Crime Unit. During those years, I also acted as technical advisor to the Dutch National Cyber Security Center and to the Japanese Ministry of Internal Affairs and Communications. A couple months before getting my Doctor degree, I got my first project proposal accepted (by SIDNfonds). Then, still in 2017, I became assistant professor at University of Twente. My current goal is to build the biggest public database with DDoS attack intelligence (<http://ddosdb.org>) and determine the future of detection and mitigation of DDoS attacks. For more information about me, please access <http://jairsantanna.com>. Following is a list of papers I published during the time I was a Ph.D. candidate, sorted in reverse chronological order:

1. J. J. Santanna, R. de O. Schmidt, D. Tuncer, A. Sperotto, L. Z. Granville, and A. Pras. Quite Dogs Can Bite: What Booters We Should Go After? and Which Are Our Mitigation Options? *IEEE Communications Magazine*, 55(7):50–56, 2017
2. J. J. Santanna, R. de O. Schmidt, D. Tuncer, J. de Vries, L. Zambenedetti Granville, and A. Pras. Booter List Generation: The Basis for Investigating DDoS-for-hire Websites. *International Journal on Network Management (IJNM)*, 2017
3. D. Douglas, J. J. Santanna, R. de O. Schmidt, L. Z. Granville, and A. Pras. Booters: Can Anything Justify Distributed Denial-of-Service (DDoS) Attacks for Hire? *Journal of Information, Communication and Ethics in Society (JICES)*, 15(1), 2017
4. J. J. Santanna, R. de O. Schmidt, D. Tuncer, J. de Vries, L. Granville, and A. Pras. Booter Blacklist: Unveiling DDoS-for-hire Websites. In *International Conference on Network and Service Management (CNSM)*, 2016
5. J. Steinberger, J. J. Santanna, E. Spatharas, H. Ammer, N. Breuer, B. Kuhnert, U. Piontek, A. Sperotto, H. Baier, and A. Pras. “Ludo”: Kids Playing Distributed Denial of Service. In *TERENA Networking Conference (TNC)*, 2016
6. A. Pras, J. J. Santanna, J. Steinberger, and A. Sperotto. DDoS 3.0 - How Terrorists Bring Down the Internet. In *International German Informatics Society (GI) and Technology-Enabled Trading Solutions (ITG) Conference*, 2016
7. J. J. Chromik, J. J. Santanna, A. Sperotto, and A. Pras. Booter websites characterization: Towards a list of threats. In *Brazilian Symposium on Computer Networks and Distributed Systems (SBRC)*, 2015
8. J. J. Santanna, R. van Rijswijk-Deij, A. Sperotto, R. Hofstede, M. Wierbosch, L. Zambenedetti Granville, and A. Pras. Booters-An analysis of DDoS-as-a-Service Attacks. In *IFIP/IEEE Symposium on Integrated Network and Service Management (IM)*, 2015
9. J. J. Santanna, R. Durban, A. Sperotto, and A. Pras. Inside Booters: An Analysis on Operational Databases. In *IFIP/IEEE International Symposium on Integrated Network Management (IM)*, 2015
10. W. Vries, J. J. Santanna, A. Sperotto, and A. Pras. How Asymmetric Is the Internet? A Study to Support the Use of Traceroute (Best Paper Award). In *International Conference on Autonomous Infrastructure, Management and Security (AIMS)*, 2015
11. M. Kerkers, J. J. Santanna, and A. Sperotto. Characterisation of the Kelihos.B Botnet. In *International Conference on Autonomous Infrastructure, Management and Security (AIMS)*, 2014
12. D. Maan, J. J. Santanna, A. Sperotto, and P.-T. de Boer. Towards Validation of the Internet Census 2012. In *EUNICE/IFIP International Workshop*, 2014
13. J. J. Santanna and A. Sperotto. Characterizing and mitigating the DDoS-as-a-service phenomenon. In *IFIP International Conference on Autonomous Infrastructure, Management and Security (AIMS)*, 2014

