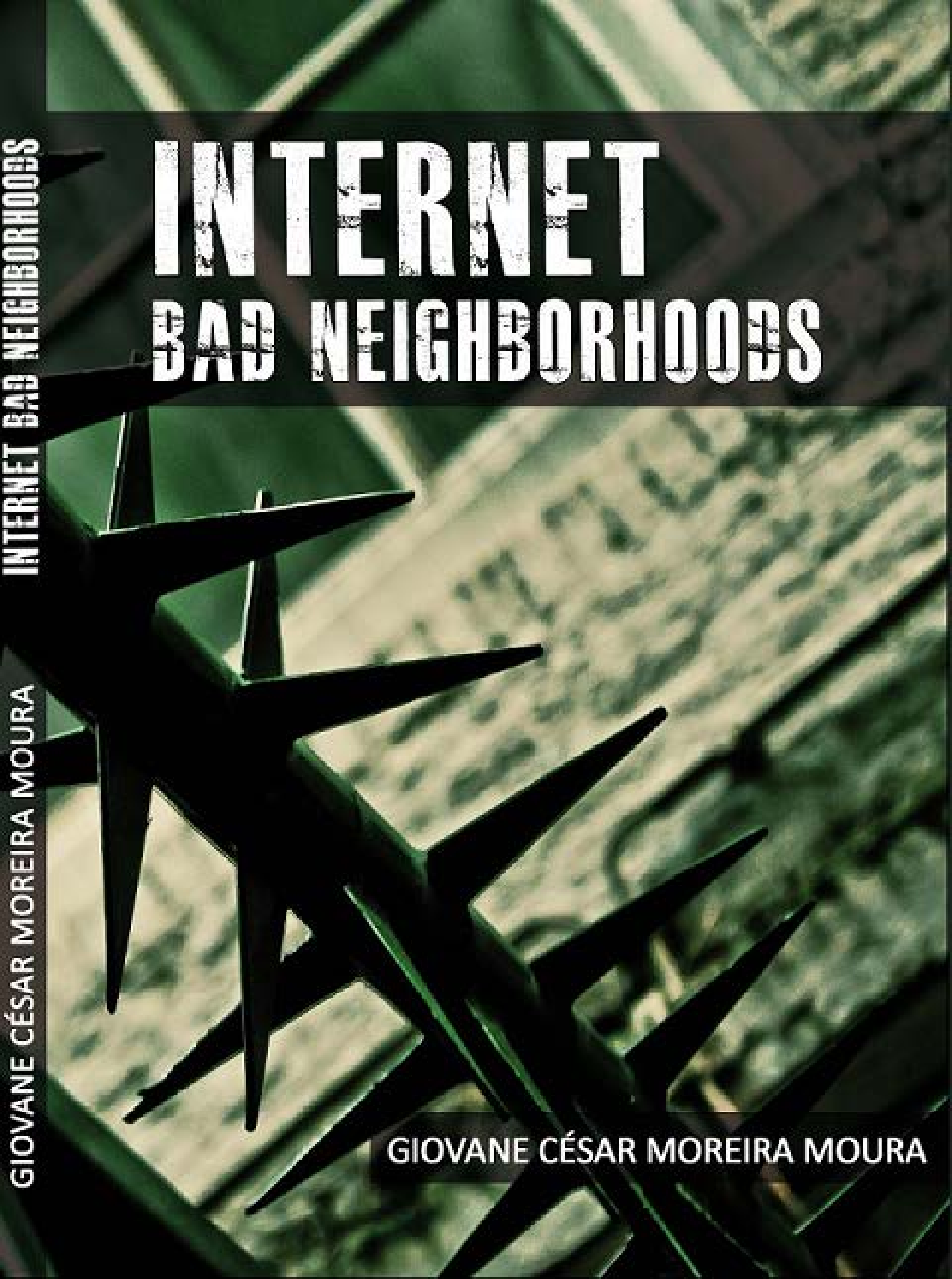


INTERNET BAD NEIGHBORHOODS

GIOVANE CÉSAR MOREIRA MOURA

INTERNET BAD NEIGHBORHOODS
GIOVANE CÉSAR MOREIRA MOURA

Internet Bad Neighborhoods

Giovane César Moreira Moura

Graduation committee:

Chairman:	Prof. Dr. ir. A.J. Mouthaan
Promoter:	Prof. Dr. ir. Boudewijn R. Haverkort
Assistant promoter:	Dr. ir. Aiko Pras

Members:

Prof. Dr. Gabi Dreo Rodosek	Universität der Bundeswehr München, Germany
Prof. Dr. Luciano P. Gaspary	Federal University of Rio Grande do Sul, Brazil
Prof. Dr. Frank Kargl	University of Twente, The Netherlands and University of Ulm, Germany
Prof. Dr. Hans van den Berg	University of Twente, The Netherlands
Dr. Ramin Sadre	Aalborg University, Denmark
Dr. Johnny H. Søraker	University of Twente, The Netherlands

CTIT Ph.D.-thesis Series No. 12-237
Centre for Telematics and Information Technology
P.O. Box 217, 7500 AE
Enschede, The Netherlands

ISBN 978-90-365-3460-4
ISSN 1381-3617
DOI: 10.3990/1.9789036534604
http://dx.doi.org/10.3990/1.9789036534604

Publisher: Ipskamp Drukkers B.V.
Cover design: Rodrigo Mantovaneli Pessoa
Cover photo: Rodrigo Mantovaneli Pessoa – in Amsterdam, The Netherlands
About the Author section photo: Peter Asaro

INTERNET BAD NEIGHBORHOODS

DISSERTATION

to obtain
the degree of Doctor at the University of Twente,
on the authority of the Rector Magnificus,
Prof. Dr. H. Brinksma,
on account of the decision of the Graduation Committee,
to be publicly defended
on Friday, March 1st, 2013 at 14:45

by

Giovane César Moreira Moura

born on September 1st, 1981
in Goiânia, Goiás, Brazil

This dissertation has been approved by:
Prof. dr. ir. Boudewijn R. Haverkort (Promotor)
Dr. ir. Aiko Pras (Assistant-promotor)

Thanks to technology, what almost anybody can do has been multiplied a thousandfold, and our moral understanding about what we ought to do hasn't kept pace. ... You can lay minefields, smuggle nuclear weapons in suitcases, make nerve gas, and drop "smart bombs" with pinpoint accuracy. Also, you can arrange to have a hundred dollars a month automatically sent from your bank account to provide education for ten girls in an Islamic country who otherwise would not learn to read and write .... You can use the Internet to organize citizen monitoring of environmental hazards, or to check the honesty and performance of government officials – or to spy on your neighbors. Now, what ought we to do?

— Daniel Dennett, 2006
In: Breaking the Spell.

Gratus animus est una virtus non solum maxima, sed etiam mater virtutum omnium reliquarum.
(A thankful heart is not only the greatest virtue, but the parent of all the other virtues).

Marcus Tullius Cicero
In: Oratio Pro Cnæo Planci, XXXIII

Acknowledgments

My advisor, Aiko Pras, used to say that pursuing a PhD is like being a F1 race driver: you are only going to make it if you are fully committed, 24/7. I am very fortunate to have had Aiko as advisor during my PhD at the *scuderia* DACS (Design and Analysis of Communication Systems, our research group). Aiko's sharp reasoning and his amazing ability to frame problems in otherwise unthought ways are only matched by his continuous joy in doing research: he always has something new or have just met someone who could contribute to improve the research. By challenging and motivating me at the same time, Aiko has not "shown me the way"; he has, in fact, taught me *how to find my own way*.

Ramin Sadre has been also another amazing mentor for generations of doctoral candidates at DACS, and also my mentor. His keen eye won't let any inconsistences go unnoticed. His patience and humility, coupled with his broad knowledge in several fields, have helped me to open the right doors and close the wrong ones during the course of the research. Discussions with Ramin have contributed significantly to strengthen the conceptual aspects of this dissertation. Ramin has been a truly advisor during these fours years of PhD, and has read painstakingly many times drafts of papers and chapters of this dissertation.

I also owe a great deal of gratitude to my promoter, Boudewijn Haverkort, who has always been enthusiastic about my research and had carefully read this dissertation. I am also very lucky to have met and worked with so many amazing people at the at DACS group, where it has been always fun to work together. Anna Sperotto has been involved later in my research and discussions with her have also contributed to improve the theoretical aspects of this dissertation. I have had great officemates, which have helped me so many times: Rafael Barbosa, Idilio Drago, and Anna Kolesnichenko.

I could not have concluded this thesis without the data that has been shared with us by so many different people and organizations. In special, thanks to Casper Joost Eyckelhof and Matthijs van Polen from Quarantainenet, and to Frederico Costa and Liliana Solha from the Security Incident Response Team of the Brazilian Research Network (CAIS/RNP). Also, many thanks to Rogier Spoor at SurfNet (Dutch Research Network), and the work of anonymous maintainers of publicly available Internet blacklists and data sets (PSBL, CBL, DShield, UCE-protect, SBL, Provider A). Special thanks to Marc Berenschot, from the University of Twente, for the great help whenever we needed. Also, thanks Wouter de Vries and Ward van Wanrooij, and to o Gert Vliek at the Dutch National Cyber Security Centre (NCSC).

I have also been very lucky to work in the FP6 network of excellence EMANICS project, which have helped me to collaborate to other project partners. Special thanks to Jérôme François and Olivier Festor for the collaboration with INRIA/France. In addition, many thanks to Burkhard Stiller to have welcomed me as a research guest at his Communication Systems Group (CSG) at the University of Zurich in the beginning of my PhD, and the amazing people I have met there (Fabio Hecht – and Anna Paula –, Guilherme Machado, David Hausheer, Martin Waldburger, Cristian Morariu). Also, and all the feedback obtained from many people of the EMANICS community, in special George Pavlou and Marinos Charalambides. And also, my special thanks to my former professors Lisandro Granville and Luciano Gaspary from the Federal University of Rio Grande do Sul, where I did my masters, whom have helped me to enter in this network management research community.

This dissertation could not be finished without the amazing support system that I have found in Enschede. As Johnny Søraker once said in one of his birthday parties, "Enschede is not about the town; it's about the people you have around". I have been extremely fortunate in finding so many fascinating people in the same situation as me: from another country/cities, far from their families, pursing a PhD or somehow involved with the University. In special, my deepest gratitude to Aimee van Wynsberghe, a true sister that I never had, for all the support and for being there for those many years. Thanks for everything you've done and for all the amazing cracking and draining sessions, I have learned so much from you! I will be back many times to visit you and Scott, and the baby girl on the way :-). Also, many thanks to Luiz Olavo and Luciana Bonino (and little Anna Martha), for helping me out so much during all these years – even teaching me recipes that would not go wrong! And Flávia "Flavinha"Souza and Arun Vydhyanathan – what a great couple and amazing friends too: thanks a lot for being there, for all the amazing energy and happiness, and for making Enschede feel like home – actually like Rio, so I became more "Carioca" ;-) – and for Molly's turning into our second home!

Even though my PhD is in Computer Science, very soon I was "adopted" by the Department of Philosophy crew, after meeting Aimee. What a wonderful group! Thanks a lot Aimee van Wynsberghe, Scott Robbins, Lucie Dalibert, Tjerk

Timan, Federica Lucivero, Steven Dorrestijn, Johnny and Linn Søraker (thanks so much for keeping the group rolling, always will remember the amazing Rock Band parties – and the Rocks!), Lise Bitsch, Josine Verhagen, Irina Avetisyan, Marianna Avetisyan. And of course, my "brother from another mother", Desmond "Des" Treacy and Clare Shelley-Egan. I have also met incredible people at the Tissue Regeneration group – in special Aliz Kunstar (thanks a lot for everything and good luck in Michigan!), Hugo Fernandes, Ana Barradas, and Björn Harink. Oh yeah, and thanks a lot Nekane Larburu and Alicia Martinez.

And the University's PhD Network (P-NUT)! What an amazing time there – meeting great and fun people and developing amazing skills. Thanks so much the P-NUT crew, old and new: Aimee van Wynsberghe, Josine Verhagen, Anika Embrechts, Shashank Shekhar, Sérgio Pacheco, Björn Harink, Nicole Georgi, Silja Eckartz, Ioana "Nana" Ilie, Adithya "Adi" Sridhar, Bijoy Bera, Febriyani Damanik, David Barata, Joana Romão, Juan Amiguet (keep on with the Juanism, man!), Harmen Mulder, Juan Carlos "JC" R. Casado, Victor de Graaff, and Rense Nieuwenhuis. Thanks so much for the awesome time at P-NUT! Moreover, all the colleagues and friends at DACS and the Software Engineering group: Pieter-Tjerk de Boer, Hans van den Berg, Geert Heijenk, Georgios Karagiannis, Anne Remke, Martijn van Eenennaam, Wouter Klein Wolterin, Stephan "Steve" Roolvink, Daniel Reijsbergen, Rick Hofstede, Karol A. Rosen, Marijn Jongerden, Marc Berenschot, Eduardo Manuel, Laura Daniele, Luiz Olavo Bonino, and Rafael Barbosa (and Aleksandra Kaspera). And of course many thanks to the great friends Ricardo and Kasia Neisse. And the "Buena Vida Social Club": Valentina Spanu – thanks so much, Vale, for being there for me and for always bringing on the fun, and for amazing summer holidays in Sardinia – Arun and Flavinha, and Arturo Balderas. Thanks very much guys, what a great time!

I have left Brazil for pursuing the PhD, but my friends in Brazil have always been there for me too. In special, I would like to thank Jéferson "Jeff" Campos Nobre (thanks for helping me to become a "DIY Psychologist"), Rodrigo Mantovaneli Pessoa (thanks for the cover and for the Rock in Rio as well!), and Fernando Guimarães (valeu Parsa!). Also, thanks Lisandro Granville so much for all the professional advice whenever we would meet in conferences. Many thanks to Prof. Jürgen Rochol, for always kindly emphasizing the importance of bearing in mind the direct usefulness and application of our research (the "reality plane"). In addition, many thanks to my great friends Jean Veríssimo, Débora Veríssimo, Carla Schwengber, Alisson Rauber, Anderson Rauber, Fabio Rauber, Wellington and Pamela Moreira, and Reverton Moreira. Many thanks to Tiago Cabral, Valdblan Freitas, and Thiago Lopez, long time high-school friends. And all the friends from Federal University of Rio Grande do Sul, that I have always met in conferences, in special Jéferson “Jeff” Campos Nobre, Carlos Raniery P. Santos, and Weverton Cordeiro.

Last, but not least, I am eternally grateful to my family: many thanks for always motivating and for supporting me in these four years in this long distance. Thanks mom, dad, Vinícius, and Thales for being a constant source of inspiration, support, and love. I would like to share this moment with you.

Gratus animus est una virtus non solum maxima, sed etiam mater virtutum omnium reliquarum.
(Um coração agradecido não é somente a maior das virtudes, ele é a origem de todas as outras).

Marcus Tullius Cicero
In: Oratio Pro Cnæo Planci, XXXIII

Agradecimentos

Meu orientador, Aiko Pras, costuma dizer que fazer um doutorado é similar a ser um piloto de Fórmula 1: você somente vai conseguir chegar até o fim se estiver totalmente comprometido, 24 por 7. Me considero bastante fortunado em ter tido Aiko como meu supervisor durante meu doutorado na *scuderia* DACS (*Design and Analysis of Communication Systems*, nosso grupo de pesquisa). O raciocínio aguçado de Aiko e sua capacidade de "enxergar" soluções de formas únicas são somente equiparadas por sua contínua empolgação em conduzir pesquisas: ele sempre tinha algo novo ou tinha acabado de conhecer alguém que poderia contribuir com a minha pesquisa. Ao mesmo tempo em que me desafiava e me motivava a continuar a pesquisa, Aiko não "me mostrou o caminho"; ele, na verdade, me ensinou "como encontrar meu próprio caminho".

Ramin Sadre também foi outro fantástico mentor para gerações de doutorandos no DACS, eu incluso. Seu olho clínico não deixa quaisquer inconsistências escaparem despercebidas. Sua paciência e humildade, junto com seu amplo conhecimento em diversos campos da ciência, me ajudaram a abrir as portas certas e fechar as erradas. As discussões com Ramin contribuíram significativamente para reforçar os aspectos conceituais desta tese de doutorado. Ramin foi verdadeiramente um supervisor nestes quatro anos de doutorado, e leu meticulosamente várias vezes vários rascunhos dos meus artigos e dos capítulos desta tese.

Eu também devo minha gratidão ao meu *promoter* Boudewijn Haverkort, que sempre foi bem entusiástico sobre minha pesquisa e leu cuidadosamente esta tese. Também tenho muita sorte de ter conhecido e trabalhado com pessoas tão fantásticas no DACS, onde sempre foi divertido de se trabalhar. Anna Sperotto foi envolvida nos estágios finais da minha pesquisa e discussões com ela também contribuíram para melhorar aspectos teóricos desta tese. Também tiver grandes colegas de sala, que me ajudaram várias vezes: Rafael Barbosa, Idilio Drago e Anna Kolesnichenko.

Eu não poderia ter concluído esta tese sem os dados que foram compartilhados conosco por várias diferentes pessoas e organizações. Em especial, Casper

Joost Eyckelhof e Matthijs van Polen da Quarantainenet B.V., e a Frederico Costa e Liliana Solha do Centro de Atendimento a Incidentes de Segurança da Rede Nacional de Ensino e Pesquisa (CAIS/RNP. Agradecimentos especiais a Rogier Spoor da rede de pesquisa Holandesa (SurfNet), e o trabalho de mantenedores anônimos de várias *blacklists* e *data sets* na Internet (PSBL, CBL, DShield, UCE-protect, SBL, Provider A). Meu muito obrigado a Wouter de Vries e Ward van Wanrooij, e agradecimentos especiais a Gert Vliek do *Dutch National Cyber Security Centre (NCSC)*.

Eu também tive sorte de trabalhar no contexto da rede de excelência FP6 do projeto EMANICS, que me ajudou a colaborar com outros colegas de projeto. Agradecimentos especiais a Jérôme François e Olivier Festor pela colaboração com a INRIA/França. Além disso, muito obrigado a Burkhard Stiller por ter me recebido como pesquisador convidado no seu Communication Systems Group (CSG) na Universidade de Zurique no início do meu doutorado, e a todas as pessoas fantásticas que lá conheci (Fabio Hecht – e Anna Paula –, Guilherme Machado, David Hausheer, Martin Waldburger, e Cristian Morariu). Também, gostaria de agradecer ao feedback da comunidade EMANICS, em especial George Pavlou e Marinos Charalambides. Meu muito obrigado aos meus professores Lisandro Granville e Luciano Gaspary da Universidade Federal do Rio Grande do Sul, onde fiz meu mestrado, que me ajudaram bastante nos primeiros passos na comunidade de gerência de redes.

Essa tese não poderia ter sido concretizada sem a incrível grupo de amigos que eu encontrei em Enschede. Como Johnny Søraker disse uma vez em uma de suas festas de aniversário, “Enschede não é a cidade; são as pessoas que temos ao redor”. Fui extremamente fortunado em encontrar tantas pessoas fascinantes na mesma situação que a minha: de outros países/cidades, longe de suas famílias, em busca do título de doutor ou de outra forma envolvidos com a universidade. Em especial, a minha mais profunda gratidão à Aimee van Wynsberghe, uma verdadeira irmã que eu nunca tive, por todo o apoio e por sempre estar presente em todos esses anos. Muito obrigado por tudo que você fez e pelas sessões “cracking and draining” – eu aprendi tanto contigo. Eu vou voltar muitas vezes para visitar você e o Scott. Também, muito obrigado a Luiz Olavo e Luciana Bonino (e à pequena Anna Martha), por terem me ajudado tantas vezes durante todos esses anos – até me ensinando receitas infalíveis! E também, Flávia “Flavinha” Souza and Arun Vydhyanathan – que casal fantástico e amigos geniais: muito obrigado por estarem presentes, por toda energia contagiante e por fazer Enschede sentir como “home” – na verdade, como o Rio, e eu acabei tornando um pouco Carioca ;-) – e por fazer Molly’s virar nossa segunda casa.

Mesmo que meu doutorado tenha sido em Ciência da Computação, desde o começo eu fui “adotado” pelo pessoal do Departamento de Filosofia. Muito obrigado Aimee van Wynsberghe, Scott Robbins, Lucie Dalibert, Tjerk Timan, Federica Lucivero, Steven Dorrestijn, Johnny and Linn Søraker (muito obrigado por manterem o grupo unido, sempre vou me lembrar das grandes Rock Band parties – e o Rocks!), Lise Bitsch, Josine Verhagen, Irina Avetisyan, Marianna Avetisyan. E claro, meu “irmão de outra mãe”, Desmond “Des” Treacy and Clare Shelley-Egan. Também conheci pessoas fantásticas no grupo Tissue Regeneration – em especial Aliz Kunstar (obrigado por tudo e boa sorte em Michigan!), Hugo Fernandes, Ana Barradas, e Björn Harink. E claro, valeu mesmo Nekane Larburu e Alicia Martinez.

E a Associação de Doutorandos da Universidade (P-NUT)! Que tempo fantástico lá – conhecendo grandes pessoas e divertindo. Muito obrigado a todo a equipe, a velha e a nova: Aimee van Wynsberghe, Josine Verhagen, Anika Embrechts, Shashank Shekhar, Sérgio Pacheco, Björn Harink, Nicole Georgi, Silja Eckartz, Ioana “Nana” Ilie, Adithya “Adi” Sridhar, Bijoy Bera, Febriyani Damanik, David Barata, Joana Romão, Juan Amiguet (mantenha o Juanismo, cara!), Harmen Mulder, Juan Carlos “JC” R. Casado, Victor de Graaff, e Rense Nieuwenhuis. Muito obrigado pelo tempo fantástico no P-NUT! Também, todos os colegas e amigos no DACS e no grupo Software Engineering: Pieter-Tjerk de Boer, Hans van den Berg, Geert Heijenk, Georgios Karagiannis, Anne Remke, Martijn van Eenennaam, Wouter Klein Wolterin, Stephan “Steve” Roolvink, Daniel Reijsbergen, Rick Hofstede, Karol Rosen, Marijn Jongerden, Marc Berenschot, Eduardo Manuel, Laura Daniele, Luiz Olavo Bonino, e Rafael Barbosa (e Aleksandra Kaspera). E também aos grandes amigos Ricardo e Kasia Neisse. E o “Buena Vida Social Club”: Valentina Spanu – muito obrigado, Vale, for sempre estar presente e por sua alegria e gentileza, e pelo verão fantástico na Sardenha – , Arun e Flavinha, e Arturo Balderas. Valeu pessoal, que tempo maneiro!

Eu deixei o Brasil para ir atrás do doutorado, porém meus amigos no Brasil sempre estiveram também presentes nas horas que sempre precisei. Eu especial, gostaria de agradecer Jéferson “Jeff” Campos Nobre (valeu por me ensinar a ser um “Psicólogo DIY”), Rodrigo Mantovaneli Pessoa (valeu pela capa dessa tese e pelo Rock in Rio, brother), e Fernando Guimarães (valeu Parsa!). Também, muito obrigado a Lisandro Granville pelos conselhos profissionais sempre que nos encontrávamos em conferências. Muito obrigado também ao Prof. Jürgen Rochol, por sempre gentilmente lembrar-nos da importância de manter o foco na direta aplicabilidade e utilidade de nossas pesquias (o “plano da relidade”). Muito obrigado também a meus grandes amigos Jean Veríssimo, Débora Veríssimo, Carla Schwengber, Alisson, Fabio, e Anderson Rauber, Wellington e Pamela Moreira e Reverton Moreira. Muito obrigado a Tiago Cabral, Valdblan Freitas, e Thiago Lopez, amigos desde os tempos de colégio. E a todos meus amigos da Universidade Federal do Rio Grande do Sul, com quem eu sempre me encontrava em conferências, em especial Jéferson "Jeff" Campos Nobre, Carlos Raniery P. Santos, e Weverton Cordeiro.

Por último, mas não menos importante. Sou eternamente grato a minha família: obrigado por sempre me motivar e apoiar em todos estes anos, nessa distância. Obrigado mãe, pai, Vinícius e Thales por terem sido uma constante fonte de inspiração, apoio, e amor. Gostaria de compartilhar esse momento com vocês.

Abstract

A significant part of current Internet attacks originates from hosts that are distributed all over the Internet. However, there is evidence that most of these hosts are, in fact, concentrated in certain parts of the Internet. This behavior resembles the crime distribution in the real world: it occurs in most places, but it tends to be concentrated in certain areas. In the real world, high crime areas are usually labeled as “bad neighborhoods”.

The goal of this dissertation is to investigate *Bad Neighborhoods on the Internet*. The idea behind the Internet Bad Neighborhood concept is that the probability of a host in behaving badly increases if its neighboring hosts (*i.e.*, hosts within the same subnetwork) also behave badly. This idea, in turn, can be *exploited to improve current Internet security solutions*, since it provides an indirect approach to predict new sources of attacks (neighboring hosts of malicious ones).

In this context, the main contribution of this dissertation is to present the first *systematic and multifaceted study on the concentration of malicious hosts on the Internet*. We have organized our study according to two main research questions. In the first research question, we have focused on the intrinsic characteristics of the Internet Bad Neighborhoods, whereas in the second research question we have focused on how Bad Neighborhood blacklists can be employed to better protect networks against attacks. The approach employed to answer both questions consists in monitoring and analyzing network data (traces, blacklists, etc.) obtained from various real world production networks.

One of the most important findings of this dissertation is the verification that Internet Bad Neighborhoods are a real phenomenon, which can be observed not only as network prefixes (e.g., /24, in CIDR notation), but also at different and coarser aggregation levels, such as Internet Service Providers (ISPs) and countries. For example, we found that 20 ISPs (out of 42,201 observed in our data sets) concentrated *almost half of all* spamming IP addresses. In addition, a single ISP was found having 62% of its IP addresses involved with spam. This suggests that ISP-based Bad Neighborhood security mechanisms can be

employed when evaluating e-mail from unknown sources.

This dissertation also shows that Bad Neighborhoods are *mostly application-specific* and that they might be located in neighborhoods one would not immediately expect. For example, we found that *phishing* Bad Neighborhoods are mostly located in the United States and other developed nations – since these nations hosts the majority of data centers and cloud computing providers – while spam comes from mostly Southern Asia. This implies that Bad Neighborhood based security tools should be application-tailored.

Another finding of this dissertation is that Internet Bad Neighborhoods are much less stealthy than individual hosts, since they are more likely to strike again a target previously attacked. We found that, in a one-week period, nearly 50% of the individual IP addresses attack *only once* a particular target, while up to 90% of the Bad Neighborhoods attacked more than once. Consequently, this implies that historical data of Bad Neighborhoods attacks can potentially be successfully employed to predict future attacks.

Overall, we have put the Internet Bad Neighborhoods under scrutiny from the point of view of the network administrator. We expect that the findings provided in this dissertation can serve as a guide for the design of new algorithms and solutions to better secure networks.

Resumo

Um parte significante dos ataques atuais na Internet são originários de *hosts* que se encontram distribuídos por toda a Internet. Entretanto, existem evidências de que a maioria desses *hosts* se encontram, de fato, concentrados em certas partes. Este comportamento lembra a distribuição de crimes no mundo real: pode ser encontrado virtualmente em todos os lugares, mas tende a ser concentrado em certas áreas. No mundo real, tais áreas que exibem concentrações de crimes mais altas são comumente chamadas de Más Vizinhanças ("*bad neighborhoods*").

O objetivo dessa tese é investigar as Más Vizinhanças da Internet. A ideia por detrás do conceito de Más Vizinhanças é que a probabilidade um de *host* em executar atividades maliciosas aumenta se seus vizinhos imediatos (*i.e., hosts* na mesma subrede) também se executam atividades maliciosas. Esta ideia, por sua vez, pode ser *explorada para melhorar as atuais soluções para segurança de Internet*, uma vez que assume que *hosts* vizinhos de hosts maliciosos têm mais probabilidade de serem maliciosos e, desta forma, conduzir ataques.

Nesse contexto, a principal contribuição desta tese é apresentar o primeiro *estudo sistemático e multifacetado sobre a concentração de hosts maliciosos na Internet*. Nós dividimos esse em duas *questões principais* (*research questions*). Na primeira, nós nos concentramos nas características intrínsecas das Más Vizinhanças da Internet, enquanto na segunda focamos em como as listas de Más Vizinhanças da Internet podem ser utilizadas para melhor proteger as redes de computadores contra ataques. A abordagem empregada para responder ambas as questões consiste em monitorar e analisar dados de redes (*traces, blacklists,* etc.), obtidos de várias redes de produção.

Uma dos resultados mais importantes obtidos nessa tese é a constatação de que as Más Vizinhanças são um fenômeno real, que podem ser observadas não somente em prefixo de rede (por exemplo, /24 em notação CIDR), mas também em níveis de agregação mais granulares, como provedores de Internet e países. Por exemplo, nós descobrimos que 20 provedores (dos 42.201 observados em nossos dados) concentram *quase metade* de todos os endereços IP envolvidos em

spam. Além disso, um único provedor teve mais de 60% de seus endereços IP associados a *spam*. Esse resultado que sugere mecanismos de segurança baseados em más vizinhanças de provedores podem ser utilizados para avaliar *e-mail* de origens desconhecidas.

Essa tese também mostra que as Más Vizinhanças são quase sempre específicas em relação a aplicação e que elas podem se concentrar em áreas que alguém não imaginaria inicialmente. Por exemplo, nós descobrimos que as maiorias das Más Vizinhanças envolvidas em *phishing* são localizadas nos Estados Unidos e outras nações desenvolvidas – uma vez que estas nações concentram a maioria dos *data centers* e *cloud computing providers* – enquanto *spam* é originado em sua maioria no sudeste asiático. Isso implica que mecanismos de segurança que utilizam más vizinhanças devem ser específicos em relação as aplicações.

Um outro resultado obtido nesta tese é que as Más Vizinhanças são muito mais furtivas que *hosts* individuais, uma vez que elas tendem a atacar os alvos mais de uma vez. Nós descobrimos que, no período de uma semana, quase metade de todos os endereços IP atacaram somente uma vez um alvo em particular, enquanto até 90% das Más Vizinhanças atacaram mais de uma vez. Consequentemente, isso sugere que o passado histórico dos ataques das Más Vizinhanças pode ser utilizado como uma forma de predizer ataques futuros.

No geral, nós colocamos as Más Vizinhanças em escrutínio sobre o ponto de vista do administrador de redes. Nós esperamos que os resultados dessa tese possam servir como guia para desenvolver algoritmos e soluções para melhor proteger as redes de computadores.

Contents

List of Figures

List of Tables

Part I

Introduction

Cyber war skips battlefield. Systems that people rely upon, from banks to air defense radars, are accessible from cyberspace and can be quickly taken over or knocked out without first defeating a country's traditional defense.

Richard Clarke and Robert Knake, 2010
In: Cyber War: The Next Threat to National Security and What to do About it

CHAPTER 1 Introduction

NOVEMBER 22nd, 1977: in the vicinity of San Francisco, California, network data was transmitted to the University of Southern California's Information Sciences Institute in Los Angeles, 400 miles away. To reach the destination, however, the data had to travel more than 100,000 miles, through three different networks: ARPANET, the Packet Radio Network, and the Atlantic Packet Satellite [6]. On this day, the widely regarded first true Internet connection was established, setting a major landmark on the history of the Internet [7].

From the seminal three networks interconnection, the Internet has evolved into one of the *most complex systems ever built in human history* [8, 9]. Currently, it figures as "a large-scale, highly engineered system" [10] that interconnects more than 800 million hosts, which are used by more than two billion people worldwide [11, 12]. The influence of the Internet on society goes way beyond the number of users and hosts. As explained by the sociologist Manuel Castells, "core economics, social, political, and cultural activities throughout the planet are being structured around the Internet" and "exclusion from it (the Internet) is one of the most damaging forms of exclusion in our economy and culture" [13].

The Internet (and the infrastructure around it – servers, routers, etc.) is currently so important for the functioning of our society that it is actually considered part of the *critical infrastructure* of many countries [14]. A myriad of critical systems, such as banking, traffic, and transportation, heavily rely upon the Internet to perform.

Such dependence has made the Internet very attractive for criminal organizations, nation states, and activists as a medium in which crimes, cyberwar, and protests can be carried out. One example is the 2007 Estonia Denial of Service (DDoS) attacks, in which many websites from Estonian organizations, such as the parliament, newspapers, banks, and ministries, were flooded with requests and became overloaded, unable to handle legitimate requests [15].

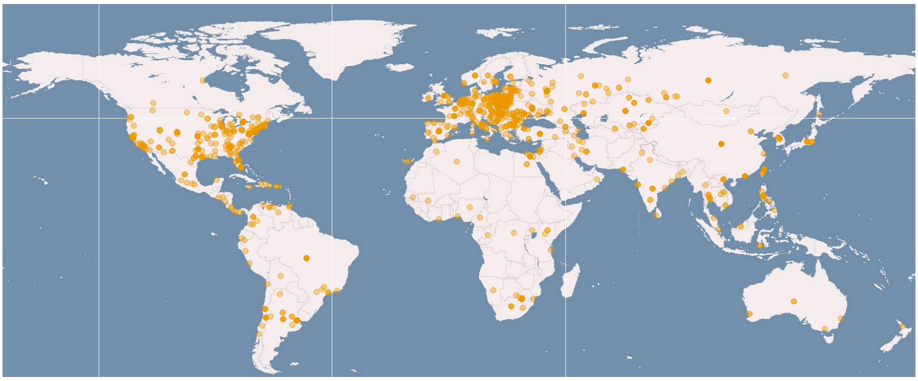


Figure 1.1: Hlux2/Kelihos.B Bots Sample Geo-location

This attack caused a *direct impact in the real world*: Estonians could not use their online banking, access their government online services or even read their online newspapers [14]. Another example of malicious activity on the Internet is spam, a misuse of electronic email. It is estimated that between 84% and 90% of all e-mail messages are spam nowadays [16, 17], and behind it, cyber gangs run lucrative operations by selling pharmaceuticals [18], distributing malicious software (malware), among other illegal activities [19, 4]. As DDoS attacks, spam also impacts the real world: it is estimated that worldwide spam causes losses from $10 billion to $87 billion yearly [20].

Behind these attacks, we typically find a large amount of IP addresses, usually distributed all over the world. Some of these attacks are even carried out by so-called botnets, which are essentially a large number of *distributed* compromised machines (called bots or zombies) under control of a botmaster [21, 22]. The zombies can be seen as "hijacked" computers, located at homes, schools, and businesses, controlled by the botmaster to carry out malicious activities. Figure 1.1 shows the geographical location of a sample of 1,193 computers belonging to the botnet Hlux2/Kelihos.B [23], which we generate by processing a trace file we have obtained from SurfNet [24]. As can be seen, the distribution of bots extends to all populated continents.

Even though the malicious hosts are distributed all over the world, there is evidence that *malicious hosts are, in fact, concentrated in certain networks*. Take as example Figure 1.2, in which we present the distribution of spamming hosts

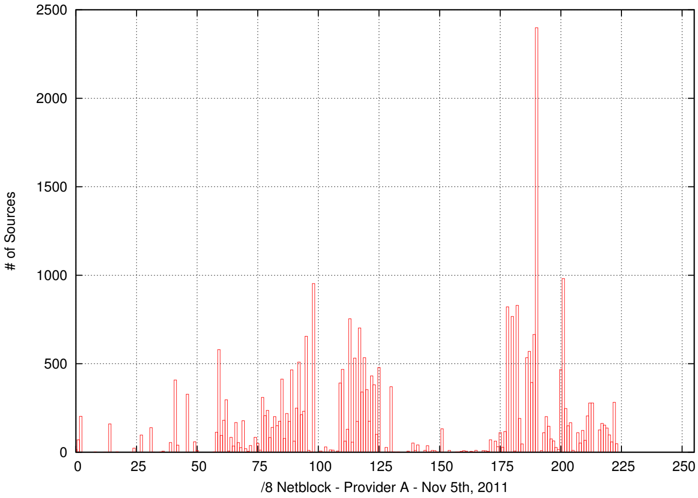


Figure 1.2: Number of spam Sources per /8 netblock

per /8 netblock[1] as seen by a major Dutch hosting provider. As can be seen, there are some /8 netblocks that had much more spammers than others. Other research works have also investigated the concentration of malicious hosts. For example, in 2006 Ramachandran *et al.* [27] have shown that the majority of spam was sent from a small fraction of the IP address space. Collins *et al.* [28], on the other hand, have defined the term "spatial uncleanliness" for clusters of compromised hosts. Chen and Ji [29] have shown that the victims of a particular worm are not evenly distributed on the Internet, and Chen *et al.* have also shown that the distribution of malicious sources is non-uniform across the IP address space over time [30]. Finally, Wanrooij and Pras [31] have introduced an heuristic to tell if a message is spam or not based on uniform resource locators (URLs) within a message and on the neighborhood of the sender's IP address, coining the term *Internet Bad Neighborhoods*.

The combination of these two factors – (*i*) that malicious hosts are distributed all over the world and (*ii*) that they are more concentrated in certain networks – resembles the distribution of crimes in the real world. For example, Figure 1.3, shows the distribution of motor vehicle theft in the continental

[1] We use the CIDR notation for network blocks/prefixes [25]. Please refer to [26] for a brief description on the subject matter.

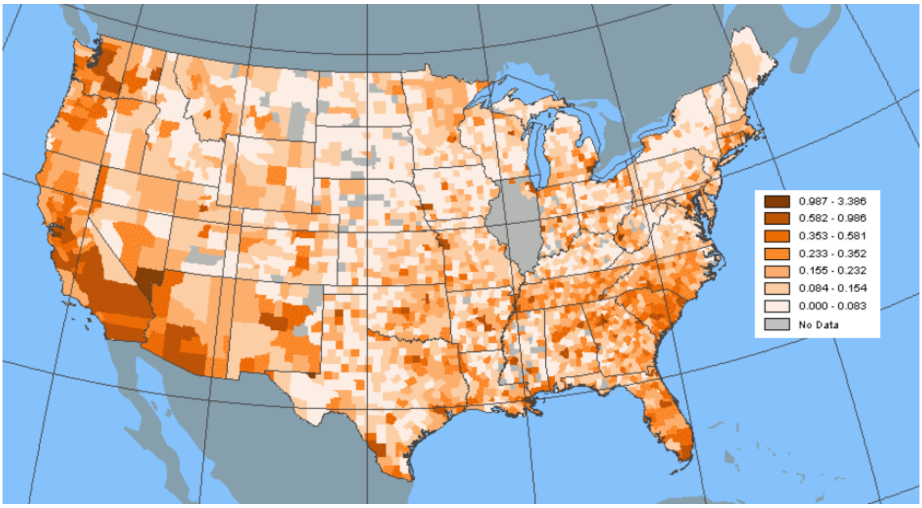


Figure 1.3: Percentage of Population Affected by Motor Vehicle Thefts (2007)
Source: *National Atlas* [1]

United States in 2007. As can be observed, vehicle theft occurs all over the country, but it is more concentrated in some areas than others.

This resemblance between the real world and the Internet regarding the crime sources distribution and concentration lead us to the topic of this dissertation: *Internet Bad Neighborhoods*. In the real world, locations having higher crime rates than the average are sometimes called *bad neighborhoods*. In such places, it is statistically more likely that a crime will occur compared to other locations. The same principle holds for Internet Bad Neighborhoods: it is more likely that malicious activities are originated from such networks than from other networks.

To better illustrate this analogy, consider the case of New York City. Figure 1.4 shows the homicide locations in the city from 2003 to 2011. As can be seen, some neighborhoods have higher homicide rates than others. If the New York Police Department (NYPD) wants to reduce crime more efficiently, a starting point would be by improving the police coverage where the homicides are concentrated – for example, in neighborhoods like Brooklyn or Bronx. On the other hand, if a random person wants to be *statistically safer*, he/she should also avoid neighborhoods having higher crime rates.

The same principle applies to Internet Bad Neighborhoods. If network secu-

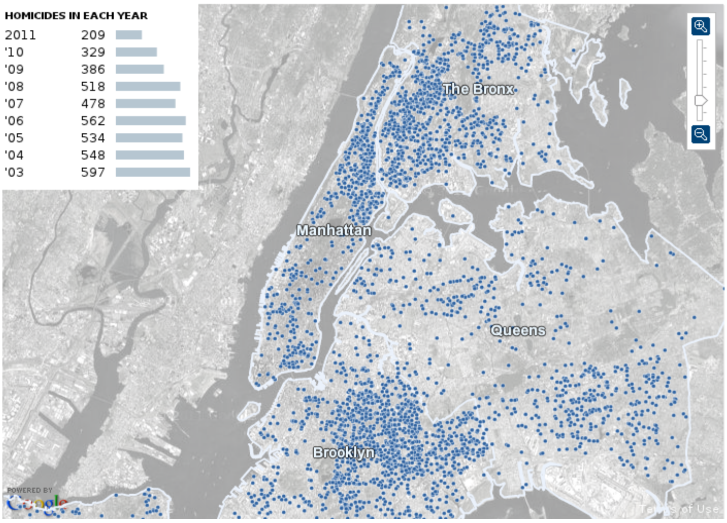


Figure 1.4: New York City Homicide Map (2003-2011)
Source: *New York Police Department* [2]

rity engineers (analogous to the NYPD) want to reduce the incidence of attacks on the Internet, they should start by tackling networks where attacks are more frequently originated. If a user (in analogy to the random person in the real world example) wants to be safer on the Internet, he/she should avoid (or at least be much more careful) connecting to computers located in such networks.

The list of Bad Neighborhoods, both in the real world and on the Internet, are usually compiled into what is popularly known as *blacklist*, which is a form of access control mechanism to allow an entity (e.g., users) to access a particular resource with exception of those entities listed [32]. On the Internet, blacklists containing IP addresses of spam senders have been used for years to filter out spam [33].

In the real world, some businesses have generated bad neighborhoods blacklists with locations they would not operate for security reasons. For example, the logistics company DHL has created a blacklist containing certain parts of London, Manchester, Glasgow, and Birmingham they would not deliver packages [34]. Microsoft has recently been granted with a patent for a Global Positioning System (GPS)-based navigation system that allows drivers and pedestrians to avoid routes through neighborhoods having high-crime rates [35] (the patent is popularly known as “avoid-ghetto” patent and has generated significant controversy [36]).

On the Internet, the main usage of the Bad Neighborhood concept is to protect network targets, by being able to statistically predict attacks from unforeseen IP addresses – which is covered in details in Section 2.7. With this purpose in mind, Wanrooij and Pras [31] have introduce the Bad Neighborhood concept for spam filtering. Whenever a new message arrives, the algorithm checks if neighbor IP addresses of the sender (*i.e.*, hosts within the same subnetwork) have been previously blacklisted and uniform resources locators (URLs) in the message. The probability of a message being spam increases if neighboring IP addresses are also spammers.

Even though the Internet Bad Neighborhood concept was proposed and employed to filter out spam [31], the very concept was not investigated in more details. This dissertation, however, focuses on a *multifaceted investigation* of the Internet Bad Neighborhoods phenomenon, and not only as an heuristic to determine the odds of a message being spam. As we shall see, we address many different aspects of Internet Bad Neighborhoods, including the basic characteristics and how to protect a network against attacks from Internet Bad Neighborhoods.

In the following, we first present our definition of Bad Neighborhoods in Section 1.1. Then, in Section 1.2, we present the goal, research questions, and approach employed in this dissertation. After that, we summarize in Section 1.3 the contributions of this dissertation, and the scope and limitations in Section 1.4. Finally, the outline of the dissertation is detailed in Section 1.5.

1.1 Defining Internet Bad Neighborhoods

In this section we present the formal definition of Internet Bad Neighborhoods used throughout this dissertation:

Definition 1. *An Internet Bad Neighborhood is a set of IP addresses clustered according to an* ***aggregation criterion*** *in which* ***a number of IP addresses*** *perform* ***a certain malicious activity*** *over a specified* ***period of time****.*

In this definition, *aggregation criterion* stands for the basic building block used to cluster malicious IP addresses into Bad Neighborhoods. Different criteria can be employed for this purpose. The main one is the IP addressing scheme. By using this criterion, we can aggregate IP addresses according to network prefixes (e.g, /24, /8, /18, in Classless Inter-Domain Routing (CIDR) notation [25]). Alternative criteria can be employed, such as geographical location (e.g., countries, cities, as in Figure 1.1) or also according to the network's Autonomous System Number (ASN) [37] of the Internet Service Provider (ISP). In this dissertation, we cover all these criteria.

The *number of IP addresses*, on the other hand, refers to the number of malicious IP addresses that were observed carrying out attacks. It is important to emphasize that this number might differ from the total number of IP addresses in the neighborhood, since some IP addresses within the bad neighborhood could actually be "good IP addresses". For example, an IP-based /24 Bad Neighborhood, such as 10.10.10.0/24, has a fixed size of 256 IP addresses. However, it can be possible that only a fraction of those were observed carrying out malicious activities, and some of those addresses are not even in use. The same principle applies for bad neighborhoods in the real world: there are innocent citizens living in such places.

A *certain malicious activity*, in turn, is related to the application that the bad neighborhood is abusing or conducting attacks on (e.g., spam, SSH brute force attacks, phishing). Therefore, a single host might belong to different Bad Neighborhoods that differ in relation to the application.

Finally, *period of time* refers to the time frame used to define a bad neighborhood (e.g, day, weeks). This is an important variable since bad neighborhoods are expected to change over time – since machines are expected to get compromised and cleaned up regularly.

1.2 Goal, Approach, and Research Questions

The goal of this dissertation is to scrutinize the Bad Neighborhood phenomenon on the Internet to better understand its intrinsic characteristics, so we can protect networks from Bad Neighborhood attacks. The general approach employed consists in monitoring and analyzing network data (traces, blacklists, etc.) obtained from real world production networks. The idea is to analyze such data sets and learn how Bad Neighborhoods behave on the Internet, so we can develop techniques that allow network administrators to better secure networks. To accomplish this, we propose and answer two main research questions:

- **Research Question 1 (RQ 1):** What are the characteristics of Internet Bad Neighborhoods?

RQ 1 focuses on *scrutinizing the Bad Neighborhood phenomenon*, by providing an investigation on why it occurs on the Internet, how they can be found, and why it is a worth using the concept to *predict attacks sources* on the Internet. In addition, we propose and evaluate algorithms to cluster malicious IP addresses into Bad Neighborhoods according to the IP addressing scheme, ISPs, countries, cities, and organizations.

After scrutinizing the Internet Bad Neighborhood phenomenon, we then assume the point of view of a network administrator who wants to defend a network against such bad neighborhoods. To carry out this, we employ *blacklisting*, which has been employed as access control method to filter out spam sources for many years [33]. Alternatives to that would be *whitelisting* – lists of IP addresses that are allowed to use a resource – and greylistings. Whitelisting is not considered in our study since it does not provide the necessary scalability to deal with the large number of IP addresses on the Internet. Greylisting (in which a mail server "temporarily rejects" a source [38]) does not suit our purposes either, since it is tailored only to spam – while the bad neighborhood definition can be employed to various applications.

Therefore, the second research question addressed in this dissertation is:

- **Research Question 2 (RQ 2):** Which blacklists should a network administrator choose to protect a network against attacks from Internet Bad Neighborhoods?

In RQ 2 we focus in providing networks administrators with insights on how to choose bad neighborhood blacklists obtained from different sources. Moreover, for this RQ, we evaluate how specific bad neighborhood blacklists are in relation to an application, determining if they can be employed to protect attacks to applications they were not originally intended. Finally, we also address the temporal attack strategies employed by bad neighborhood in order to determine how often blacklists should be updated and provide insights on when to expect attacks.

1.3 Contributions

The contribution of this dissertation is to present, to the best of our knowledge, the first *systematic* and *multifaceted* study on the Bad Neighborhood phenomenon on the Internet. By first acknowledging and verifying the Bad Neighborhoods existence on the Internet, we then scrutinize Internet Bad Neighborhoods in a multifaceted approach in order to reveal their characteristics and provide network administrators with guidelines to protect networks from attacks originated from Bad Neighborhoods.

The main contributions of this dissertation are:

- A formal definition for Internet Bad Neighborhoods;
- A discussion on the ethical implications of the Internet Bad Neighborhood concept;
- Two application-independent algorithms to aggregate malicious IP addresses into Bad Neighborhood of various IP prefix sizes;
- An investigation of the Bad Neighborhoods not only in the IP addresses, but also in relation to ISPs, organizations, countries, and cities;
- A study case on spamming Bad Neighborhoods, in which the specifics of spam are leveraged to the Bad Neighborhood concept;
- An evaluation of the efficacy of employing third-party Bad Neighborhood blacklists to protect IP addresses on other networks;
- An evaluation on the overlap between Bad Neighborhoods associated with different applications;
- A comprehensive analysis on the temporal attack strategies employed by Bad Neighborhoods when attacking targets.

The contribution provided in this dissertation aims at providing network administrators and networks security engineers with information to better develop security tools and protect networks.

1.4 Scope and Limitations

The bad neighborhood concept is aimed at dealing with attacks that employ a large number of distributed hosts, such as DDoS and spam campaigns. However, as other security approaches, it does not cover all types of Internet attacks. For example, highly sophisticated and precisely targeted cyber-weapons, such as StuxNet, are likely to be stealthy as much as possible, and therefore, likely not captured by Bad Neighborhood-based security systems (StuxNet is the first confirmed cyber-weapon designed by a nation state [39], developed to subvert industrial systems located at Iranian uranium enrichment facilities).

In addition, in this dissertation we evaluate only IPv4 Bad Neighborhoods. Currently, IPv6 [40] traffic accounts for less than 1% of the total traffic observed in networks such as Internet2 [41] and the Amsterdam Internet Exchange Point (AMS-IX) [42]. Due to that, IPv6 attacks remain relatively rare – only in 2012 the first IPv6 DDoS attacks were reported [43]. With the increasing adoption of IPv6, we can expect more attacks from IPv6 Bad Neighborhoods. To cope with that, we present in Appendix C an analysis on what to expect from IPv6 Bad Neighborhoods. As we show in Appendix C, the Internet Bad Neighborhoods approach *is a requirement* to help blacklist-based security systems to cope with the vast number of valid IPv6 addresses.

1.5 Dissertation Outline

Figure 1.5 outlines the structure of this dissertation, divided in four parts, each of them having a different emphasis on the Internet Bad Neighborhoods phenomenon.

In Part I (Introduction), we present the introduction to this dissertation and the background information. We cover the formal definition, an approach to locate bad neighborhoods on the Internet, and we verify the Bad Neighborhoods assumption. In addition, we cover the ethical issues and values involved in this research.

In Part II (Characteristics), we address RQ 1 ("What are the characteristics of Internet Bad Neighborhoods?"), by covering Bad Neighborhood aggregation as well as their location, and a case study in which we tailor the Bad Neighborhood definition to the spammer's specifics.

In Part III (Defending against Bad Neighborhoods), we investigate RQ 2 ('Which blacklists should a network administrator choose to protect a network against attacks from Internet Bad Neighborhoods?"), by showing how a network administrator can protect the network he/she maintains by employing Internet Bad Neighborhoods blacklists from different sources and applications. In addition, we investigate the temporal attack strategies employed by Bad Neighborhoods.

Finally, in Part IV (Conclusion), we present the conclusions of this dissertation.

Following this structure, we divide Part I into the following chapters:

- In **Chapter 1 – Introduction**, we present the introduction to this dissertation.

- In **Chapter 2 – Background**, we show three possible reasons that had helped to emergence of Internet Bad Neighborhoods. Also, we propose

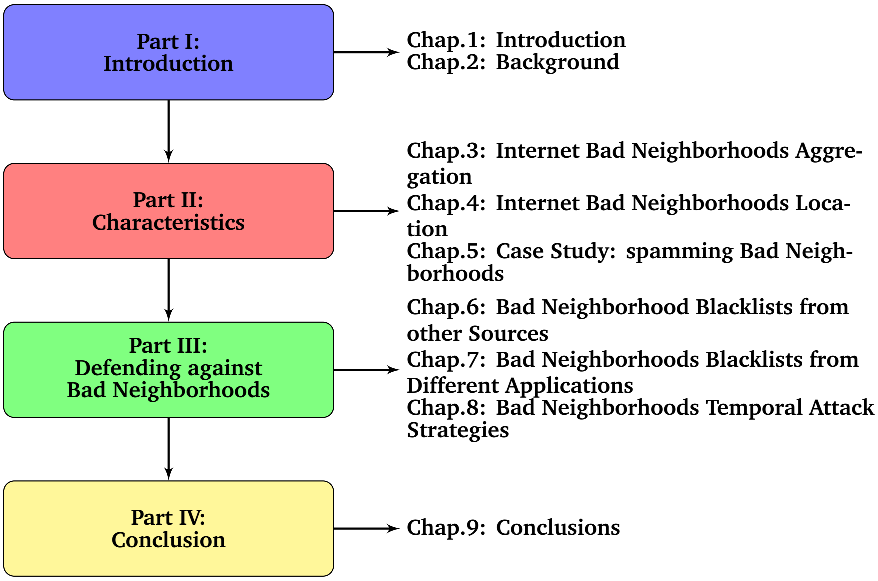

Part I:
Introduction
Chap.1: Introduction
Chap.2: Background
Part II:
Characteristics
Chap.3: Internet Bad Neighborhoods Aggregation
Chap.4: Internet Bad Neighborhoods Location
Chap.5: Case Study: spamming Bad Neighborhoods
Part III:
Defending against
Bad Neighborhoods
Chap.6: Bad Neighborhood Blacklists from other Sources
Chap.7: Bad Neighborhoods Blacklists from Different Applications
Chap.8: Bad Neighborhoods Temporal Attack Strategies
Part IV:
Conclusion
Chap.9: Conclusions

Figure 1.5: Dissertation Outline

an approach to locate Internet Bad Neighborhoods and discuss the related issues. In addition, we carry out an experiment to verify the Bad Neighborhoods assumption – proving that it is an worthy idea to predict new sources of attacks on the Internet. Last, we address the ethical issues implicated by the Internet Bad Neighborhood concept.

In Part II, we provide three chapters that investigate the characteristics of Internet Bad Neighborhoods:

- In **Chapter 3 – Internet Bad Neighborhoods Aggregation**, we propose two approaches to aggregate Internet Bad Neighborhoods into network prefixes and evaluate them, employing real world data sets.

- In **Chapter 4 – Internet Bad Neighborhoods Location**, we reveal where are the Internet Bad Neighborhoods concentrated – in terms of countries, cities, Autonomous Systems [37]), and organizations.

- In **Chapter 5 – Case Study: spamming Bad Neighborhoods**, we take spam Bad Neighborhoods as a case study and refine our general definition of Internet Bad Neighborhoods.

In Part III we focus on protection against bad neighborhoods, by providing three chapters:

- In **Chapter 6 – Bad Neighborhood Blacklists from other Sources**, we determine what is the best strategy to generate Internet Bad Neighborhood blacklists: (*i*) trust others or (*ii*) carry out local measurements.
- In **Chapter 7 – Bad Neighborhoods Blacklists from Different Applications**, we investigate if there is a significant overlap between Internet Bad Neighborhood blacklists obtained from one application in relation to another application.
- In **Chapter 8 – Bad Neighborhoods Temporal Attack Strategies**, we scrutinize the temporal strategies employed by bad neighborhoods to carry our their malicious activities.

In Part IV, we present **Chapter 9 – Conclusion**, in which we finalize this dissertation, by providing the reader with the main contributions of this dissertation as well as guidelines for future work.

Our minds are adapted to a world that no longer exists, prone to misunderstandings correctable only by arduous education, and condemned to perplexity about the deepest questions we can ascertain.

Steven Pinker, 2002
In: The Blank Slate: The Modern Denial of Human Nature

CHAPTER 2
Background

In this chapter we provide background information on Internet Bad Neighborhoods. We start by discussing the reasons that have led to the existence of Bad Neighborhoods on the Internet in Section 2.1. Next, we proceed by presenting an approach to locate Internet Bad Neighborhoods in Section 2.2 and the issues associated with each step of the approach in Sections 2.3–2.6. Then, in Section 2.7, we scrutinize the Bad Neighborhood assumption, and evaluate it experimentally. Finally, in Section 2.8 we discuss the ethical implications associated to this Bad Neighborhood concept.

2.1 Why Internet Bad Neighborhoods Exist

We assume in this dissertation that the existence of Internet Bad Neighborhoods – *i.e.*, concentration of malicious hosts in certain networks – is due to three possible reasons:

1. *Some Internet Services Providers (ISPs) neglect malicious activities in their networks.*

2. *Whenever a host is infected by a malware, it is more likely that this malware is going to succeed in infecting neighboring hosts belonging the same badly managed network than hosts in well managed networks.*

3. *Non-technical local factors may contribute, such as the rate of software piracy, legislation, culture, economic, education level in a country.*

The first reason for the existence of Bad Neighborhoods on the Internet is that we can expect different ISPs to have security policies differing on effectiveness. As discussed by Ramachandran *et al.* [44], there are some ISPs that “turn a blind eye” to the problem in their networks. An extreme case of it it is when the ISPs is *deliberately* engaged in malicious activities, as the case of McColo Corp.. When McColo was disconnected from the Internet by two of their upstream providers (Global Crossing and Hurricane Electric) due to the large amount of malware and botnets in their networks [45], several reports have shown that the volume of worldwide spam was reduced in 2/3 [46].

In such “malware tolerant” ISPs, one can also expect also malware to be more successful in infecting other neighboring hosts [47] (*second reason)*. These hosts, in turn, usually become part of botnets under control of a botmaster (a review on the rise of botnets is covered in Appendix B). Ultimately, this contributes even more the concentration of malicious hosts and occurrence of BadHoods in such ISPs.

Finally, non-technical local factors (*third reason*) may also contribute to the BadHood phenomenon. One could expect that ISPs are more likely to neglect malicious traffic in their networks if there is no Internet crime legislation in their countries (e.g., the United States has a specific anti-spam legislation [48], as well as the European Union [49]). In addition, one could expect countries having high levels of software piracy to be more likely to run outdated and therefore more vulnerable software.

It is important also to mention that there *is an economic drive behind these assumptions*. Cyber-gangs continue on carrying out malicious activities on the Internet simply because there is a profitable business model — which is not in the scope of this dissertation. On this topic, however, McCoy *et al.* [18] have analyzed “leaked” business data from illegitimate online pharmaceutical affiliate programs and shown that “online sales of counterfeit or unauthorized products drive a robust underground advertising industry that includes email spam[...]”, showing a profit margin of 10-20%. Since the recruitment of new customers is heavily based on e-mail spam [18], there is a business demand for effective spamming methods – which provides incentive for having more compromised hosts, mostly likely to be observed in the networks of poorly managed ISPs in more permissive countries.

We investigate these assumptions in Chapter 4.

2.2 Finding Internet Bad Neighborhoods

In the real world, crime statistics are of importance when deciding if a neighborhood should be considered “bad” or not. These statistics are generated by companies, police departments, and governments, by keeping track of mali-

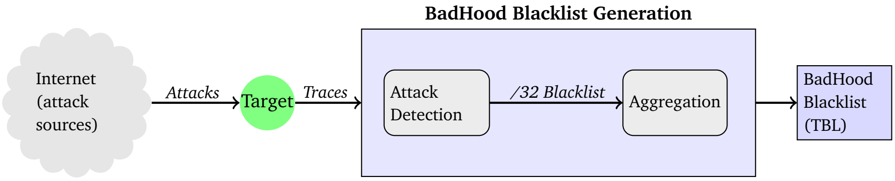


Figure 2.1: Approach to Find Internet Bad Neighborhoods

cious activities perpetrated in neighborhoods, based on the reports and charges pressed by the victims.

We propose an analogous approach to find Internet Bad Neighborhoods (BadHoods in the rest of this dissertation). The idea is to compile statistics per neighborhood based on the security incidents observed by targets (analogous to victims), which are devices connected to Internet.

Figure 2.1 summarizes the approach we propose to find Internet BadHoods. In the first step, malicious *sources* on the Internet carry out attacks against a *target*. After being attacked, the target feeds the *attack detection* system with information related to the attack (e.g., trace files) so attacks can be detected. These trace files are processed and the sources of the attack are identified based on the source IP address. In addition, other data might be obtained from the IP packets, such as timestamps, number of bytes, etc. After that, a blacklist containing the IP addresses of the sources is generated (a so-called/32 blacklist) and used as an input to the aggregation process, in which sources get aggregated into BadHoods, according to an aggregation criterion (e.g., IP prefix such as /24, or geographical information). In the end, a final *BadHood blacklist* is generated (we use the term throughout this dissertation to refer to a list of malicious Bad Neighborhoods and to differ from traditional blacklists).

In the next sections we present more details about each step involved in the proposed approach.

2.3 Attack Sources and Attribution

Attack sources are devices connected to the Internet that are involved in the attack to a particular target. Theoretically, any host connected to the Internet is a potential malicious source. Traditionally, desktop/laptops have been the main source of attacks on the Internet. However, we can expect in the near future more attacks to be originated from mobile devices (e.g., smart phones, as in the

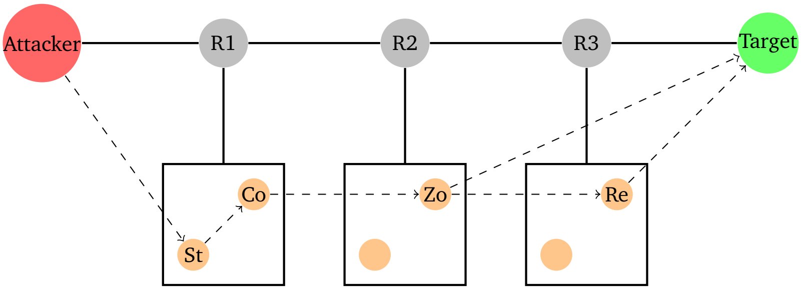


Figure 2.2: Attribution Problem (adapted from Wheeler and Larsen [3])

case of the recently found Android-based botnet [50]) as well as from devices that, in the past, were not connected to the Internet and currently are (part of the so-called “Internet of Things”), such as TV sets, satellite receivers, Blu-ray players, refrigerators, SIP phones, just to mention a few.

Identifying the *responsible attacker* for the attack is referred in the literature as *attack attribution*, that is, “determining the identity or location of an attacker or an attacker’s intermediary” [3]. As defined by Wheeler and Larsen [3], identity may be the attacker’s user name, name, alias, or related information associated with the person orchestrating the attacks. Location, on the other hand, refers to attacker location in terms of geographical location or virtual location (e.g., IP address).

As in the real word, smart attackers try at any cost to make attribution more difficult on the Internet. In this sense, attackers commonly employ *intermediary nodes* between themselves and the target system. By employing such hosts, attackers hide their identity, since IP packets perceived as attacks at the target appear to be originated from the intermediary hosts.

Figure 2.2 illustrates the attack attribution problem. In this figure, solid lines represent network links, and circles R1, R2, and R3 are the routers connecting the attacker to the target. Each router is connected to a local network (square), to which hosts (orange circles) are connected. To illustrate the attribution problem, consider that the attacker in Figure 2.2 is a botmaster controlling a botnet (botnets are currently one of the major security threats on the Internet – see also Appendix B for more on this matter). Consider also that the target is a legitimate e-mail server.

Instead of attacking directly the target, the attacker uses another logical path (dashed line in Figure 2.2) to hide his original identity. First, the attacker connects to a stepping stone node (St) – which is a host used to redirect the connections from the attacker to the Co, the Command and Control center of the botnet. Multiple St hosts can be used in this process. After connecting to the Co, the attacker sends the commands to the command and control (Co), which then send the orders to a zombie (or a set of zombies, as Zo), which are the machines that actually carry out the spam campaigns, ending up at the target. Optionally, zombies can employ reflector hosts (Re), which works like a proxy between the target and the zombie, hiding the zombie identity. At the end of the process, the target receives the attack (e.g., a spam message) having the source IP address of the zombie (Zo) or the reflector host (Re).

To make the attribution problem even more complex, the attacker may benefit from other network features, such as network address translation (NAT), which changes the source and destination field address of the IP packet header. Also, the intermediary hosts may be connected to the network using dynamic IP addresses, which may frequently change over time. Moreover, since the IP source address of the attackers is not used in the routing process, it may be easily forged, which is commonly known as IP spoofing [51]. Other techniques can also be employed; for a more detailed view on the matter, please refer to the work of Wheeler and Larsen [3].

The approach presented in this dissertation, however, focuses on the *attribution of the last host* in the logical path of the attacks (Zo or Re). In this sense, *Bad Neighborhoods are ultimately vulnerable networks having compromised machines*, which may or may not be intermediary hosts between the actual attacker and the target (considering the IP address is not forged). As a consequence, hosts flagged as malicious might not represent the behavior of the host's owners, who actually might be unaware that his/her computer is involved in such attacks (we discuss the ethical implications of this in Section 2.8).

We choose to focus on the attribution of the last host because we assume the point of view of a network administrator who wants to protect a network from malicious sources. For the network administrator, knowing the identity of the attacker does not help to better protect the network he/she maintains, since blocking traffic from the attacker IP address to the network the administrator maintains does not stop spam messages from originating from Zo or Re in Figure 2.2. In contrast, we see the attribution of the responsible attacker as a task of cyber police forces instead. Such type of research is outside the scope of this dissertation.

2.4 Targets

In the scope of this dissertation, we define as a *target* any device connected to the Internet that is victim of attacks carried out by malicious sources. Traditional examples of targets are servers and desktop/laptops. However, mobile devices (e.g., smart phones, tablets) are also potential targets, as devices such as TV sets, refrigerators, sound systems, media players, as long they are connected to the Internet.

As shown in Figure 2.1, the generation of BadHood blacklists is coupled with the monitoring of one or more targets. We refer to the resulting BadHood blacklists as Target's BadHood List (TBL), because it lists the neighborhoods attacking that particular target. This, however, does not imply that the particular target has observed *all existing BadHoods*.

To observe more Bad Neighborhoods, one idea is to monitor a large number of targets and generate a single blacklist. However, this also does not guarantee that all existing BadHoods are listed in the resulting BadHood blacklist.

One approach to generate a *complete* BadHood blacklist would be to monitor every single target on the Internet and generate a single BadHood blacklist. This approach, however, is unfeasible for a series of reasons, the main one being the sheer size and complexity of the Internet. Monitoring the whole Internet and then coordinate efforts to share the resulting BadHoods imposes challenges that go well beyond technical problems, including legal and ethical issues.

2.5 Data Collection and Attack Detection

In order to locate BadHoods on the Internet, we have to obtain network data (e.g., traces) and perform the attack detection. Several sources of data can be used in the data collection process, and the data is classified according to the monitoring point:

- Target-centric data sources: this category encompasses monitoring the various applications and incoming traffic to the target. For example, a network administrator might monitor all the network traffic (in PCAP format [52]) to an individual server.

- Network-centric data sources: this category covers monitoring the network that the target is connected to instead of the host directly. For example, consider the network router (such as `Rn` in Figure 2.3). In this case,

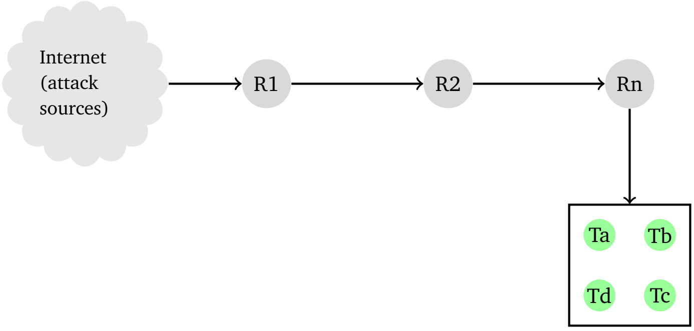


Figure 2.3: Data Collection

a network administrator could monitor the network flows [53] exported from the particular router in order to detect attacks.

After obtaining the data, the next step consists in detecting attacks. In the literature, various techniques are employed to detect attacks. Intrusion Detection System (IDS), for example, are classified according to the technique employed to detect attacks. Signature-based IDS compares network traffic to pre-determined attack patterns, which are popularly know as signatures. `Snort` is an example of a signature-based IDS [54]. The other type of IDS are anomaly-based IDS, which compare incoming data to a model of normality that describes the expected or "normal" behavior. Statistical analysis and Markov models are used for anomaly-based IDS [55, 56].

Attacks can also be detected by application servers. For example, the mail filter `SpamAssassin` analyses e-mail message contents against a set of signatures [57]. In addition, *honeypots*, which are essentially systems that act as traps to detect malicious activities, can be employed to detect attacks [58]. Finally, attacks can also be detected by correlating various sources of information. For example, OSSEC [59] is a host intrusion detection system (HIDS) that correlates various log files (e.g., SSH server, Web servers) to detect attacks.

In this dissertation, *we do not focus on the detection itself; rather we rely upon other systems/techniques for this particular purpose.* As a consequence, *the quality of the BadHood blacklist depends on the monitored data and techniques employed to detect attacks.* Due to that, errors might occur in identifying attacks and, consequently, false positives and false negatives can be expected. This ultimately impacts the correctness of the resulting /32 blacklist, which is the

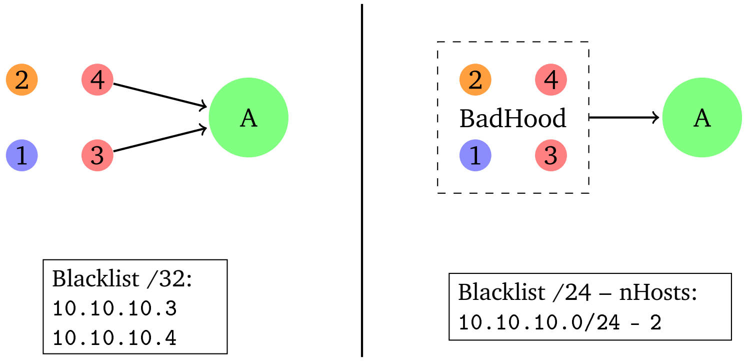


Figure 2.4: Aggregating Malicious Hosts into BadHoods

input for the aggregation process, to be described next.

2.6 Aggregating Hosts into Bad Neighborhoods

The main idea behind the Bad Neighborhood concept is the *aggregation of malicious hosts* according to an aggregation criterion – for example, IP address range, city, country, etc. The advantage of doing so is that it allows one to predict attacks from unforeseen sources (neighbors) before they occur (see Section 2.7 for the investigation of this assumption).

To show how the aggregation principle works, take as example Figure 2.4. In the left part of this figure, consider that `A` is a target (e.g., a mail server) while hosts on the Internet $(1-4)$ share the same prefix (`10.10.10.`x, where x is the number of the host within each circle).

In this figure, target `A` is attacked by hosts `10.10.10.3` and `10.10.10.4`. If `A` were to generate a blacklist, these two host would be listed, as shown in the same figure. By employing this blacklist, `A` could block any new attempts from *both hosts* in the future.

However, if `A` were to generate a blacklist employing the Bad Neighborhood concept, it would have to cluster these hosts under a common criteria. Let's consider, as an example, that `A` aggregates the malicious hosts into /24 prefix. Since hosts $(1-4)$ share the same 3 octets of the IP address (`10.10.10`), they can be aggregated into a single /24 prefix (`10.10.10/24`), which comprises all addresses in the range `10.10.10.0-10.10.10.255`. As a consequence, the only

/24 Netblock	Number of Sources (hosts)	Number of Attacks (spam)
117.242.140.0	90	198
87.252.243.0	64	133
41.254.0.0	227	814
84.36.152.0	40	103
188.50.57.0	20	30

Table 2.1: Example of a /24 BadHood Blacklist

entry in the BadHood blacklist would be `10.10.10.0/24`, as shown on the right in Figure 2.4. Associated with this neighborhood would be a numerical value indicating the number of observed malicious hosts (2 in this case).

This information would allow a mail filter to consider `10.10.10.0/24` as a single neighborhood, and judge equally all the 256 hosts in this netblock in case a new message arrives. This, in turn, allows A to be protected from *any* host from the same /24, or the same BadHood – which can be seen as a form of *predicting new attacking sources*, and not only to *reacting* to observed /32 sources. However, this comes at a price: in this neighborhood, hosts 1 (not malicious, in blue) and host 2 (malicious, but it has not attacked A yet) are both considered malicious, even though they have not yet attacked `A` – and may never attempt it.

Table 2.1 shows a sample of a /24 BadHood blacklist. It lists five randomly chosen /24 BadHoods (out of more than 500,000). For this BadHood blacklist, the target was the mail server of Provider A, a major hosting provider in the Netherlands. In the first column, the bad neighborhoods are listed, while in the second we list the number of distinct malicious sources that were observed sending spam; the number of spam messages is shown in the third column. For example, out of the 256 hosts listed in neighborhood `41.254.0.0`, 227 have actually sent spam to Provider A, in a total of 814 messages.

2.7 Verifying the Bad Neighborhood Assumption

As explained in Chapter 1, previous work has shown that malicious hosts tend to be *concentrated* in certain networks instead of being *evenly* distributed over the IPv4 address space [27, 28, 29, 30, 31]. As shown in Section 2.6, this has lead to the **Bad Neighborhoods assumption – that BadHoods can provide an indirect approach to predict new sources of attacks**, by assuming that neighboring hosts of malicious ones are more likely to be malicious as well and,

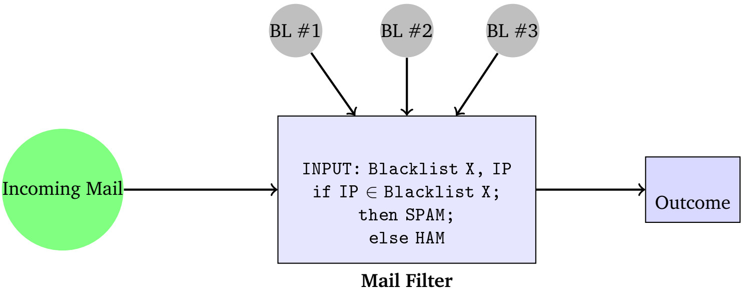


Figure 2.5: Simple Mail Filter Used in Evaluation of the Bad Neighborhoods Assumption

therefore, more likely to carry out attacks.

In this section, we verify the Bad Neighborhood assumption. To do so, we carry out a case study to determine if Bad Neighborhoods are useful in predicting new spam sources, by comparing spam detection rates when BadHoods are employed in relation to non-BadHood-based solutions.

With this purpose in mind, we consider a simplistic mail filter, as shown in Figure 2.5. To verify if a message is spam or not, the mail filter looks up the sender IP address (`IP`) in the blacklist (`Blacklist X`, which can be one of the three blacklists `BL#1-BL#3` in the same figure). If the sender's IP address is found in the blacklist, the message is considered malicious (spam), otherwise is considered legitimate (ham).

We verify the Bad Neighborhood assumption by comparing the performance delivery by the mail filter when using three different blacklist (`BL #1`, `BL #2` , `BL #3`), all generated based on the public spam blacklist Composite Black List (CBL) [60]. As input data, we use the IP addresses of spammers observed by the Electrical Engineering, Mathematics, and Computer Science Faculty of the University of Twente (UT/EWI), for a period from November 11th to 23rd, 2011. The list of spammers was extracted from the log files of SpamAssassin [61], the mail filter employed in the mail servers.

In Section 2.7.1 we present more details about the generated blacklists, while in Section 2.7.2 we show the UT/EWI data set used as input. Finally, in Section 2.7.3, we present and discuss the results.

2.7.1 Blacklists Evaluated

In the previous section we have stated that we evaluated three blacklists to verify our assumption. Many criteria can be used to generate blacklists, and there are many blacklists publicly available on the Internet (a comparison of blacklists from various sources is covered in Chapter 6).

We have chosen to employ the Composite Blacklist (CBL) [60] as the standard blacklist, that is, the blacklist that is employed as a comparison standard. CBL was selected because (i) it has been previously investigated by academic and Internet security communities [62, 63, 64, 65, 66], and (ii) the CBL provides bulk-access to the blacklist data, to ensure we have a complete view of the malicious IP addresses. We call this blacklist `CBL32-STD`. We have downloaded CBL once a day, and we therefore have one blacklist per day; we used the same monitoring period for both blacklists and the incoming mail.

From the standard blacklist (`CBL32-STD`), we create, for each day, a /24 Bad Neighborhood list, as described in Section 2.6. We refer to this blacklist as `CBL-BadHood24`. By comparing the spam detection rates of this BadHood blacklist against `CBL32-STD`, we can observe the improvement on the detection rate incurred by using the Internet Bad Neighborhood concept.

However, as also discussed in Section 2.6, by using the BadHood concept, we are able to protect from all the addresses from the neighborhood even if we have only observed a single host belonging to the neighborhood. In practice, it is as if we have blacklisted all /32 (256 addresses) from each /24 prefix. Therefore, by comparing directly the performance of `CBL-BadHood24` list to the `CBL32-STD` list is *not* a fair approach, since `CBL-BadHood24` *lists, equivalently, 256 times more hosts than* `CBL32-STD` (in the worst case scenario, of one malicious addresses per /24). Therefore, in this case, one could expect `CBL-BadHood24` to deliver a better detection rate than `CBL32-STD`.

To create a fair comparison, we create a third blacklist, to which we refer as `CBL32-EQUIV24-RND`. The idea behind this blacklist is create an /32 blacklist with an equivalent number of hosts as `CBL-BadHood24`, and, therefore, provide a fair comparison. However, instead of using the Bad Neighborhood assumption (that a host is more likely to be malicious if its neighboring hosts are malicious), we use *random* hosts. If the performance delivered by employing `CBL32-EQUIV24-RND` is similar to the one delivered by `CBL-BadHood24`, then the BadHood assumption is false – that is, malicious hosts are randomly distributed instead of concentrated in certain networks. Otherwise, the approach is valid.

To illustrate how the blacklist generation works, consider Table 2.2 [1]. On

[1] We do not have CBL data for November 18th due to a problem in the download script.

Day	# `CBL-BadHood24`	**/32 Equiv**	# `CBL32-STD`	**# Diff**
10	819,698	209,842,688	6,452,536	203,390,152
11	812,217	207,927,552	6,470,870	201,456,682
12	809,268	207,172,608	6,668,200	200,504,408
13	798,345	204,376,320	6,663,645	197,712,675
14	792,098	202,777,088	6,701,460	196,075,628
15	795,763	203,715,328	6,915,802	196,799,526
16	803,126	205,600,256	7,113,607	198,486,649
17	812,598	208,025,088	7,402,563	200,622,525
18	NA	NA	NA	NA
19	808,819	207,057,664	7,402,770	199,654,894
20	798,169	204,331,264	7,247,960	197,083,304
21	792,489	202,877,184	7,230,438	195,646,746
22	797,062	204,047,872	7,233,956	196,813,916

Table 2.2: Blacklists Used in the Mail filter

November 10, for example, we have obtained after aggregating CBL blacklist into /24, 819,688 Bad Neighborhoods. This, in turn, is equivalent to 209,842,688 /32 hosts (/32 Equivalent, by multiplying each BadHood by 256, which is the number of hosts in a /24 BadHood [25]).

Therefore, to create a fair comparison, we create `CBL32-EQUIV24-RND` blacklist (one per day) using two steps:

- Use all inputs of `CBL32-STD` for the same day (to make sure that these /32 hosts are also blocked here)

- Add x new *random* /32 IP addresses to the list, where x is shown in the `Diff` in Table 2.2 for the particular day.

In doing so, we create a blacklist that is able, in terms of /32 entries, to block the same number of hosts as `CBL-BadHood24`. It is the difference between the *performance of* `CBL-BadHood24` *and* `CBL32-EQUIV24-RND` that tests the Bad Neighborhoods assumption, by comparing if a *BadHood-based blacklist* is more efficient than a *randomly* generated one, providing that both of them are able to block the same number of hosts.

To create the random IP addresses, we have developed a simple Java program. We have only considered valid unicast /8 prefixes in this process (e.g., we have not considered 127/8, multicast addresses, reserved /8, as described by IANA [67]). We have employed the method `nextInt(int)` available in the `java.util.Random` API, which generates “a pseudo-random, uniformly distributed integer value between 0 (inclusive) and the specified value (exclusive)" [68].

Even though we run the risk that some of addresses generated by our program may not have been in use (e.g., not allocated by the RIRs), we expect that they will represent less than 10% of the total[2]. Currently, there are 13.43 /8-equivalent IP addresses in the RIRs reserved pool. This corresponds to 8.6% of the 155.71 /8-equivalent addresses allocated. To compensate for this, we repeat the experiment 10 times and present the average results.

2.7.2 Incoming Mail

In order to evaluate the performance of the mail filter shown in Figure 2.5, we need three blacklists (already described) and incoming mail. We have considered, for this case the incoming mail of the Electrical Engineering and Computer Science Department of the University of Twente (UT/EWI), from November 11 to 23, 2011. The mail has been previously analyzed using SpamAssassin [61], and we have obtained the IP addresses of the malicious sources.

In this data set, all IP addresses are, therefore, malicious IP addresses which have sent at least one spam message. Table 2.3 presents more details about the data set. As can be seen, this mail server has observed, on average, 246,072 spam messages a day, over the monitoring period.

2.7.3 Mail Filter Performance Evaluation

In this section, we compare the blacklists described in Section 2.7.1 to the UT/EWI data set, as shown in in Table 2.3. In this comparison, we impose a day difference between the incoming mail blacklist (UT/EWI) and the blacklists `BL1-3` (e.g., we compare November 11th UT/EWI spam addresses to the blacklists generated based on November 10th).

Figure 2.6 shows the results of our evaluation (also shown in Table 2.4). We can see that the performance of `CBL32-STD` (by using the previous day CBL original /32 blacklist) allows us to block, on average, 54.33% of all the spam messages observed by UT/EWI, for each day. That means that by employing a blacklist containing individual hosts observed by CBL, we are able to filter out roughly half of UT/EWI spam, regardless the day. These results are to be used as a comparison standard to the other two curves.

[2] `http://www.potaroo.net/tools/ipv4/index.html`

Day	# Spam Messages	# of /32 Hosts
11	182,333	85,242
12	96,028	46,720
13	286,872	109,803
14	228,916	97,685
15	383,083	129,027
16	281,330	117,844
17	282,655	105,140
18	153,539	16,808
19	249,429	84,422
20	276,402	126,283
21	160,863	73,812
22	449,201	124,066
23	168,285	32,459
Average	**246,072**	**88,409**

Table 2.3: UT/EWI Data Set - November 2011

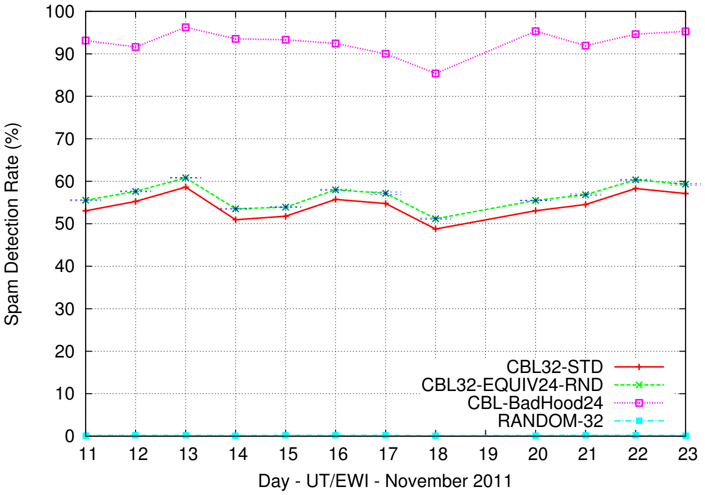


Figure 2.6: Performance of Various Blacklists

The curve of CBL-BadHood24 shows the improvements in the spam detection rate incurred by the use of the Internet Bad Neighborhood concept. Among all the data sets, this is the one that performs best, delivering, on average, 92.74%

Day	RANDOM-32	CBL32-STD	CBL32-EQUIV24[3]	SDEV[4]	CBL-BadHood24
11	299	96,734	101,263.8	220.77	169,813
12	170	53,067	55,336.4	104.07	87,973
13	527	168,225	174,506.6	181.05	276,159
14	336	116,629	122,529.3	264.82	214,116
15	711	198,392	206,590.1	252.81	357,509
16	512	156,811	163,147.9	366.00	260,054
17	524	154,811	161,573.3	852.97	254,450
18	175	74,874	78,594.8	215.28	131,098
19	NA	NA	NA	NA	NA
20	462	146,719	153,381	185.98	263,505
21	331	87,727	91,425	279.07	147,877
22	844	261,847	271,040.5	529.64	425,085
23	287	96,113	99,814.6	327.02	160,356
AVG	**431.50**	**134,329.08**	**139,933.61**	**314.96**	**228,999.58**

Table 2.4: Number of Spam Messages Detected According to Input Blacklist

spam detection. That means that blocking all the individual hosts listed on CBL and *their neighboring hosts*, we have been able to block most of UT/EWI spam.

However, to make it a fair comparison, we have to compare this results to the performance of CBL32-EQUIV24-RND, as discussed in Section 2.7.1. We have then generated the 10 random CBL32-EQUIV24-RND blacklists in order to eliminate statistical uncertainty. In Figure 2.6, the vertical bars on each point on CBL32-EQUIV24-RND represents the standard deviation obtaining from running 10 times the algorithm (also shown in Table 2.4). As can be seen, this blacklist performs far worse than CBL-BadHood24 , delivering an average performance of 56.64% spam detection. That means that even though both blacklists contain an equivalent number of /32 entries (CBL32-EQUIV24-RND and CBL-BadHood24), *blocking randomly chosen hosts will not significantly improve spam detection* (CBL32-STD to CBL32-EQUIV24-RND do yield comparable results).

The same conclusion can be drawn from the curve RANDOM-32, which is a /32 blacklist that has the same number of entries as CBL32-STD. The difference, however, is that they were all *randomly* generated (instead of being observed spamming CBL infrastructures). As can be seen, the effective detection rate of randomly generated IP addresses is very low for /32 blacklist, that is, almost 0%.

[3] Average for 10 experiments

[4] With relation to the 10 experiments of CBL32-EQUIV24

This confirms that selecting millions hosts at random does not improve spam detection (205 million, in average or 4.7% of the maximum theoretical in the IPv4 address space). On the other hand, *selecting neighboring hosts of malicious ones* to be blacklisted as well significantly improved the results. Therefore, we can conclude that *Bad Neighborhoods* are a much better approach to predict new sources of attack, when compared to random IP addresses – which validates the Bad Neighborhood assumption. This, in turn, supports the rest of the work presented in this dissertation. These findings also support similar results presented in [27, 28, 29, 30, 31], which were covered in Chapter 1.

2.8 Ethics and Internet Bad Neighborhoods

According to the University of Tennessee's Internet Encyclopedia of Philosophy, *Ethics* (or moral philosophy) is a field of philosophy that "involves systematizing, defending, and recommending concepts of right and wrong behavior" [69]. We believe that ethics should be taken into account in the development and the actual deployment of any technology, not only to ultimately obtain a more "Ethical product", but also to provoke a reflection on the impact of the technologies on individuals, society, and the environment.

As asserted in 1961 by the cybernetics pioneer Norbert Wiener, "individuals developing interactive technologies have an ethical responsibility to take likely consequences, positive and negative, of their designs into account" [70, 71]. The members of the Institute of Electrical and Electronics Engineers (IEEE) have also a code of Ethics to be followed [72], and the first article states that members "accept responsibility in making decisions consistent with the safety, health, and welfare of the public, and to disclose promptly factors that might endanger the public or the environment".

This dissertation is no exception. Even though the focus of this dissertation is not on the Ethical aspects associated with Bad Neighborhoods, we do provide *an introduction to the ethical issues* involved in the research and deployment of BadHood-based technologies. We recommend that before implementing the findings obtaining in this dissertation, the responsible persons should carry a complete ethical assessment following the guidelines presented here.

To assist us in the Ethical evaluation, we have carried out a series of interviews and discussions with Dr. Aimee van Wynsberghe[5], the Ethical Adviser of the Centre for Telematics and Information Technology[6] (CTIT) of the University

[5] http://www.aimeevanwynsberghe.com/

[6] http://www.utwente.nl/ctit/

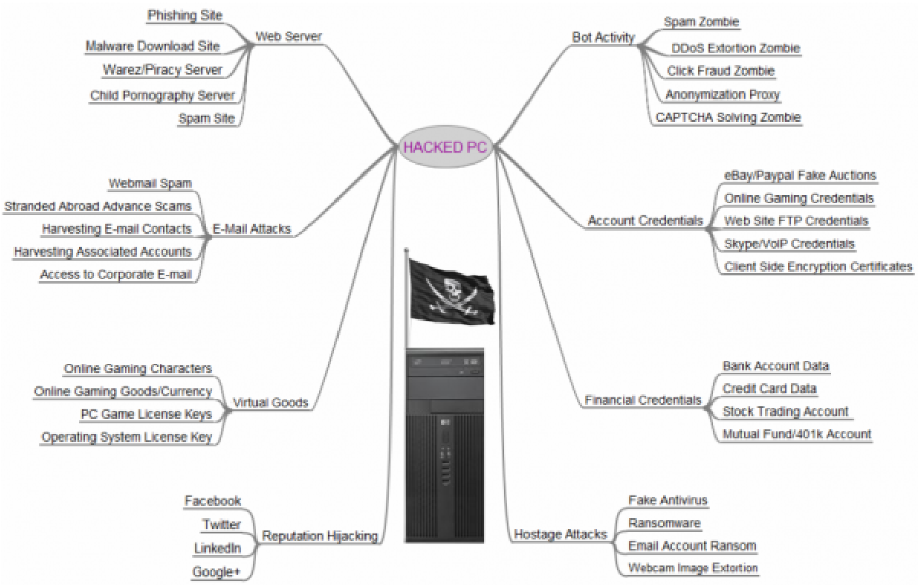


Figure 2.7: Possible Malicious Uses of a Hacked by Criminals (source: Brian Krebs[7], in The Washington Post [4], updated version from [5].)

of Twente.

2.8.1 Ethical Technology Design and Deployment

As discussed in Chapter 1, behind certain types of Internet attacks, we find a large number of compromised hosts, which are typically computers at homes, schools, and businesses that have been "hijacked" and carry out their malicious activities in a stealthy way so the human user behind the desk would not notice it. The investigative reporter Brian Krebs [4] published in his Washington Post column an article showing the reasons why criminals would hack standard PCs, summarized in Figure 2.7. As can be seen, there are many ways that a criminal would benefit from breaking into computers of unaware users.

In order to scrutinize the Internet Bad Neighborhood research under the ethics lenses, we follow the recommendations proposed by Nathan *et al.* [70],

[7] ©Brian Krebs – permission to use here, but not to reproduce anywhere else.

in which the authors propose four criteria to provide ‘perspective and focus for considering long-term system effects of current and future technologies”[70]. We present these criteria here and show a non-exhaustive list of examples on how our research relates to it:

1. *Stakeholders*: people that are directly or indirectly affected by the BadHood-related technologies:
    - ISPs that want to protect from attacks from BadHoods.
    - Users that have to be protected from attacks.
    - Users that have their computer hijacked but are unaware of it.
    - Cyber-gangs that want to exploit users’ resources.
    - Governments that want to “police” the Internet.
    - Security companies that want to employ and use the technology.
2. *Time*: the time that BadHood-related technologies are intended to be used; one should consider 3, 5, or 10+ years since most successfully deployed technologies “remain in use in society” for such extend periods[70].
3. *Values*: the set of values that “a person or group consider important in life”. We here present some of the values involved with the research (for Internet Research Ethics in particular, please refer to [73]):
    - Security: users want their computers and networks to be secure.
    - Fairness: users should not be wrongly punished by actions caused by others.
    - Economic value: ISPs and users want “clean” networks so less money and time hast to be invested to secure it.
    - Privacy: “principles of research ethics dictate that researchers must ensure there are adequate provisions to protect the privacy of subjects and to maintain the confidentiality of any data collected” [73].
4. *Pervasiveness*: the implication in case the technology become pervasive in the society. For example, who “controls” the one who is filtering network data based on BadHood-based technologies?

We next present the ethical dimensions related to our research and the criteria proposed by Nathan *et al.* [70].

2.8.2 Three Dimensions of the Ethical Issues and BadHoods

In this subsection we address three dimensions of Ethics and Internet Bad Neighborhoods research. We first explain each process, the ethical issues associated, and the proposed ethical solution.

Labeling Malicious Hosts

We have summarized in Figure 2.1 the approach we employ to find Internet Bad Neighborhoods. In Section 2.5, we have covered how the data collection and attack detection is performed. This is carried out to detect individual malicious hosts (/32) carrying out malicious activities.

Even though the detection of attacks on the Internet is somehow effective, the process itself is not error-prone. There is not currently a technology that is able to detect 100% of the security incidents. Moreover, *false positives* pose a threat to *fairness* value discussed in Section 2.8.1: if a system (e.g., a network intrusion detection system) wrongly classifies a certain network flow as malicious, the IP address assigned to it (and consequently, the user that belongs to this computer) will be taken into the blacklist, and, ultimately, into a Bad Neighborhood.

Not only that, the information used to create the /32 blacklist is the source IP address of the last "hop" host (Figure 2.2). However, the very IP address might be forged, and, consequently, many IP address that are not carrying out attacks may end up in the blacklist, which also violates the *fairness* value.

Finally, as shown in Figure 2.2, usually the IP address that is perceived as malicious is the last hop in a chain of many, which hides the original identity of the attacker. If the user is unaware his/her computer is being exploited to carry out malicious activities, is he/she responsible? To which degree? And to which degree software developers should be held responsible for releasing vulnerable technologies?

Taking all these issues into account, we acknowledge that the labeling process of individual hosts violates values such as fairness. However, blacklist based technologies have been used to filter spam since 1997 to provide other values: security to the users and ISPs and economic value (reducing potential damage caused by attacks). It is supported by the industry community in various products, such as `SpamAssassin` mail filter [57] and blacklists providers such as SpamHaus [74, 60, 75] as well as by the research community [76], to mention a few. Therefore, our answer to this ethical dimension is that this is the "best-effort" approach, that is, provides a compromise of values (analogous to

IP routing, which provides unreliable service, by not guaranteeing the delivery of network packets, but it does the "best it can" [77]).

Labeling Bad Neighborhoods

We have summarized in Figure 2.4 the aggregation of malicious hosts (/32) into Internet Bad Neighborhoods. In this process, we aggregate/cluster together hosts that have been observed carrying out malicious activities according to their IP addresses (or neighborhoods). As shown in the same figure, instead of judging an individual host as malicious, we label the whole "neighborhood" as well, considering the number of malicious hosts observed. By aggregating individual hosts into Bad Neighborhoods, we actually loose the ability of telling which hosts within the neighborhood are malicious, and actually judge them "equally" bad.

As discussed in Chapter 1, "in the real world, locations having higher crime rates than average are sometimes called *bad neighborhoods*. In such places, it is statistically more likely that a crime will occur compared to other locations. The same principle holds for Internet Bad Neighborhoods: it is more likely that malicious activities are originating from such networks than from other networks".

In this particular case, by aggregating individual hosts into BadHoods, we actually violate some ethical principles, by introducing *bias* and *prejudice* towards the other IP addresses in the cluster (which are associated to individual computers and, ultimately, users behind the desks). Such IP addresses may have never ever carried out an attack, but they are judged by the behavior of their neighboring IP addresses.

As an analogy to the real world, consider the scenario in which a bank has a list of clients that it would deny services due to previous problems in the past. If the bank would also include in this blacklist all the immediate neighbors of the ones previously listed, then the bank would have a Bad Neighborhood blacklist of customers.

We also acknowledge that the labeling and the aggregation into BadHoods is not ethically correct. However, this technique has proved to be effective and able to provide other values: security and economic values. As we have shown in Section 2.7.3, for IP-based BadHoods, such labeling has proven that the neighbors of malicious hosts are more likely to carry out malicious activities than randomly chosen neighborhoods of the same size.

The Deployment of Bad Neighborhood-based Technologies

The direct application of Bad Neighborhood-based technologies lies in providing Internet security engineers and software with information on the reputation of certain subnetworks (or BadHoods). Such information should be used in a way that *complements* current solutions for Internet security, and it is not intended to be the single technology for Internet filtering. In addition, the way it should be employed depends on the application being exploited (e.g., mail, ssh), due to the different application specifics.

One important observation is that BadHood-based technologies are intended to be used as *protective measures* towards incoming traffic. For example, a company would look up blacklists before accepting e-mail to its own mail severs. However, this does not interfere with the communications between malicious hosts and their own Internet Service Providers: they are "free" to connect to any other mail servers, for example.

To illustrate the envisioned use scenario, consider Figure 2.8. In this figure, we have three Internet Service Providers (ISPs): ISP A, B, and C. Consider, also, that each ISP provides e-mail services to its costumers. As shown in the figure, only ISP A makes use of a BadHood blacklist to filter e-mail messages. However, as we have mentioned, that is not the only criterion for classifying mail as spam or not (there are various different techniques that can be employed for that – e.g., content analysis). In fact, the BadHood blacklist can be used in the process when deciding if a message is spam or not. Also, in Figure 2.8, consider the neighborhood with 4 hosts (1–4), being 2 and 4 "malicious", that is, have been observed sending spam, while 1 and 3 are legitimate. We envision the following possible outcomes:

- *Legitimate hosts are penalized for the behavior of their neighbors*: that happens in the case of ISP A, which uses a BadHood blacklists which encompasses all hosts of the BadHood in the figure. However, as we pointed, BadHoods are not intended to be the single criterion in classifying a message. In the worst case, hosts 1-4 will not be able to send messages to ISP A. However, that does not affect their capability to communicate with ISP B and C, since these do not use BadHood-based systems (even if they would, they might not have the same neighborhood blacklisted, and they even might employ different criteria and weights when analyzing a message).

- *Malicious hosts are able to keep on spamming oblivious ISPs*: if malicious

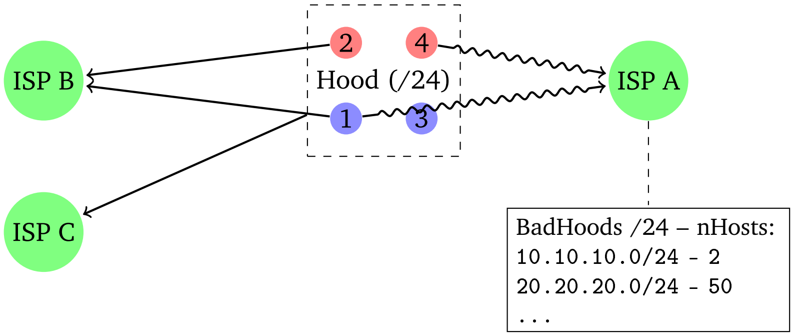


Figure 2.8: Envisioned BadHood-based Application Scenarios

hosts are not blacklisted, they will always be able to maintain their communications. For example, host 2, malicious, is able to spam ISP B.

- *Malicious hosts are blocked*: this is the case when BadHoods help to legitimately filter out spam messages, as in the case of hosts 2 and 4 being blocked at ISP A.

Considering the fact that BadHood-based technologies are not supposed to be the single solution for network filtering and that the filtering occurs on the receiver side, the ethical implications are limited. In addition, BadHood-based technologies also provide values such as security and economic value. However, since this dissertation does not focus on the deployment of BadHood-based filtering technologies, we strongly recommend that the ethical implications when deploying the technology should be considered and evaluated.

Part II

Bad Neighborhoods Characteristics

We are going to die, and that makes us the lucky ones.
Most people are never going to die because they are never
going to be born. The potential people who could have
been here in my place but who will in fact never see the
light of day outnumber the sand grains of Sahara.

Richard Dawkins, 1998
In: Unweaving the Rainbow

CHAPTER 3

Internet Bad Neighborhoods Aggregation[1]

THE Bad Neighborhood concept is based on the assumption that malicious hosts tend to be concentrated in certain networks instead of being evenly distributed over the entire IP address space (as verified in Section 2.7 and in [27, 28, 29, 30, 31]). In Section 2.6, we have shown how to aggregate individual hosts into /24 Bad Neighborhood (in CIDR notation [25]). However, a question one may ask is *how to aggregate* malicious hosts into prefixes other than /24 (e.g., /20, /18, /12, etc.), and what prefixes suit best to express Bad Neighborhoods (BadHoods hereafter), given a certain data set.

To better illustrate this, let's call upon our analogy to bad neighborhoods in the real world. Consider Figure 3.1, in which the x axis represents addresses, whereas the y axis shows how malicious each individual is (values close to 0 mean legitimate hosts – shown as squares – while malicious have higher values for y – shown as circles). If the local Police Department were to release a list of the most dangerous areas, the areas could be represented by employing a *fixed aggregation level* (e.g., only boroughs, shown as dashed rectangles in Figure 3.1, having a fixed size of 4) or *variable aggregation levels* (e.g., blocks, streets, boroughs, represented as dashed ellipses of different sizes in the same figure, having sizes equal to 2, 4, etc.).

This aggregation into BadHoods, however, has to deal with two conflicting requirements: *(i)* the aggregated list should be concise and *(ii)* the aggregation process should minimize the error incurred.

The first requirement – a concise BadHoods list (in the Internet this means prefixes $<= /24$) – allows one, in terms of Internet BadHoods, to reduce memory storage requirements and increases lookup speeds for BadHood-based security software.

[1] This chapter is based on the following publication: Moura, G. C. M., Sadre, R., Sperotto, A., Pras, A.: *Internet Bad Neighborhoods Aggregation*. In: IEEE/IFIP Network Operations and Management Symposium (NOMS 2012), Maui, Hawaii, USA, 16-20 April 2012.

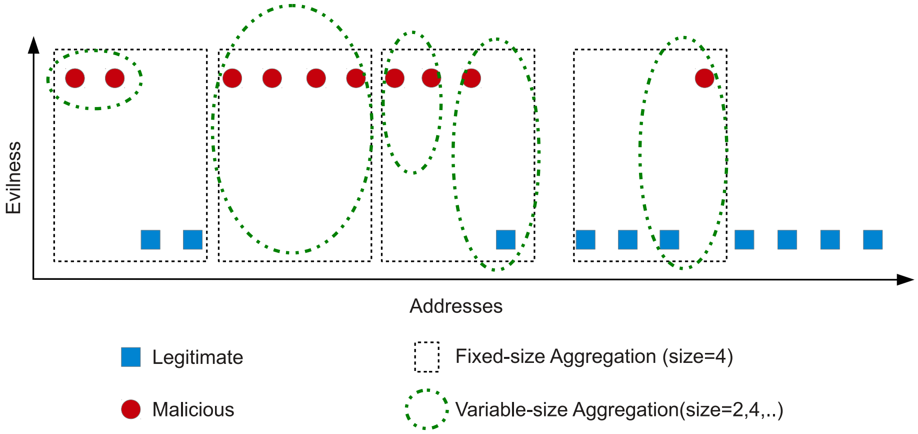

Evilness
Addresses
Legitimate
Malicious
Fixed-size Aggregation (size=4)
Variable-size Aggregation(size=2,4,..)

Figure 3.1: Aggregation into Bad Neighborhoods

The second requirement – minimize aggregation error – derives from how legitimate hosts are mistakenly included in the aggregation process. Consider Figure 3.1: in the case of the left most fixed-size Bad Neighborhood (BadHood): it has a fixed size of four consecutive addresses – in which two of them (squares) were mistakenly included in the process. This is similar to what occur in the real word: not all residents of a bad neighborhood are necessarily malicious; but *some* are, and therefore, the entire neighborhood may be labeled as bad.

In this chapter, we investigate at what aggregation levels Internet BadHoods should be expressed, considering the aggregation requirements aforementioned. Since the aggregation level may depend on the input data, the *goal of this chapter* to present and evaluate *data set independent algorithms* to aggregate malicious hosts into Internet Bad Neighborhoods of various prefixes [/24 – /8].

The contribution of this chapter consists of two BadHood aggregation algorithms. The first one, *fixed-prefix* (dashed rectangles in 3.1), aggregates malicious hosts using the same aggregation prefix, while the *variable-size* algorithm aggregates hosts into different aggregation prefixes (ellipses in 3.1). Both algorithms deal differently with the aggregation requirements aforementioned: in Figure 3.1, the fixed size algorithm generates a list of 4 BadHoods (dashed rectangles), with an aggregation error proportional to 6 legitimate individuals mistakenly included (small squares inside dashed rectangles). The variable-size algorithm, on the other hand, yields 5 BadHoods (ellipses), where it wrongly aggregates only 2 legitimate individuals.

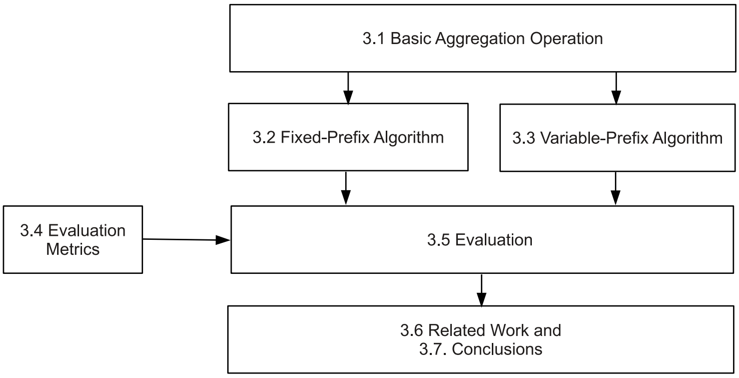

3.1 Basic Aggregation Operation
3.2 Fixed-Prefix Algorithm
3.3 Variable-Prefix Algorithm
3.4 Evaluation Metrics
3.5 Evaluation
3.6 Related Work and
3.7. Conclusions

Figure 3.2: Chapter Structure

The rest of this chapter is divided as shown in Figure 3.2: in Section 3.1, we present the basic principles used as basis for the two aggregation algorithms. Next, in Section 3.2, we present the fixed-prefix aggregation algorithm, in Section 3.3 we presented the variable prefix aggregation algorithm. After that, in Section 3.5 we evaluate those algorithms by using real world data sets, which is done by employing the metrics defined in Section 3.4. Related work is then presented in Section 3.6, and conclusions are discussed in Section 3.7.

3.1 Aggregation Principles

In this section, we describe the basic aggregation operation to aggregate /24 BadHoods into larger BadHoods, which is employed by both algorithms presented in this chapter. We begin by introducing in 3.1.1 the BadHood Score employed in the basic aggregation operation. Then, in Section 3.1.2, we formalize the process of merging two $/n$ BadHoods into one $(/n - 1)$ BadHood.

3.1.1 Bad Neighborhood Score

Given a list of malicious IP addresses, a $/n$ BadHood (in CIDR notation [25]) is a $/n$ netblock B^n with a score $score(B^n)$. We define this score as the number of malicious hosts in the block:

$$score(B^n) = \#\{\text{malicious hosts in block } B^n\} \tag{3.1}$$

#	**/24 netblock**	$score$	P_n
1	10.10.10.0	22	0.086
2	10.10.11.0	21	0.082
3	10.10.12.0	20	0.078
4	10.10.13.0	41	0.160
5	20.20.24.0	130	0.508
6	20.20.25.0	1	0.004
7	30.30.34.0	60	0.234

Table 3.1: Example of /24 BadHoods and their scores

Since /24 is "the minimum prefix routable on the Internet" [78], we use /24 as the starting aggregation level for IP prefixes in the rest of this chapter. Table 3.1 provides a short example of a /24 BadHood list.

The score value leads to an intuitive definition of the "evilness[2]" of a netblock: the higher the score, the higher the probability that a host address from the block is a source of malicious activities. However, the score depends on the size of the block. For the following, it is useful to have a *normalized* measure of BadHood score, which represents the percentage of hosts within a netblock that are malicious. Let B^n be a netblock of size $/n$ with score $score(B^n)$. We define the *normalized score* of B^n as

$$p_n(B^n) = \frac{score(B^n)}{max\_hosts(B^n)}, \tag{3.2}$$

where $max\_hosts(B^n) = 2^{32-n}$ is the maximum number of IP addresses in a $/n$ netblock (neglecting the addresses reserved for broadcast and network identification). P_n can also be interpreted as the *probability* of a random host h in the netblock B be malicious.

3.1.2 Basic Aggregation Operation

Given two $/n$ BadHoods $B_i{}^n$ and $B_j{}^n$, these BadHoods can be aggregated into the $/(n-1)$ BadHood $B_i{}^n \oplus B_j{}^n$ if $B_i{}^n$ and B_j^n have a common address prefix of $n-1$ bits. The aggregated BadHood $B_i{}^n \oplus B_j{}^n$ spans the IP addresses of $B_i{}^n$ and B_j^n. For example, in Table 3.1, blocks #1 and #2 can be aggregated from /24 to /23, while blocks #1 and #7 can not.

[2] By evilness we refer to the potential damage that can be incurred by a netblock, and not by the the intention of the possible unaware users behind zombie computers (see Section 2.8 for more).

#	/23 netblock	$score$	P_n
1	10.10.10.0/23	43	0.084
2	10.10.12.0/23	61	0.119
3	20.20.24.0/23	131	0.256
4	30.30.34.0/23	60	0.117

Table 3.2: /Fixed Prefix Aggregation (1^{st} iteration)

Consequently, the normalized score of the aggregated BadHood ${B_i}^n \oplus {B_j}^n$ is as follows:

$$p_{n-1}({B_i}^n \oplus {B_j}^n) = \frac{score({B_i}^n) + score({B_j}^n)}{max\_hosts({B_i}^n \oplus {B_j}^n)} = \frac{1}{2}\left(p_n({B_i}^n) + p_n({B_j}^n)\right). \quad (3.3)$$

This basic operation provides the basis for both fixed prefix and variable prefix aggregation algorithms, described in Sections 3.2 and 3.3, respectively.

3.2 Fixed Prefix Aggregation Algorithm

The *fixed prefix aggregation* algorithm iteratively aggregates bad neighborhoods into larger netblocks. In the first iteration, all /24 BadHoods are aggregated into /23 BadHoods according to the aggregation operation described in Section 3.1.2 (provided that ${B_i}^n$ and B_j^n have a common address prefix of $n-1$ bits). For example, the /24 BadHoods provided in Table 3.1 will be aggregated into the /23 BadHoods shown in Table 3.2. In the next iteration the /23 BadHoods are aggregated into /22 ones, and so on.

Algorithm 1 presents the pseudocode for this algorithm. The algorithm takes as input the initial list S_{24} of /24 netblocks B_i^{24} with $score(B_i^{24})$ and the largest desired aggregation level m. In each iteration (line 1), the algorithm builds the list S_{n-1} of $/(n-1)$ BadHoods by merging all pairs of $/n$ BadHoods B_i^n, B_j^n (where possible) according to the basic aggregation operation described in Section 3.1.2 (lines 3–5).

It is important to note that, in this algorithm, empty netblocks ($score = 0$) are included if no matching BadHood is found for aggregation (line 3 in Algorithm 1). In our example, the /24 BadHood 30.30.34.0 in Table 3.1 is aggregated with the zero-score netblock 30.30.35.0.

The fixed prefix aggregation algorithm effectively reduces the number of BadHoods in each iteration because it progressively builds larger netblocks re-

Algorithm 1 Fixed prefix aggregation

```
Require: $S_{24} = \{(B_i^{24}, score(B_i^{24})), i = 1 \ldots num\_entries\}$
Require: largest aggregation level $m$
Ensure: $S_m$
for $n = 24 \rightarrow m + 1$ do
    $S_{n-1} := \emptyset$
    for all $B_i^n, B_j^n \in S_n, i \neq j$ with common $n-1$ prefix (use an empty netblock if no matching $B_j^n$ found) do
        $S_{n-1} := S_{n-1} \cup \{(B_i^n \oplus B_j^n, score(B_i^n \oplus B_j^n))\}$
    end for
end for
```

gardless their scores. However, this simple approach also exhibits some drawbacks. First, aggregating two BadHoods with normalized scores a and b will result in a BadHood with an normalized score of $\frac{a+b}{2}$. The larger the difference between a and b, the more information about the behavior of the individual /24 BadHoods in the aggregated BadHood is lost. Secondly, enlarging the BadHoods can have the side effect of including also netblocks that were not initially flagged as malicious, as already illustrated in our example by the netblock 30.30.35.0. This effect aggravates with each iteration.

3.3 Variable Prefix Aggregation Algorithm

Differently from the fixed prefix aggregation algorithm, the variable prefix algorithm does not apply the same degree of aggregation to all BadHoods. Instead, the main idea is to merge two BadHoods only they are *sufficiently similar*, which is verified by a a *merging condition*. Otherwise, aggregation does not occur.

Algorithm 2 presents the pseudocode for variable prefix aggregation algorithm. As in the previous algorithm, the algorithm takes as input the initial list S_{24} of /24 netblocks B_i^{24} with $score(B_i^{24})$ and the largest desired aggregation level m. Then, for each aggregation level n (line 2), the algorithm merges all $/n$ BadHoods B_i^n, B_j^n which would form a valid aggregated BadHood according to the basic aggregation operation (see Section 3.1.2) that satisfy the merging condition (line 3). BadHoods that do not fulfill those conditions are not aggregated and therefore not considered further for aggregation in this or the next iterations. The merging condition is defined as:

$$merge(B_i^n, B_j^n) = p_{n-1}(B_i^n \oplus B_j^n) \geq \beta \cdot \max(p_n(B_i^n), p_n(B_j^n)). \qquad (3.4)$$

Algorithm 2 Variable prefix aggregation

```
Require: $S_{24} = \{(B_i^{24}, score(B_i^{24})), i = 1 \ldots num\_entries\}$
Require: largest aggregation level $m$
Require: merging condition parameter $\beta$
Ensure: $S$
1: $S := S_{24}$
2: for $n = 24 \rightarrow m+1$ do
3:     for all $B_i^n, B_j^n \in S, i \neq j$ with common $n-1$ prefix $\wedge$ $merge(\ B_i^n, B_j^n)$ do
4:         $S := S \setminus \{(B_i^n, score(B_i^n)), (B_j^n, score(B_j^n))\} \cup \{(B_i^n \oplus B_j^n, score(B_i^n \oplus B_j^n))\}$
5:     end for
6: end for
```

The condition is such that we allow a merge only if the resulting normalized score $p_{n-1}(B_i^n \oplus B_j^n)$ is at least equal to a fraction β of the most malicious of the blocks to be merged. The parameter β prevents therefore the aggregation algorithm from merging dissimilar BadHoods. This value can be tuned according to the scenario and application. β ranges between 0.5 and 1: smaller values make the aggregation less strict, thus allowing more BadHoods to be merged. Values close to 1 will instead lead to a less permissive aggregation algorithm.

Finally, at line 4, the algorithm progressively builds the new BadHood set by removing BadHoods and replacing them with the merged one. Note that, in contrast to the fixed prefix aggregation algorithm, now BadHoods of different sizes coexist in the result set S.

In order to illustrate the algorithm, we apply it to the example given in Table 3.1. For $\beta = 0.8$, we obtain after one iteration the BadHoods shown in Table 3.3. Blocks #1 and #2 are merged, because $p_{24}(B_1) = \frac{22}{254}$, $p_{24}(B_2) = \frac{21}{254}$, and $p_{23}(B_1 \oplus B_2) = \frac{43}{510}$, so $p(B_1 \oplus B_2) > 0.8 \cdot max(\cdot) \Rightarrow 0.086 > 0.069$. The other blocks, on the other hand, do not match the condition, so they are not aggregated. After the first iteration, the list contains both /23 and /24 entries. In the next iterations, no further aggregation occurs, and the final result contains entries using mixed prefixes (/23 and /24).

Comparing the results of both algorithms (Tables 3.2 and 3.3) for the first iteration, the output of the variable prefix algorithm has more entries. However, the blocks aggregated by the variable prefix algorithm have been matched against a stricter merging criteria. In the next section we evaluate both algorithms using real world data.

Finally, we have implemented both algorithms in a Java prototype. We have observed runtimes of less than 10 seconds even for large input files (1M BadHoods).

#	**/23 netblock**	$score$	P_n
1	10.10.10.0/23	43	0.084
2	10.10.12.0/24	20	0.078
3	10.10.13.0/24	41	0.160
4	20.20.24.0/24	130	0.508
5	20.20.25.0/24	1	0.004
6	30.30.34.0/24	60	0.234

Table 3.3: BadHoods resulting from variable prefix aggregation

3.4 Evaluation Metrics

In this section we introduce the metrics used to evaluate the aggregation algorithms aforementioned. As previously described, the aggregation algorithms have to deal with two conflicting requirements – generate a concise BadHood blacklist and minimize the aggregation error.

The evaluation metrics are derived from these requirements. The first one is the reduction achieved by the algorithms in terms of number of entries in the initial BadHood input list. We measure it as the difference between the number of entries (also called "lines" in the following) in the initial /24 BadHood list and in the resulting list generated by both fixed prefix and variable prefix algorithms.

The second metric derives from the aggregation error. In the following, we consider an (hypothetical) application, such as a Spam filter, that relies on the aggregated lists. Let be $\{X^{24}, Y^{24}, \ldots\}$ a set of /24 BadHoods with normalized scores $\{p_{24}(X^{24}), p_{24}(Y^{24}), \ldots\}$. We can interpret $p_{24}(X^{24})$ as the probability that a particular IP address in block X^{24} be a source of malicious activities. After we have aggregated the /24 BadHoods to a $/n$ BadHood B_i^n, with $n < 24$, only the normalized score $p_n(B_i^n)$ of the aggregated BadHood is available to the application. The "evilness" of a particular IP address in X^n can now only be estimated by $p_n(B_i^n)$ (Equation (3.2)). Consequently, we define the error $err(X^{24})$ introduced by the aggregation for the BadHood X^{24} as

$$err(X^{24}) = p_n(B_i^n) - p_{24}(X^{24}). \tag{3.5}$$

A positive (negative) error indicates that the application would overestimate (underestimate) the evilness of X^{24} after the aggregation. To assess the global error for an entire blacklist, we sum up the absolute errors for each /24 BadHood X^{24}:

$$Err_{abs} = \sum |err(X^{24})| \tag{3.6}$$

Alternatively, we sum the squares of the individual errors:

$$Err_{square} = \sum err(X^{24})^2 \tag{3.7}$$

The difference between both errors is that Err_{square} places greater weight on individual errors that are further apart – which ultimately emphasizes the differences between the error after aggregation ($p_n(B_i^n)$) and before the aggregation ($p_{24}(X^{24})$).

To illustrate how they are calculated, consider block B_1 in Table 3.2. Before being aggregated into a /23 BadHood, the normalized score $p(B_1)$ was $\frac{22}{256}$ (Table 3.1). After that, the same netblock gets the mean value $\frac{43}{512}$. The error $err(B_1) = \frac{22}{256} - \frac{43}{512} = -0.0019$. After calculating the individual errors, the global errors, as defined in (3.6) and (3.7), can be obtained.

The interpretation of the error values depends on the application and other definitions of the global errors are possible. For example, for an intrusion detection system, large errors may cause increased False Positive or False Negative rates. In such a scenario, calculating the global errors separately for positive and negative $err(X^{24})$ values could be of interest. In order to be independent of a particular application and suitable for different scenarios, we have chosen the rather flexible definitions in (3.6) and (3.7).

3.5 Evaluation

In this section we evaluate and analyze the impact of the aggregation algorithms. We first present the data sets that we use for our evaluation in Section 3.5.1. The performance of both algorithms is compared for the largest of our datasets, the Composite Blocking List (CBL; see below), in Section 3.5.2. In Section 3.5.3, we study the impact of the merging parameter β on the performance of the variable prefix aggregation algorithm. Finally, we compare the results for different datasets in Section 3.5.4.

3.5.1 BadHood Input Blacklists

We evaluate our aggregation algorithms on the real case of a Spam blacklist. The considered data set is the Composite Blocking List (CBL) [60] – an online Spam DNS blacklist. CBL maintains four large spamtrap infrastructures from where the source IP addresses of spammers are harvested. We have obtained the list for the April 28th, 2010. On this day, CBL listed 8,177,138 /32 IP addresses, which result in an initial blacklist of 960,167 /24 BadHoods. As described in Section 3.1.1, we start with /24 since this is the minimum prefix "routable" on the Internet.

In addition to the above list, we use the following datasets for our experiments in Section 3.5.4:

- Passive Spam Block List (PSBL) [75], obtained on April 28th, 2010: the list consists of more than 2.8M /32 distinct IP addresses;
- Passive Spam Block List (PSBL) [75], obtained on October 24th, 2011: the list consists of more than 283K /32 distinct IP addresses;
- Mail server logs from Provider A: Provider A is a major hosting provider in the Netherlands. We have obtained the IP addresses of spammers on April 28th, 2010. For this day, 256K distinct /32 IP addresses were observed.

3.5.2 Performance of the Aggregation Algorithms

In this section, we present the results of the aggregation algorithms applied to the CBL data set. We first analyze the gain on the size of the blacklist as result of the aggregation. Then, we discuss the impact of the aggregation algorithms on the global errors.

Blacklist size

Figures 3.3(a) shows the number of entries (in thousands) in the resulting blacklists as function of the aggregation level m for the fixed prefix aggregation algorithm, whereas Figure 3.4(a) shows it for the variable prefix aggregation algorithm. For the later, we have chosen a rather moderate merging parameter of $\beta = 0.8$. The influence of the parameter is discussed in Section 3.5.3.

As expected, both algorithms are able to reduce the number of entries of the initial input blacklist. If compared, however, we can see that their performance in terms of the number of entries is very dissimilar. The fixed prefix algorithm progressively aggregates listed BadHoods into larger netblocks, regardless their scores and normalized scores. As a result, the number of entries decreases with the aggregation level, i.e., with increasing block sizes.

The variable prefix algorithm, on the other hand, only aggregates blocks that meet the merging condition specified in (3.4). As a result, once no more candidate blocks satisfy the condition, the number of entries in the blacklist stabilizes. As can be seen in Figure 3.4(a) we observe that there is no more aggregation after the /15 prefix. Indeed, most of the aggregation is achieved when moving from level /24 to /23. Therefore, the variable prefix aggregation can be seen as a “less aggressive” approach than the fixed prefix one. While the fixed prefix algorithm reduces the blacklist, from the original 960,167 /24 BadHoods to 162 entries for /8, the variable prefix algorithm stabilizes at around 711k entries. However, the /8 fix prefix aggregation level is very aggressive and is expected to generate large aggregation errors. We show next aggregation error evaluation.

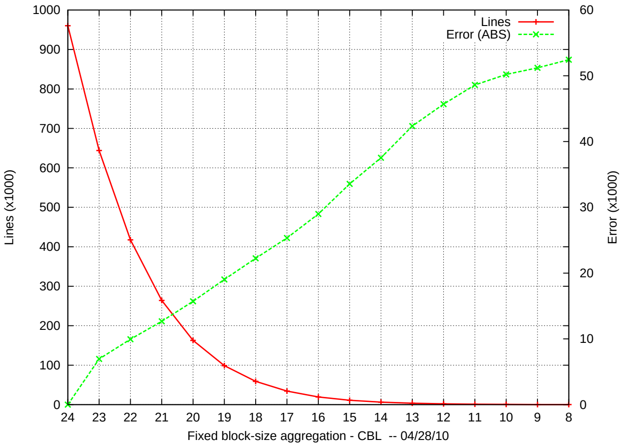
Error

Figures 3.3(a) and 3.4(a) also show the global absolute errors (see (3.6)) for the two algorithms as function of the aggregation level. The results for the global squared errors (see(3.7)) are shown in Figures 3.3(b) and 3.4(b).

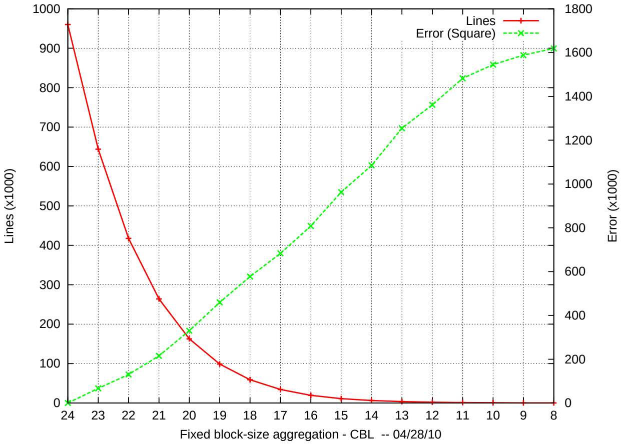
First and foremost, we observe that the fixed prefix aggregation algorithm results in much larger errors than the variable prefix algorithm. This is an expected result since the former aggregates blocks regardless of their normalized score. Therefore, many dissimilar blocks (in regard to their scores in (3.1)) are aggregated, leading to large differences between the normalized scores of the individual /24 blocks and the normalized score of the aggregated block.

In the case of the fixed prefix aggregation algorithm, we also observe that the errors almost linearly increase with the (decreasing) aggregation level, although the achieved reduction of the number of lines is not linear at all. This is due to the fact that the algorithm also considers empty blocks, i.e., blocks with a score of 0. Aggregating an empty block with a non-empty block does not reduce the number of lines, but increases the error. This effect aggravates with the aggregation level (see Section 3.2). In fact, up to around aggregation level /18, substantial reductions in the number of lines are achieved by the algorithm. After this point, as we can see from the constant error increase, the aggregation of two netblocks becomes more expensive in terms of errors. At aggregation level /8, the absolute and square errors are respectively 2.36 and 2.8 time larger than at the /18 level, as expected. This leads to the conclusion that the aggregation to larger netblocks (small prefixes) has a huge impact on the correctness of the final blacklist and, consequently, our analysis proves that the fixed-prefix aggregation algorithm should be stopped before prefixes larger than /8 (e.g„ /20 for this case, depending on the input data).

In contrast, the error curves of the variable prefix aggregation algorithm mostly mirror the achieved reduction of lines. Both the number of lines and the global errors significantly change up to around level /18. After this point,

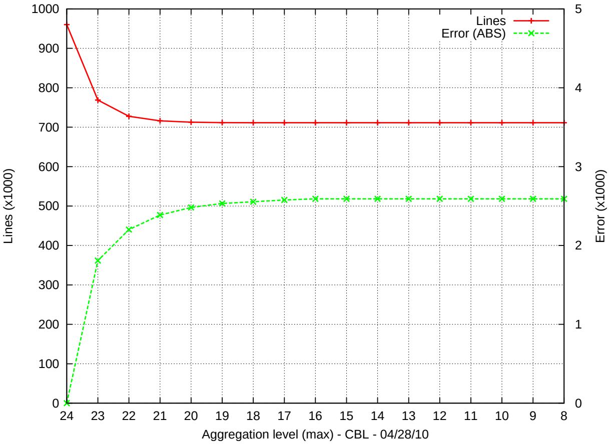


(a) Absolute error

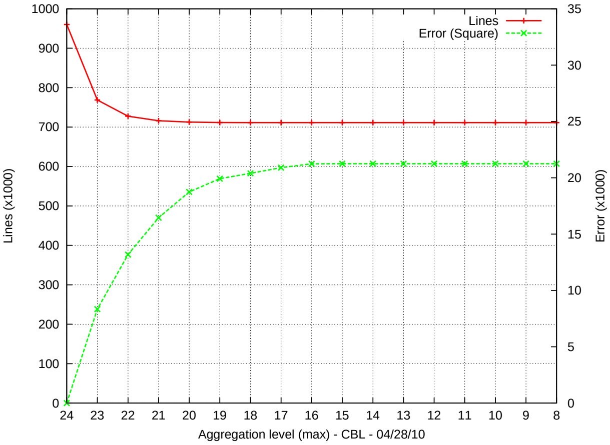


(b) Squared error

Figure 3.3: Fixed prefix aggregation algorithm - CBL - 04/28/10



(a) Absolute error for β=0.8



(b) Squared error for β=0.8

Figure 3.4: Variable prefix aggregation algorithm - CBL - 04/28/10

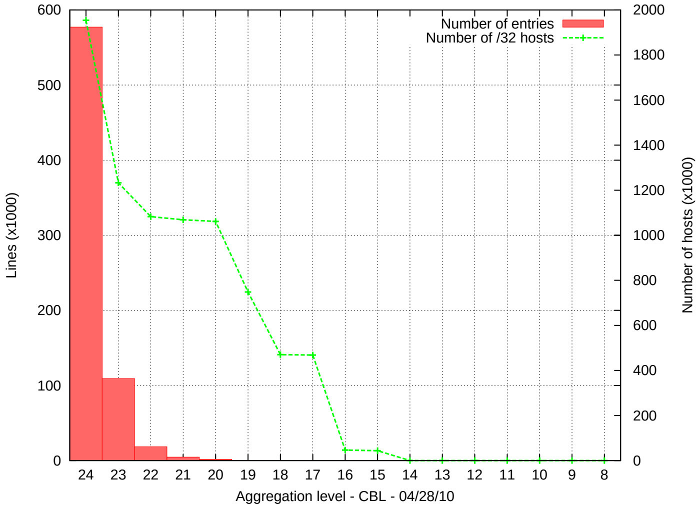


Figure 3.5: Variable Prefix Aggregation for $\beta = 0.8$

there are nearly no more aggregations and the error stabilizes. As expected, the squared error (see Figure 3.4(b)) is more sensitive to changes than the absolute error (see Figure 3.4(a)).

Distribution of malicious hosts

As already stated, the variable prefix aggregation algorithm achieves most of the reduction when moving from level /24 to /23. After that, only a small portion of the BadHoods fulfill the merging condition and can be aggregated further. The bar chart in Figure 3.5 (left y axis) shows the resulting distribution of the BadHood sizes for aggregation level $m = 8$ (i.e., /8) and $\beta = 0.8$. As can be seen, a large portion (around 580k) of the initial 960,167 /24 BadHoods are not aggregated at all and stay at level /24. Around 109k entries are aggregated into /23 BadHoods and only a few entries are aggregated into /22 or higher.

In the same figure, we also show the distribution of the number of /32 host addresses (right y axis). Remember, that the original data set contains 8,177,138 host addresses (see Section 3.5.1). According to the figure, around 2 million host addresses stay in /24 BadHoods after aggregation, but most of the bad hosts can now be found in /23 through /17 BadHoods.

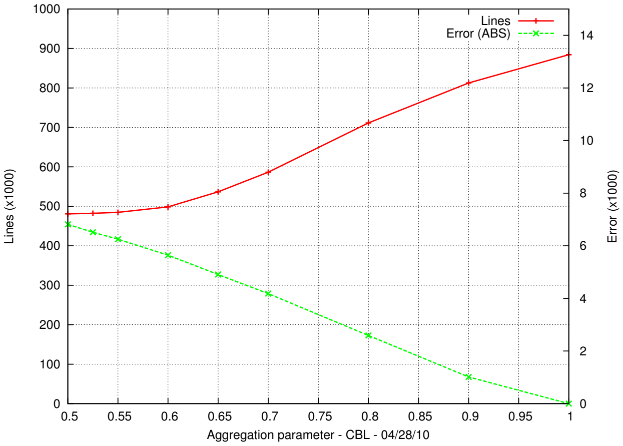
Surprisingly, the distribution of the malicious hosts does not match the distribution of the BadHood sizes, even when considering that a $/(n-1)$ BadHood is twice as large as a $/n$ one. For example, we have observed an average of 3.38 malicious hosts per /24 BadHood in Figure 3.5 (dividing the right y axis value by the left y value). Intuitively, one could expect the average for /23 to be twice the average for /24 – which is 6.76. However, the average is, in fact, 11.29, a value is 67% above what was expected. This can be explained by the nature of BadHoods. Since a /24 netblock with a high score indicates a badly managed subnetwork, it is natural to expect that similar netblocks can be found in its own neighborhood. Such netblocks are, then, preferred by the merging condition in (3.4) and, hence, are more likely aggregated. These results illustrate the benefits of the aggregation.

3.5.3 The Impact of β on the Aggregation

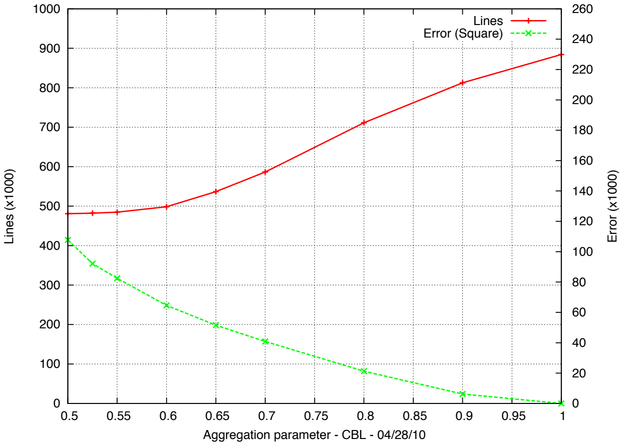
In the following experiments we study the impact of the merging parameter β on the performance of the variable prefix aggregation algorithm. Intuitively, if β is too permissive (β close to 0.5), it might results in a blacklist in which most of the blocks are aggregated, while a more strict value for β (β close to 1) would aggregate very few blocks. However, at the same time, a permissive aggregation algorithm would also result in a larger aggregation error, while a algorithm aggregating only very similar blocks would result in a small error.

Figure 3.6 shows, for varying values of β, the number of entries in the blacklist output by the aggregation algorithm, as well as the corresponding global errors (absolute and squared). For $\beta = 0.5$, the resulting BadHood list contains around 470k entries. For increasing values of β, the blacklist becomes progressively larger, while the errors decrease almost linearly. Finally, for $\beta = 1$, the final blacklist has almost the same number of lines as the original non-aggregated one, since the algorithm only aggregates valid netblocks with exactly the same normalized score. Therefore, no error is observed for $\beta = 1$.

The figure clearly shows that there is a trade-off between having a short and efficient blacklist and having a small merging error. Therefore an appropriate value of β should be chosen case by case, and, accordingly to the scenario, the security manager should decide to favor a fast blacklisting process, or a precise one.



(a) Absolute Error



(b) Square Error

Figure 3.6: The impact of β on the variable prefix aggregation

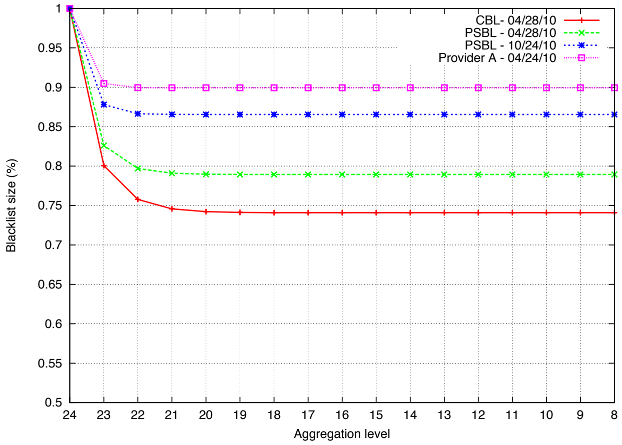


Figure 3.7: Variable prefix aggregation algorithm applied to different data sets for β=0.8

3.5.4 The Impact of Different Blacklists on the Aggregation

We now discuss the results provided by the variable prefix aggregation algorithm for the other data sets presented in Section 3.5.1. We omit the result of the fixed prefix aggregation algorithm due to its large errors.

In Figure 3.7, we show the number of lines of the result blacklists relative to the original sizes of the /24 data sets, as computed by the variable prefix aggregation for varying aggregation level and $\beta = 0.8$. We observe that our aggregation algorithm is able to reduce the blacklist size for each of the considered data sets. For $\beta = 0.8$, the data sources experience a reduction on the number of entries from 10% for the “Provider A” data set to 26% for the CBL.

A second observation is that the two largest lists (CBL and PSBL) from April 28th, clearly benefit more from the aggregation than the smaller lists. This is expected because the BadHoods in the smaller lists are more sparsely distributed over the Internet address space and, hence, are harder to aggregate. In addition, the “Provider A” data set experiences the smallest reduction of all four traces.

3.6 Related Work

IP aggregation has been previously investigated by the IP routing community. Back in 1993, only classfull addresses were used (former classes A, B, and C). This addressing scheme was causing several problems – including exhaustion of the Class B network address space and "growth of routing tables in Internet routers beyond the ability of current software, hardware, and people to effectively manage" [25].

Therefore, in 1993 the IETF introduced the Classless Inter-domain Routing (CIDR) addressing [25], and the prefix notation used here. This new addressing scheme allowed blocks to be allocated under prefixes different than the ones specified by classes A, B, and C. That allowed route entries with the same prefix to be aggregated in what is called supernets [79]. By aggregating them, the number of entries in routing tables of BGP [80] routers was reduced, and that decreased the requirements for storing routing information on routers and the overhead when matching routes. Current BGP routers have typically 372k entries in their routing tables [81], a small value compared to current /24 BadHoods blacklists, such as CBL (1M+ entries) .

3.7 Conclusions

In this chapter, we investigate the aggregation of individual hosts on the Internet into IP prefixes expressed in CIDR notation. We have proposed two algorithms and evaluated them according to conflicting requirements: – create a concise BadHood blacklist and to minimize the aggregation error.

The first aggregation algorithm – *fixed prefix* – has proven to be very efficient when it comes to reducing the number of lines in BadHood blacklists. By aggregating the entries to /18, we observe a reduction of 93.85% on the /24 original size. However, the error incurred by this algorithm is high. For the evaluated data, the results have shown that the aggregation further than /18 corresponds to a reduction of very few entries at the expense of a large error. The implications of a large error is that a large number of hosts end up being label as bad (by taking part of the aggregated BadHood) even though no malicious activities have been observed from such hosts.

The second aggregation algorithm – *variable prefix* – has been designed with the goal to aggregate BadHoods if they are sufficiently similar – defined by the merging condition. In this sense, this algorithm has shown in our evaluation that it is able to reduce the aggregation error. However, the final blacklists it generates is less concise than the one generated by the fixed prefix algorithm. The aggregation parameter β was introduced to allow one to fine tune the trade off between a concise blacklist and aggregation error. We have shown that by having $\beta = 0.8$, we reduced the number of entries by 10% to 26%, depending on the blacklists source.

The answer to the question “what is the best algorithm” and “at which aggregation level both algorithms should stop” depends on evaluated scenario, which includes the application (e-mail, DDoS, etc.) and the infrastructure in question. Therefore, we recommend network operators to carry out an evaluation using both algorithms before deploying BadHood based security, considering the value provided by each algorithm regarding the conflicting aggregation requirements.

In the rest of the dissertation, whenever we employ IP prefix aggregation, we do not aggregate BadHoods any further than /24 in order to minimize the impact that aggregation errors might incur in the results.

Countless creatures for millions of years have had active minds happening in their brains, but only after those brains developed a protagonist capable of bearing witness did consciousness begin, in the strict sense, and only after those brains developed language did it become widely known that minds did exist.

António Damásio, 2011
In: The Self comes to Mind

CHAPTER 4

Internet Bad Neighborhoods Location

So far in this dissertation we have focused on IP addresses-based Bad Neighborhoods. However, as discussed in Section 2.1, we assume that the existence of Internet Bad Neighborhoods is due to *three possible reasons*: (i) that some Internet Service Providers (ISPs) *neglect* malicious activities in their networks, (ii) that malware is more likely to spread on the networks of such ISPs, and (iii) that non-technical local factors may play a role, such as legislation in place, piracy software levels, economics, and education level in a country.

The goal of this chapter is to address these assumptions. We investigate them by assessing the total number and the ratio of malicious hosts found in ISPs and individual countries – that is, by aggregating malicious hosts into Bad Neighborhoods according to these aggregation criteria.

The motivation to carry out this research is to evaluate our assumptions and, ultimately, to provide network administrators with concise information to better protect their networks. E.g, this can be used to filter traffic not only based on IP addresses, but also on the ISP and/or their geographical origin.

Therefore, in this chapter we address the following research questions:

- Research Question 4.1 (RQ. 4.1): *How are malicious hosts distributed over ISPs?*

- Research Question 4.2 (RQ. 4.2): *How are malicious hosts distributed over geographical areas (countries and cities)?*

RQ 4.1 focuses on evaluating the first two assumptions for the existence of Internet Bad Neighborhoods – that some ISPs neglect malicious traffic and *malware* propagation in their networks. These assumptions should hold in case we find ISPs having a significant concentration of malicious hosts.

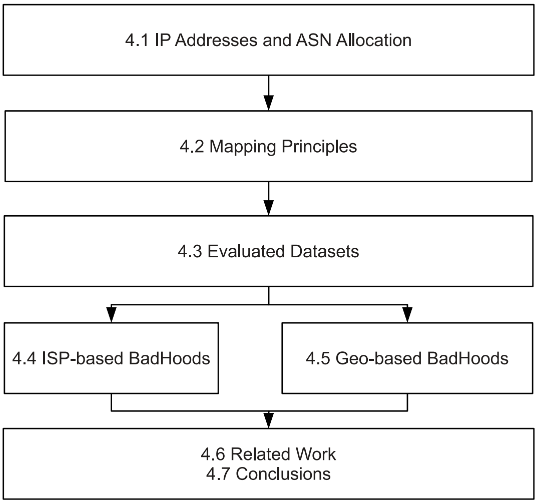


Figure 4.1: Chapter Structure

RQ 4.2, on the other hand, covers the third assumption for Internet Bad Neighborhoods (BadHoods): that non-technical local factors play a role. A concentrated distribution of malicious hosts in a limited number of countries would support our assumption.

To answer the research questions, we have obtained real-world data sets for two applications (spam and *phishing*) for a period of one week, from July 19th to 25th, 2012. After that, for each data set, we have extracted the /32 IP addresses of the malicious sources. To answer RQ 4.1 , we have aggregated (and ranked) malicious hosts into Autonomous Systems (AS) [37] and organizations, while to answer RQ 4.2 we have aggregated (and ranked) the IP addresses into countries and cities.

The rest of this chapter is divided following the structure presented in Figure 4.1: first, we review in Section 4.1 how IP addresses and ASes are allocated on the Internet. Then, in Section 4.2, we show how to map individual /32 IP addresses into ISPs and geographical location, while in Section 4.3 we cover the datasets employed. After that, RQ 4.1 is covered in Section 4.4, while RQ 4.2 is addressed in Section 4.5. Section 4.6 discusses the related work and Section 4.7 presents the conclusions.

4.1 IP Addresses and ASes Allocation

In order to understand how it is possible to map individual IP addresses into ISPs and geographical location, it is necessary to understand how IP addresses and ASes are allocated on the Internet.

The best practices for IP address allocation are covered in the RFC 2050 [82], which we briefly review in this section. The entity responsible globally for this task is the Internet Assigned Numbers Authority (IANA) [83], which is a department of the non-profit private organization Internet Corporation for Assigned Names and Numbers (ICANN) [84], located in the United States.

IANA performs its tasks by *delegation*: it allocates entire /8 prefixes (e.g., 120.0.0.0/8 in CIDR notation [25]) to the Regional Internet Registries (RIRs). The IPv4 prefixes assigned by IANA to the RIRs can be found at [67], while the assigned IPv6 global unicast prefixes can be found at [85]. Currently, there are five RIRs, divided according to geographical regions:

- African Network Information Centre (AfriNIC) [86]: RIR for Africa.
- American Registry for Internet Numbers (ARIN) [87]: RIR for the United States, Canada, parts of Caribbean region, and Antarctica.
- Asia-Pacific Network Information Centre (APNIC) [88]: RIR for Asia, Australia, New Zealand, and neighboring countries.
- Latin America and Caribbean Network Information Centre (LACNIC) [89]: RIR for Latin America and parts of Caribbean region.
- *Réseaux IP Européens* Network Coordination Centre (RIPE NCC) [90]: RIR for Europe, Russia, the Middle East, and Central Asia.

Figure 4.2 shows a “snapshot” of the the IPv4 allocation map in 2006. In this figure, each of the 256 numbered blocks represents one /8 netlock (CIDR notation), arranged according the Hilbert curve [91], while green blocks were unallocated ones.

After obtaining these prefixes, each RIR re-allocates IP address ranges to its customers, typically ISPs and other organizations. The allocation information is kept in a public database, which is made available through the `whois` [92] software. Listing 1 shows a partial output of the `whois` command, issued to query ARIN for the IP address “`208.80.152.201`”. As we can observe, the prefix `208.80.152.0/22` is allocated to WIKIMEDIA (`NetName`).

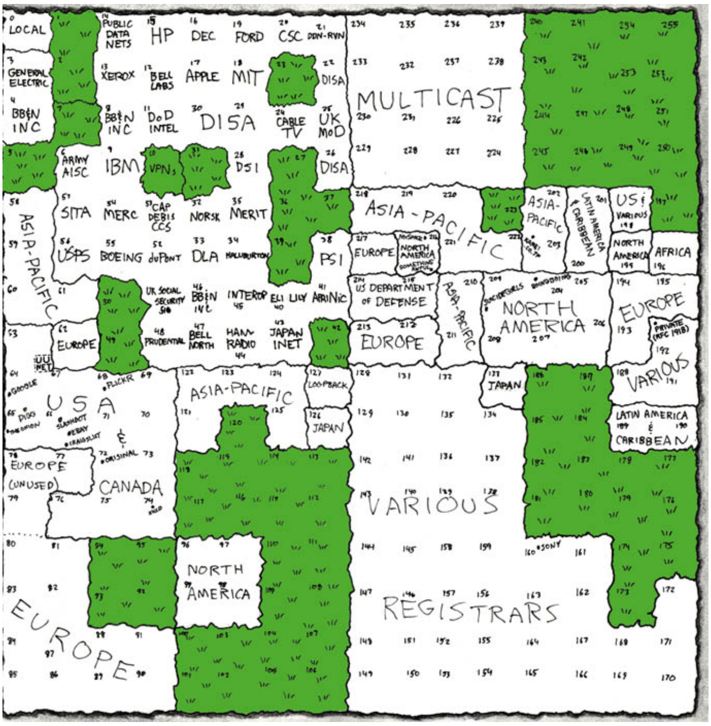


Figure 4.2: IPv4 Allocation Map (2006) - Source: xkcd[1],[2]

The allocation of Autonomous System numbers (ASN) is defined in RFC 1930 [37]. According to the same RFC, an autonomous system is "a connected group of one or more IP prefixes run by one or more network operators which has a single and clearly defined routing policy" [37]. Each AS is identified by an unique number (AS number) assigned by the RIRs that is employed in the core routing decisions on the Internet, using the Border Gateway Protocol (BGP) protocol [80].

In relation to ASes, IANA also allocates Autonomous System Numbers (ASNs) to the regional RIRs [93], which, in turn, are allocated them to their customers, in a similar way as IP prefixes are allocated. For example, in Listing 1, we

[1] http://xkcd.com/195/

[2] http://blog.icann.org/2007/05/mapping-the-internet-one-node-at-a-time/

Listing 1 Partial `whois` output for IP “`208.80.152.201`”

```
giovane@dennett:~/whois 208.80.152.201
NetRange:       208.80.152.0 - 208.80.155.255
CIDR:           208.80.152.0/22
OriginAS:       AS14907
NetName:        WIKIMEDIA
NetHandle:      NET-208-80-152-0-1
Parent:         NET-208-0-0-0-0
NetType:        Direct Assignment
Comment:        http://www.wikimediafoundation.org
RegDate:        2007-07-23
Updated:        2012-03-02
Ref:            http://whois.arin.net/rest/net/NET-208-80-152-0-1

OrgName:        Wikimedia Foundation Inc.
OrgId:          WIKIM
Address:        149 New Montgomery Street
Address:        3rd Floor
City:           San Francisco
StateProv:      CA
PostalCode:     94105
Country:        US
```

can observe that the IP address `208.80.152.201` belongs to the network prefix `208.80.152.0/22` (CIDR field) and its Origin AS is AS14907, which is the Autonomous System Number (ASN) of the Wikimedia Foundation[3].

4.2 Mapping Principles

In this section we show the mapping principles of /32 IP addresses to ISPs in Section 4.2.1 and into geographical information in Section 4.2.2.

4.2.1 Mapping IP addresses and ISPs (ASN)

In order to map malicious IP addresses into ISPs, we employ the ASN associated to the IP address in question. By definition, an ISP is *always* a transit AS – that is, an AS that provides connectivity between various networks. In addition, the ASN is a *unique number* and by using ASN to identify ISPs, one can easily filter traffic in network firewalls/IDS.

[3] `http://bgp.he.net/AS14907`

However, as defined in Section 5 of RFC 1930[37], not every AS is an ISP. In fact, any organization could potentially apply for an ASN, but it is not necessary, since organizations can employ the ISP's ASN even if the organization has been allocated with multiple IP prefixes by the RIR. To illustrate this, consider Figure 4.3. In this figure, the ISP `ISP`, with its own `ASN`, provides connectivity to three organizations (`Org1, Org2, Org3`) to the Internet. `Org1` has been allocated with its own ASN and network prefixes by the RIR, while `Org2` has been allocated only with network prefixes. `Org3`, however, has not been allocated with prefixes or ASN by the RIR.

In this scenario, a malicious IP address x from `Org1` is seen on the Internet as part of `Org1` (`NetName` field in Listing 1) – that is, is listed on the RIR's public database as allocated to Org1. In addition, x is also seen as part of the AS of `Org1`, since `Org1` has its own ASN (`OriginAS` in Listing 1. In this case, the ASN associated to x does not represent the ISP, but `Org1`. An example of IP address that fits into this category is `198.168.230.1`, which is allocated to the Canadian Law firm Stikeman Elliot[4], which, has its own ASN (AS 25805[5]) and employs Cogent Communications (AS 174) as its ISP[6].

Another case is the one in which a malicious IP address y, connected to `Org2`, is seen on the Internet as part of `Org2`, since `Org2` has been allocated with a prefix that contains y by the RIR. However, y is seen on the Internet as part of `ISP`'s ASN, since `Org2` does not have its own ASN and employs the ISP's one. In this case, the ISP would "be blamed" as source of malicious IP that has been, in fact, allocated to another `Org2` by RIR. This is the case for the IP `12.129.19.1`, which is allocated to coffeehouse chain Starbucks[7]. Starbucks employs AS17226, which belongs to AT&T [8], a major ISP in the United States.

The last case is when z is employed by `Org3`, an organization that has not been allocated IP addresses by RIRs. In this case, `Org3` employs IP addresses from its own ISP, and also the ISP ASN. A small business with ADSL connection is a good example of this case. It is important to notice that these organizations are not registered at the local RIR, and therefore are globally seen as part of the ISP.

To cope with the fact that not all ASN are associated to ISPs, we aggregate malicious IP addresses into both ASes-based BadHoods and Organization-based BadHoods. By comparing the results, we can determine whether or not the

[4] `http://www.stikeman.com`

[5] `http://www.ris.ripe.net/dashboard/AS25805`

[6] `http://bgp.he.net/AS25805#_graph4`

[7] This information can be obtained using the command-line `whois` tool.

[8] `http://bgp.he.net/AS17226`

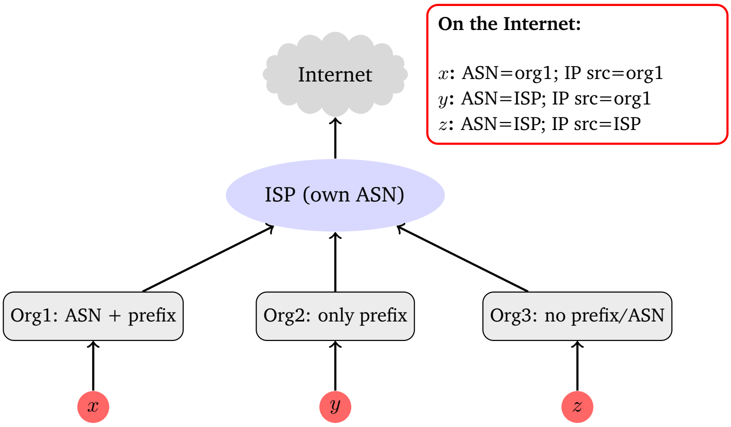


Figure 4.3: IP addresses, ASN, and Routing on the Internet

malicious hosts are more concentrated in ISPs or organizations.

Several sources can be used to map individual IP addresses into ASN. For example, Team Cymru builds a database of IP to ASN mapping based on BGP feeds of more than 50 feeds, updated every four hours [94]. Other providers, like MaxMind, provide as well a publicly available ASN to IP database, named GeoLite Autonomous System Number Database [95]. Due to simplicity of use, we have employed the MaxMind database to resolve IP addresses into ASNs.

Mapping IP addresses into Organizations

Since organizations can have their own ASN, it is necessary to also aggregate malicious hosts in Organization-based BadHoods. As described in RFC 2050 [82], organizations are allowed to ask the RIR for a block of IP addresses. If the organization fulfills the RFC 2050 requirements for having IP addresses assigned to it, the RIR will allocated the block of IP addresses and keep this information publicly available via `whois`. In Listing 1, we can observe that the prefix `208.80.152.0/22` (in the `CIDR` entry) was assigned to Wikimedia Foundation (`OrgName` entry).

Each organization can therefore choose any provider available to connect itself (and the assigned IP addresses) to the Internet. By aggregating IP addresses into organization-based BadHoods, however, we are able to spot which organizations are owners of the most active malicious IP addresses on the Internet, regardless the ISP they choose to connect to the Internet. To resolve an IP address to the organization, one could use the RIR publicly available `whois` databases. However, the `whois` output format varies – that is, some organizations provide more information than others, and the field names may vary. Therefore, in this chapter, we employ the standardized database built by MaxMind, which is based on the RIR `whois` databases, for ease of use [96].

ASes: are they legally responsible for malicious traffic?

Due to the fact that organizations might not need an AS of their own and end up using their ISP's AS (e.g., such as `Org2` and `Org3` in Figure 4.3), one could wonder if an AS should or should not be held legally responsible for malicious traffic observed in the network of the clients (e.g., IPs x and y from `Org1` and `Org2`).

This question is very controversial, as discussed by G. Houston [97]. It deals with the question if an ISP should or should not filter content in their network, which might violate their customers privacy according to the country's legislation. Therefore, we recommend that the answer should be sought in a multidisciplinary way, involving engineers, legislators/law experts, and ethicists (see Section 2.8).

In the past, we have seen some ISPs disconnecting malicious ASN, as in the case of AS26780, back in 2008. The AS26780, belonging to the San Jose based web hosting service provider McColo Corp., was disconnected from the Internet by two of their upstream providers (Global Crossing and Hurricane Electric), due to the large amount of malware and botnets in their networks [45]. The impact of this action was clear: after being disconnected, several reports have shown that the volume of worldwide spam was reduced in 67% [46]. McColo was owned by the Russian national Oleg Nikolaenko (named "King of Spam" by the FBI), who is currently being held in prison in the United States.

Regardless the legal question of whether an ISP should be held responsible for the transit traffic on their networks, in this dissertation we aggregate /32 addresses into AS-based BadHoods in order to identify which ISPs malicious IPs are concentrated.

4.2.2 Mapping IP addresses into Geographical Information

In this section we present the mapping principles of /32 IP addresses geographical location. We first focus on country-based BadHoods, and then extend the concept to city-based BadHoods.

IP Geolocation aims to determine the Internet users' geographical location based on their IP address [98, 99]. It has been used by industries and businesses for many purposes, including targeted advertisement (e.g., a global portal can deliver customized ads according to the user's location), fraud detection (e.g., online stores can check the physical location of a client against its billing address), media licensing (e.g., broadcasters, such as those on Hulu [100], only stream content to IP addresses allocated to certain countries) and even spam filtering [101].

Even though the organization's city and country is provided by the the RIR database via `whois` (as shown in Listing 1), that does not mean that these are the geographical locations of the hosts having these IP addresses: this is actually the physical address of the organization, which may have their computers located in a data center hundreds of miles away from the physical address.

As described by Poese *et al.* [99, 102], there are currently two main paradigms to map IP addresses to geographical location: active IP geolocation and database-driven geolocation. Active IP techniques are typically based on network delay measurements – but they do lack scalability and present a high measurement overhead. Database-driven approach, on the other hand, consists of "a database engine(e.g., SQL/MySQL) containing records for a range of IP addresses, which are called blocks or prefixes" [99, 102].

Since the methodology employed by companies to create geolocation databases is usually not made explicit (therefore, its precision is questionable), Poese *et al.* have investigated the reliability of the databases's geolocation information. They have carried out an experiment comparing the results obtained from the database with the actual results of a large European provider. They found that the databases perform very well when geolocating IP addresses to country-level (96% to 98% success rate, depending on the database), while for city-level, the results were far less precise: for the Maxmind database [103], 60% of the locations presented a 100km location error. However, the experiment carried by the authors have only evaluated data of a single European provider, in a single country (most likely Germany, due the authors' affiliation and the 800km maximum distance limit in the country), which might not reflect the overall accuracy of the database. In addition, for the case of MaxMind, the company provides a web page with the precision of their results [104]. For example, for Germany, the company claims that it is able to resolve 78% of the IP addresses within a range of 40km, while 18% are wrongly resolved and 4% belong to unknown locations.

Even though there are limitations regarding city-level geolocation, still it is widely used by many companies. In this chapter, we have chosen to employ the

Maxmind database. As shown by Poese *et al.* [99, 102], Maxmind [105] is one of the most precise commercially available databases. In addition, simplicity of use and availability were also considered in this choice.

4.3 Evaluated Datasets

In order to answer the research questions raised in this chapter, we have obtained representative real world data sets (blacklists). Since we want to evaluate if the results hold for different malicious activities, we have obtained blacklists for spam and *phishing*. In addition, we have chosen these sources because they have been employed both by research and Internet security communities. The chosen data sets are:

- Spam: for spam blacklists, we have employed the Composite Block List (CBL) , which is "a group of computer security, spam and virus professionals, dedicated to developing and maintaining an anti-spam and anti-virus DNS blacklist (DNSBL) of the highest possible quality and reliability, that large organizations can use with confidence" [60]. CBL maintains their own spam trap infrastructure, and the IP source address of any message reaching their traps is blacklisted.

- Phishing: we have obtained data from Phishtank, which is an open community web site in which anyone can "submit, verify, and track phishing websites" [106]. It provides a blacklists of URLs that contain forged websites. Since we need IP addresses instead of URL to proceed with our analysis, we have obtained this blacklist and resolved all the URLs to IP addresses using Google Public DNS [107].

After choosing the data sets, we have obtained data for the same monitoring period: from July 19th to 25th, 2012. We have then generated a final blacklist containing all /32 unique IP addresses observed in the monitoring period, for both CBL and Phishtank data sets. In the end, we have obtained 9,320,197 unique /32 IP addresses of spam sources, and 3,016 unique /32 IP sources of phishing sites.

Both spam and phishing /32 blacklists were then aggregated into AS-based BadHoods, organization-Based BadHoods, country-based BadHoods, and city-based BadHoods, using the approach described in Section 4.2, with the help of a small program developed in Java.

Aggregation Criteria	Spam BadHoods	Phishing BadHoods
Autonomous Systems	15,078[9]	884[10]
Organization	54,280 [11]	1,304 [12]
Countries	229[13]	92[14]
Cities	25,266 [15]	415 [16]

Table 4.1: Number of BadHoods according to Various Aggregation Criteria

Table 4.1 shows the results of this aggregation for both data sets. We found 15,078 ASes that were observed originating Spam: this represents 35.72% of the currently employed Autonomous Systems (42,201 in total [108]). Moreover, 54,280 organizations were found sending spam – which means, that on average, there are ~ 3.6 organizations per ISP (not excluding the ISP own organization). In addition, 229 out of 250 countries in the database were found sending Spam, from 25,266 different cities.

By comparing both applications, we can observe that there are much more Spam BadHoods than phishing BadHoods. This was expected due to the difference in number of unique /32 IP addresses (approximately 9 million against 3 thousand). For phishing, only 884 ASes were found hosting phishing web sites (roughly 2% of all ASN in use), from 1,304 organizations and 92 countries.

It is also important to point that some entries were not resolved, as showed in the footnotes. The worst case was resolving spam into cities: 17% of all /32 IP addresses could not be resolved. In the next sections, we discuss in details the BadHoods obtained for each aggregation criteria.

4.4 ISP-based Internet BadHoods

In this section, we cover RQ 4.1, by employing the approach described in Section 4.2. We first cover the AS-based BadHoods in Section 4.4.1, and then the organization-based BadHoods in Section 4.4.2. For both cases, we first present

[9] 25,499 out of 9,320,197 IPs were not resolved.

[10] 17 out of 3,016 IPs were not resolved.

[11] 25,866 out of 9,320,197 IPs were not resolved.

[12] 65 out of 3,016 IPs were not resolved.

[13] 227 out of 9,320,197 IPs were not resolved.

[14] 13 out of 3,016 IPs were not resolved.

[15] 1,619,787 out of 9,320,197 IPs were not resolved.

[16] 78 out of 3,016 IPs were not resolved.

the results for spam, and then for the phishing data set.

4.4.1 AS-based Internet BadHoods

AS-based Spam BadHoods

Table 4.2 shows the Top 20 ASes ranked according to the total number of spamming IP addresses (absolute numbers). In this table, *Sources* refers to the number of malicious IP addresses observed, while *IPv4 Orig.* refers to the number of /32 IP addresses the autonomous system announces (including its own prefixes plus the prefixes of its customers that do have their own ASN). We have obtained this information from the Hurricane Electric[17] BGP toolkit web site [108], which generates it based on the BGP tables. We could have also obtained the same information from BGP routing tables from other sources; we have chosen Hurricane Electric since it provides easier (text-based) access to this information.

By employing the number of IPv4 addresses originated per AS, we are able to calculate the ratio of compromised IPs within the AS in question. This is a very important metric, since it shows the percentage of compromised IP addresses within an AS. This is shown in Table 4.2 in the column *Ratio* (`Ratio=100` × `(Sources/IPv4 Originated)`).

Country's %, in the same table, refers to the percentage of malicious IP addresses the particular AS is responsible for in relation to all malicious IPs from its country of origin (Table 4.7).

As can be seen in the table, the first AS in terms of spamming IP addresses is AS9829, which belongs to BSNL (Bharat Sanchar Nigam Limited). BSNL is a state-owned telecommunications company – including telephony (mobile and landline) and broadband Internet, being the largest in India. The second AS in terms of spamming IP addresses is AS45595, which belongs to Pakistan Telecom Company Limited. As BSNL in India, Pakistan Telecom is the major telecommunications company operating in Pakistan.

In Table 4.2, it is also possible to observe the number of IP addresses were originated from the AS in relation to the all malicious IP addresses for the same country (column *Country's %*). This information can be used by government authorities to tackle, within their own country, the ASes having most of spamming IP addresses. For example, 5 out of the 20 listed ASes in Table 4.2 are

[17] Hurricane Electric is a major network backbone operator, being connected to 51 Internet Exchange Points (ranked as # 3 in the world [109]).

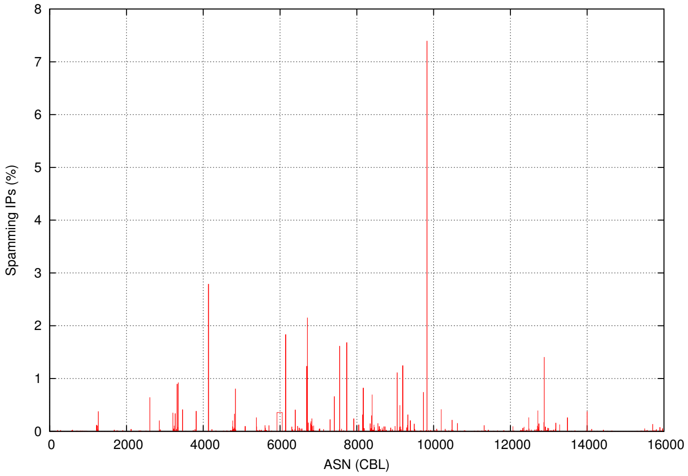


Figure 4.4: Percentage of Spamming IPs per ASN - CBL

responsible for more than 90% of the malicious sources from their own countries. Even though these ASes might not be legally responsible for the activities in their networks (as discussed in Section 4.2), this provides a clear indication of the “health” status of their networks.

Finally, in the *World’s %*, we can observe the percentage of malicious IPs the particular AS is responsible for in relation to all malicious IPs observed. AS9829, from India, was responsible for 7.39% of all the observed malicious IP addresses. This is a very large number for a single ISP, considering that there were 42,201 active ASes at the moment of this analysis. Figure 4.4 shows the percentage of spamming IP addresses for all ASes observed (x axis refers to the AS number (ASN)). As can be noticed by the peaks, few ASes have the highest ratio of spamming IP addresses in relation to the total observed. These results suggests that ASes provide a very good aggregation criteria for identifying Internet BadHoods. This finding supports the two first assumptions behind BadHoods that some ISPs neglect/turn a blind eye to malicious activities in their networks.

It is also important to take into account the percentage of spamming IP addresses in these ASes (*Ratio* column). We have observed 15,078 AS having spamming IP addresses, having, on average, 0.58% of their IP addresses sending spam.

Taking this into account, when analyzing Table 4.2, we can observe, the top 20 Autonomous Systems (ASes) *concentrate almost 50% of all spamming IP addresses* observed in our data sets, from a total of 42,201 active ASes in the moment of our analysis. This finding shows how clear is the existence of BadHoods at the ISP level.

In addition, among the top 20, the AS that has the largest ratio is SaudiNet (27.65% of its IPv4 addresses). Once again, this finding supports our assumption about ISPs that neglect malicious traffic. ChinaNet, on the other hand, is ranked 5th in absolute number of malicious hosts. However, this AS announces more than 110 million IP addresses – and the spamming hosts within this AS represents only 0.23% of the total – which is below the average ratio observed for all the observed AS. However, this should be put into perspective: even though the ratio for Chinanet is smaller, one can not neglect the potential damage it can incur due to its absolute number of malicious hosts in comparison to SaudiNet.

Regardless the ratio exhibited by the top 20 ASes, it is important to emphasize that these ASes have an alarming large number of spamming hosts in their networks, and are truly "spam havens", from which spammers can operate almost freely.

Since ASes can have different "sizes" (that is, the number of IP addresses that they originated), we present in Table 4.3 the top 20 ASes ranked according to the ratio of spamming IP addresses. AS25019, belonging to SaudiNet, from Saudi Arabia, is the only AS that appears in this table and in Table 4.2 (the particular case of Internet censorship in Saudi Arabia will be discussed in Section 4.5.1). In this table, the worst AS is AS37340, in which 62.55% of its IPv4 were found spamming. However, this AS can be seen as a small one, since it only originates 5,632 IPv4 addresses. In fact, in this table, 17 of the 20 ASes can be seen as small ones (IPv4 Originated < 25,000). In such providers, a representative but small (in absolute terms) number of IP addresses were found spamming (< 7,169 per AS).

What we can conclude is that the Bad Neighborhood phenomenon is certainly existing at the AS-level: in only 10% of the 15,078 ASes have a concentration of malicious IP addresses of at least to 2%.

AS-based Phishing BadHoods

Table 4.4 presents the top 20 ASes that were found hosting phishing IP addresses. The number of phishing IP addresses ranges from 22 to 140. All the 20

#	Sources	ASN	AS Name	Country	IPv4 Orig.	Ratio	Country's %	World's %
1	687,107	AS9829	BSNL (Bharat Sanchar Nigam)	IN	4,439,552	15.47	37.46	7.39
2	523,679	AS45595	Pakistan Telecom Company	PK	2,739,968	19.11	92.35	5.63
3	485,944	AS25019	SaudiNet	SA	1,757,440	27.65	60.35	5.23
4	396,885	AS45899	VNPT Corp	VN	2,351,104	16.88	92.73	4.27
5	258,996	AS4134	Chinanet	CN	110,884,096	0.23	93.27	2.79
6	199,679	AS6713	Itissalat Al-MAGHRIB	MA	2,660,864	7.50	99.83	2.15
7	174,056	AS24560	Bharti Airtel, Telemedia	IN	1,578,752	11.02	9.48	1.87
8	171,575	AS17803	BSES TeleCom Limited	IN	1,097,984	15.62	9.35	1.85
9	170,318	AS6147	Telefonica del Peru S.A.A.	PE	1,381,376	12.32	98.58	1.83
10	156,308	AS7738	Telecomunicacoes da Bahia S.A.	BR	4,300,800	3.63	24.22	1.68
11	149,963	AS7552	Vietel Corporation	VN	4,358,144	3.44	22.80	1.61
12	130,283	AS12880	Information Technology Company	IR	1,515,776	8.59	38.35	1.40
13	129,930	AS45609	Bharti Airtel Ltd. AS for GPRS	IN	3,948,288	3.29	7.08	1.40
14	115,776	AS9198	JSC Kazakhtelecom	KZ	1,328,896	8.71	82.39	1.25
15	114,731	AS6697	Rep. Association BELTELECOM	BY	1,082,624	10.59	83.88	1.23
16	113,708	AS17813	Mahanagar Telephone Nigam	IN	1,392,896	8.16	6.19	1.22
17	112,326	AS22927	Telefonica de Argentina	AR	2,998,016	3.74	52.27	1.21
18	109,286	AS45271	ICLNET-AS-AP	IN	762,624	14.33	5.95	1.18
19	105,927	AS27699	TELEFONICA BRASIL SA	BR	3,178,496	3.33	16.41	1.14
20	103,190	AS9050	ROMTELECOM S.A	RO	1,544,960	6.67	73.88	1.11

Table 4.2: Top 20 Spam ASes (ordered according to the absolute number of sources)

#	Ratio (%)	ASN	Sources	IPv4 Originated	AS Name	Country
1	62.55	AS37340	3,523	5,632	SpectraNet Limited	NG
2	55.56	AS50604	1,138	2,048	SC Media SUD SRL	RO
3	43.77	AS31208	1,793	4,096	OJSC MegaFon Network	RU
4	40.81	AS131222	78,992	193,536	Udyog Vihar	IN
5	39.20	AS57704	803	2,048	SpeedClick for ITC	PS
6	37.03	AS56995	1,517	4,096	NetStream Technology	PS
7	35.97	AS36912	2,947	8,192	Orange Cameroun SA	CM
8	35.93	AS43766	552	1,536	MTC KSA Mobile	SA
9	35.35	AS58251	181	512	Dade Pardazi Novin Yaran Tosei	IR
10	34.17	AS50948	700	2,048	Behkoush Rayaneh Afzar Co.	IR
11	33.20	AS197486	1,360	4,096	Uzbektelecom	UZ
12	32.81	AS55740	86,017	262,144	TATA INDICOM CDMA Division	IN
13	32.61	AS45053	668	2,048	Delta Telecom Ltd.	RU
14	31.97	AS48359	2,292	7,168	Hesabgar Pardaz Gharb	IR
15	31.82	AS29497	7,169	22,528	CJSC Kuban-GSM	RU
16	27.66	AS131442	779	2,816	E-14, Rooprajat Nagar	IN
17	**27.65**	**AS25019**	**485,944**	**1,757,440**	**SaudiNet**	**SA**
18	27.34	AS55734	280	1,024	001 IT Complex	IN
19	26.46	AS42601	542	2,048	Gostaresh Ertebatat Mabna	IN
20	26.37	AS45774	3,511	13,312	Chandra Net Pvt. Limited	IN
All	**0.58**	–	**9,294,353**	–	–	–

Table 4.3: Top 20 Spam ASes (ordered according to ratio (%))

ASes are involved with Internet technology, mostly hosting and cloud providers in which malicious users set up and host their phishing websites.

Since the number of phishing IPs is small in comparison to the number of IPv4-equivalent IP addresses announced by an AS, we do not proceed with the calculation of ratio of phishing IPs in relation to the total observed.

4.4.2 Organizations-based Internet BadHoods

As discussed in Section 4.2.1, ISPs may connect to the Internet IP addresses belonging to other client organizations (`Org1` and `Org2` in Figure 4.3). This, consequently, implies that the ISP might be, in fact, routing malicious traffic originated at other organizations.

In order to clarify the responsible organization for originating the malicious traffic, in this section, we present our findings for organization-based BadHoods. Differently from AS-based BadHood, the organization-based BadHood is done by aggregating /32 IP address according to the organization they belong to – that is, the organization that has been allocated with that particular IP address.

Table 4.5 presents the Top 20 Spamming Organizations in terms of Spamming IP addresses. Comparing this table with Table 4.2, we can observe that 13 out of the 20 AS owners are found as the most Spamming Organizations. In addition, 4 organizations are subsidiaries of the AS owners: VDC is a subsidiary of the AS-owner VNPT, Reliance Communications has acquired the AS owner BSES Telecom, Telemar Norte Leste S.A. is part of the AS-owner Telecomunicacoes da Bahia S.A., ChinaNet Guangdong Province Network is routed using ChinaNet ASN. The new organizations in the list were FPT Telecom, from Vietnam, Telefonica de Espana, and Deutsche Telekom, the major telecommunications in Spain and Germany, respectively.

Table 4.6 shows the results for the organization-based Phishing BadHoods. We found that 15 out of 20 organizations, 14 are the same as the AS owners showed in Table 4.4. One organization (eToxic) is connected via SoftLayer, the #1 AS in terms of phishing IP addresses. New organizations where the following: the Endurance International Group, Media Temple, Jumpline, Interserver, which have their own ISP and ASN.

From both spam and phishing results, we can observe that organizations having more malicious IPs are correlated to the ASes having more malicious IPs. The reasons for that is that organizations having as core business Internet-related activities are more likely to have more IP addresses allocated, which increases the chances of having more malicious IP addresses. To illustrate this,

#	# of Sources	ASN	AS	CC	Business
1	140	AS36351	SoftLayer Technologies Inc.	US	Cloud Provider
2	92	AS32475	SingleHop	US	Cloud Provider
3	92	AS16276	OVH Systems	FR	Hosting Provider
4	87	AS46606	Bluehost Inc.	US	Hosting Provider
5	77	AS21844	ThePlanet.com Internet Services, Inc.	US	Merged with Softlayer
6	50	AS24940	Hetzner Online AG RZ	DE	Hosting Provider
7	48	AS47583	Aurimas Rapalis	LT	Hosting Provider
8	46	AS32613	iWeb Technologies Inc.	CA	Hosting Provider
9	44	AS26496	GoDaddy.com, LLC	US	Hosting Provider
10	40	AS7162	Universo Online S.A.	BR	Hosting and Content Provider
11	33	AS33182	HostDime.com, Inc.	US	Hosting Provider
12	29	AS11042	Landis Holdings Inc	US	Hosting Provider
13	28	AS14618	Amazon.com, Inc.	US	Cloud Provider/E-business
14	27	AS26347	New Dream Network, LLC	US	Hosting Provider
15	27	AS23352	Server Central Network	US	Cloud Provider, Datacenter
16	26	AS21788	Network Operations Center Inc.	US	Hosting and Cloud Provider
17	23	AS8560	1&1 Internet AG	DE	Hosting Provider
18	22	AS16265	LeaseWeb B.V.	NL	Hosting Provider, Cloud
19	22	AS13335	CloudFlare, Inc.	US	Cloud Provider
20	22	AS12322	Free SAS	FR	Internet Services Provider

Table 4.4: Top 20 Phishing ASes (ordered according to the absolute number of sources)

#	# of Sources	Organization	CC
1	662,224	BSNL	IN
2	468,895	SaudiNet, Saudi Telecom Company	SA
3	376,307	PTLC	PK
4	228,806	VDC	VN
5	167,167	Telefonica del Peru	PE
6	153,366	Pakistan Telecommuication company limited	PK
7	147,112	Reliance Communications	IN
8	146,533	Airtel	IN
9	135,875	Vietnam Posts and Telecommunications(VNPT)	VN
10	113,802	Mahanagar Telephone Nigam Ltd.	IN
11	112,838	Telefonica de Argentina	AR
12	102,838	RABAT3G Maroc Telecom	MA
13	100,050	Telemar Norte Leste S.A.	BR
14	99,343	Viettel Corporation	VN
15	94,524	Maroc Telecom	MA
16	94,415	ChinaNet Guangdong Province Network	CN
17	85,368	Telefonica de Espana	ES
18	85,145	FPT Telecom Company	VN
19	84,790	Tata Indicom	IN
20	82,563	Deutsche Telekom AG	DE

Table 4.5: Top 20 Spamming Organizations (absolute)

#	# of Sources	Organization	CC
1	87	Bluehost	US
2	65	OVH SAS	FR
3	47	WebsiteWelcome	US
3	47	Main Hosting Servers	US
5	44	Universo Online S.A.	BR
6	42	eToxic	US
7	39	GoDaddy.com, LLC	US
8	35	Amazon.com	US
9	32	HostDime.com	US
9	32	Hetzner Online AG	DE
11	29	Landis Holdings	US
12	27	New Dream Network, LLC	US
13	23	Website Welcome	US
14	22	CloudFlare	US
15	20	Network Operations Center	US
16	19	SingleHop	US
17	16	The Endurance International Group	US
18	15	Media Temple	US
19	14	Jumpline	US
19	14	Interserver	UK

Table 4.6: Top 20 Phishing Organizations (absolute)

consider two organizations: the American brewing company Anheuser-Busch (which produces Budweiser beer) and the Dutch Telecommunications Company KPN. Even though these companies have a comparable number of employees ($\sim$ 30,000) and revenue ($\sim$ €13 Billion), KPN's AS286 announces 3.4 million IP addresses, while Anheuser-Busch's AS15117 announces only 69.6 thousand IP addresses (a factor of 50).

Therefore, these findings also support the findings of previous sections, in which we have shown that the first two assumption behind BadHoods are valid.

4.5 Geographical Internet BadHoods

In this section we present which countries (Section 4.5.1) and cities (Section 4.5.2) having the highest concentrations of malicious IP addresses.

4.5.1 Country-based Internet Bad Neighborhoods

Country-based Spamming BadHoods

We start by showing graphically the distribution of malicious IP addresses per country in Figure 4.5. In this figure, the color assigned to each country corresponds to the number of spamming IP addresses found in each country, as shown in the legend. Analyzing the figure, we can observe that:

- *Spamming hosts are distributed all over the world.* In total, *the top 20 countries were responsible for 76,31% of all the spamming IP addresses* – which confirms the concentration of BadHoods on country-level. Moreover, the countries having more malicious hosts are concentrated in Asia, followed by South America.

- *The BRIC countries (Brazil, Russia, India, and China) are among the countries with most malicious hosts*. These countries currently experience a significant economy growth, and, in comparison to the advanced economies countries, stills have a significant part of their population without Internet access (The Internet penetration ratios are: Brazil - 40.6%. Russia - 43.0%, India - 7.5%, China - 34.3%, World Average - 35% [110]). According to a Boston Group report [111], the Internet penetration should increase between 9% to 15% per year until 2015 in the BRIC countries. Combining a growing economy with a large demand for Internet access, we can expect the number of malicious hosts in these countries to increase as more users obtain Internet access, in case measures are not taken to improve the security in the networks in these countries. To illustrate a bad scenario, if India would have the same Internet penetration rate of a comparable large country – the United States (79%) – while keeping the same ratio of malicious hosts, it would have, alone, almost 20 million spammings hosts, which is more than twice the current number of spamming IP addresses we have observed in our datasets for the whole world.

Table 4.7 presents the top 20 countries having most spamming hosts. In this table, **CC** stands for Country Code (please refer to Appendix D for the the list of countries codes). **Pop** refers to the country population, in millions (obtained from the United Nations website [112]), while **SRCs** refers to the number of malicious hosts observed per country. Finally, **Ratio** as the number of malicious IP addresses per *million inhabitants* ($\texttt{Ratio=SRCs/(Pop}\times 10^{-6}\texttt{)}$).

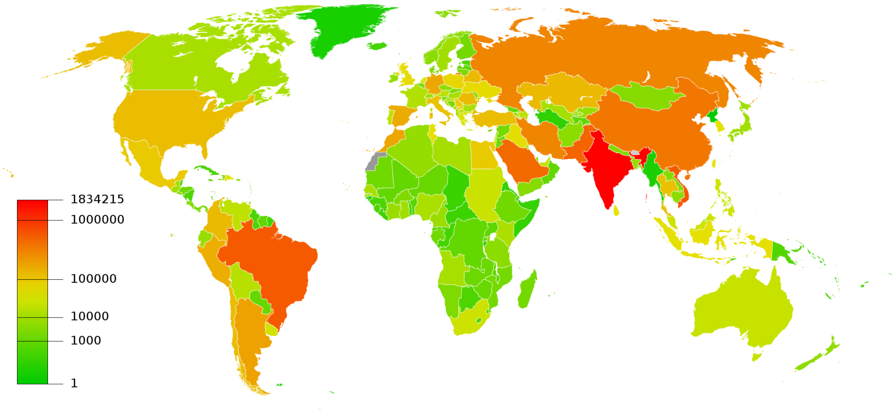


Figure 4.5: Spamming Hosts World Distribution (absolute number of spamming IP addresses per country)

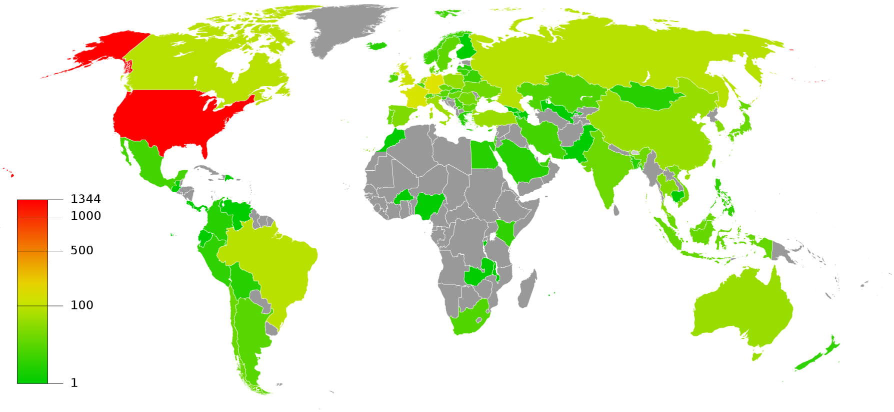


Figure 4.6: Phishing Hosts World Distribution (absolute number of phishing IP addresses per country)

Therefore, **SRCs** gives an indication of *the potential damage that can be incurred by a country*, whereas **Ratio** provides a *normalized value for the number of malicious hosts* in the country.

	Absolute				*Proportional*			
#	**CC**	**Pop(10^6)**	**SRCs**	#	**CC**	**Pop(10^6)**	**SRCs**	**Ratio**
1	IN	1,189.17	1,834,215	1	**SA**	26.13	520,979	19936.66
2	**VN**	90.54	657,611	2	**BY**	9.57	136,764	14279.64
3	BR	194.03	645,160	3	RS	7.31	89,483	12240.24
4	PK	187.34	567,029	4	MK	2.07	22,585	10872.13
5	**SA**	26.13	520,979	5	UY	3.30	35,427	10707.76
6	CN	1,336.718	427,984	6	**KZ**	15.52	140,514	9052.35
7	RU	138.73	345,594	7	**CL**	16.88	128,903	7632.47
8	**IR**	1.64	339,684	8	**VN**	90.54	657,611	7262.45
9	**AR**	41.76	214,869	9	KW	2.59	18,666	7191.32
10	**MA**	31.96	200,011	10	**RO**	21.90	139,665	6376.07
11	DE	81.47	192,322	11	AW	0.10	672	6332.87
12	ES	46.75	181,334	12	**MA**	31.96	200,011	6256.52
13	**PE**	29.24	172,767	13	QA	0.848	5,145	6067.10
14	**KZ**	15.52	140,514	14	MU	1.30	7,906	6064.19
15	**RO**	21.90	139,665	15	**PE**	29.24	172,767	5906.77
16	CO	44.72	137,162	16	TN	10.62	59,452	5593.27
17	**BY**	9.57	136,764	17	PS	2.56	13,241	5155.03
18	US	313.23	130,136	18	**AR**	41.76	214,869	5144.13
19	**CL**	16.88	128,903	19	DO	9.95	50,592	5081.22
20	TR	78.78	128,310	20	MP	0.04605	212	4603.69

Table 4.7: Top 20 Spamming Countries (Absolute and Proportional to the Population)

Analyzing the left part of Table 4.7 (Absolute), we can observe the countries that can, potentially incur more “damage”, measured by the absolute number of hosts (SRCs). In this table, India is the number one country, followed by Vietnam and Brazil. Out of the 20 countries, 17 are classified as developing countries, whereas Germany, Spain, and the United States are the only developed nations (or advanced economies, in the CIA terminology [113]) in the list of top 20 countries in absolute number of malicious spamming hosts.

On the right side of Table 4.7 (*Proportional*), we present the top 20 countries having most spamming hosts per million inhabitants, which provides an overview of the networks of these countries. Countries codes in bold font appear in both absolute and proportional lists. We can observe that India, the country having the highest absolute number of malicious IP addresses, does not even appear in the top 20 in proportional terms. In fact, none of the BRIC countries are among the top 20 most proportionally spamming countries. Furthermore, all the countries in the proportional list are classified as developing nations.

Internet Censorship Countries

An ironic (to the government) and very unfortunate fact (to their citizens) is that 6 countries among the top 20 just presented implement Internet censorship measures, by restricting their citizens' freedom of information and curtailing their access to the Internet. These countries are:

- Saudi Arabia (SA): According to the OpenNet Initiative (a joint project by Harvard and Oxford universities – among others – that has as mission "to identify and document Internet filtering and surveillance" [114]), Saudi Arabia uses a proxy farm in King Abdulaziz City for Science & Technology to "filter sites related to opposition political groups, human rights issues, and religious content deemed offensive to Muslims. Pornographic sites are pervasively filtered, as well as circumvention and online privacy tools. Bloggers have been arrested, and blogs and sites run by online activists have been blocked." [115]. Saudi Arabia is listed in the "Enemies of the Internet" list by *Reporters sans Frontières* (RSF), which is a Paris-based non-governmental organization that advocates in favor of freedom of the press [116].

- China (CN): China is responsible for maintaining the "Great Firewall Of China", which can be seen as "one of the most pervasive and sophisticated regimes of Internet filtering and information control in the world" [117]. The government performs pervasive filtering on political and conflict/security content, and substantial filtering on social content and Internet tools. China is classified as an "Enemy of the Internet" nation by RSF.

- Belarus (BY): Also part of "Enemies of the Internet" list of RSF, Belarus performs selective filtering on political, social, conflict/security and Internet tools areas, according to tests carried out by OpenNet [118].

- Kazakhstan (KZ): Kazakhstan is listed in the "Countries Under Surveillance" list of RSF. According to OpenNet, Kazakhstan preforms selective filtering on political and social content [119]

- Vietnam (VN): Vietnam is classified as an "Enemies of the Internet" by RSF. It performs pervasive filtering on political content, and selective filtering in social content according to OpenNet [120].

- Tunisia (TN): Tunisia is classified as a "Country Under Surveillance" by RSF, but it was considered an "Enemy of the Internet" before the Arab

Spring when President Ben Ali was removed from power. Under Ben Ali's regime, there was pervasive filtering on political, and social, according to OpenNet [121]. After the fall of Ben Ali, the new government has lifted the ban on social networks sites – Facebook and Youtube included.

One of the reasons why the 6 countries that implement Internet censorship and surveillance are among top 20 of countries having more malicious hosts is while trying to circumvent censorship, users might end up getting their computers infected, by accessing open proxies, malicious websites, or installing malicious tools. By becoming infected, such computers may take part in botnets, which are the source of most of current spam on the Internet [122, 123]. In addition, one could assume that censorship in ISPs of these countries is regarded as more important than the number of spammers in their networks.

Country-based Phishing BadHoods

Figure 4.6 shows the countries of the world where phishing hosts are concentrated. Analyzing this figure, we can observe that:

- Different from Spam, phishing is not distributed in the whole world. In fact, lest than 40% of the countries in the world were found having phishing hosts (92 out of 250, as in Table 4.1).

- Phishing hosts are mostly concentrated in advanced economy nations, being the United States the number one in phishing hosts. In addition, the four BRIC countries are listed as well among the top 20 phishing countries.

- There is a correlation between the number of datacenters a country has with having phishing hosts in a country[18]: all are top 20 countries in number of data centers in their territories.

The reasons for this difference between Spam and Phishing distribution over countries lies in the nature of the application/attack: spammers typically employ a large number of bots to carry out spam campaigns, while phishing has to rely on stable and reliable hosts that should be online whenever a user is redirected to it so they can steal their personal information.

Table 4.8 presents the top 20 countries in terms of number of phishing IPs. The left side of the table, the countries are ranked according to the absolute

[18] The map of data centers per country can be found at `http://www.datacentermap.com`.

number of phishing IPs, while on the right side, they are ranked according to the number of phishing IPs per million inhabitants (Ratio=SRCs/(Pop×10^{-6})).

When analyzing the right side of Table 4.8, we can notice that 13 countries listed have less than 20 phishing hosts, which are mostly classified because of their small population. The other 7 countries have observed at least 56 hosts and have a population of at least 16 million inhabitants.

Absolute				*Proportional*				
#	**CC**	**Pop(10^6)**	**SRCs**	#	**CC**	**Pop(10^6)**	**SRCs**	**Ratio**
1	**US**	313.23	1344	1	VG	0.025	10	393.96
2	**DE**	81.47	153	2	KY	0.051	1	19.46
3	**FR**	65.10	140	3	IS	0.311058	2	6.42
4	BR	194.03	94	4	**NL**	16.84	73	4.30
5	**CA**	34.03	91	5	**US**	313.23	1,344	4.29
6	RU	138.73	88	6	SG	4.74	16	3.37
7	**NL**	16.84	73	7	BN	0.31	1	3.13
8	IT	61.01	60	8	DK	5.52	15	2.71
9	TR	78.78	56	9	CY	1.12	3	2.67
10	**AU**	21.76	56	10	**CA**	34.03	91	2.67
11	CN	1336.71	54	11	**AU**	21.76	56	2.57
12	PL	38.44	52	12	**FR**	65.10	140	2.15
13	KR	48,75	36	13	BG	7.09	15	2.11
14	TH	66.72	35	14	HK	7.12	15	2.10
15	ES	46.75	33	15	IE	4.67	9	1.92
16	RO	21.90	26	16	**DE**	81.47	153	1.87
17	VN	90.54	25	17	PT	10.76	19	1.76
18	UA	45.13	22	18	CZ	10.19	17	1.66
19	IN	1189.17	22	19	SE	9.08	15	1.65
20	MY	28.72	20	20	SI	2.00	3	1.49

Table 4.8: Top 20 Phishing Countries (Absolute and Proportional to the Population)

4.5.2 City-based Internet Bad Neighborhoods

In this section we present the results of aggregating malicious hosts into city-based Internet Bad Neighborhoods. We have also investigated this topic in a previous research work [124]. However, in that work, we have employed a smaller data set from honeypots obtained from Quarantainenet [125], a Dutch Internet security firm, for a different monitoring period. We have focused on the cities and countries having more malicious hosts.

In the following, we first cover the cities having spamming hosts and then cities having phishing hosts.

City-based Spam BadHoods

We start by showing the distribution of the top 400 cities in terms of spamming hosts, as can be seen in Figure 4.7. In this figure, the size of each circle is proportional to the number of malicious hosts[19]. As shown in Table 4.1, 1.6 million /32 IP addresses (out of 9.3 million) could not be mapped into cities for spam, which were disregarded in this analysis. We should remind that 400 is only a tiny fraction of the 25,266 spamming cities we have observed.

Analyzing in Figure 4.7 we can observe that the majority of the top 400 cities are located in India (88, to be more precise), followed by Brazil with 46, then 34 in Russia, and 19 in China. Table 4.9 shows the number of spamming host for the top 20 cities. As can be seen, among the top 20, 6 cities are located in India, 2 in Saudi Arabia, 2 in Pakistan, and 2 in Brazil.

Comparing the results of this table to Table 4.7 (country-level BadHoods), we can observe that all the cities belong to countries that were found being the most malicious ones. Therefore, we can conclude that *the cities having most of spamming hosts are located in the countries that have the largest number of spamming hosts as well.*

Table 4.10 shows the cities that have most spamming hosts per million inhabitants. To obtain the population per city, we have used Maxmind's World Cities database [127], due to easy of use. Except for cities #1, #5, and #10, the cities presented a high ratio due to the small population ($< 31,000$) and small number of sources ($< 3,766$). Cities 1,5, and 10 are located in India and Pakistan, countries #1 and # 4 in terms of spamming hosts, as shown in Table 4.7.

City-based Phishing BadHoods

The results for cities having more phishing hosts are shown in Figure 4.8. In this figure, we show the geographical location of the top 400 cities having phishing hosts, out of a total of 437 (as shown in Table 4.1). Table 4.11 lists the top 20 cities having more phishing hosts in absolute numbers.

As can be seen, there is a concentration in American cities; the top four cities in terms of phishing hosts are Dallas (TX), Chicago (IL), Provo (UT), and

[19] This maps were generated using Google Geochart, which has a limitation of 400 entries [126].

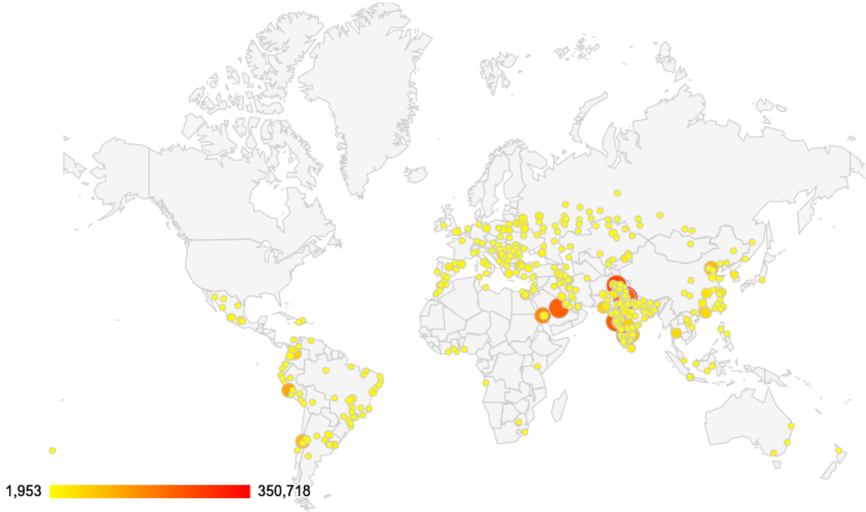


Figure 4.7: Top 400 Spamming City-Based BadHoods

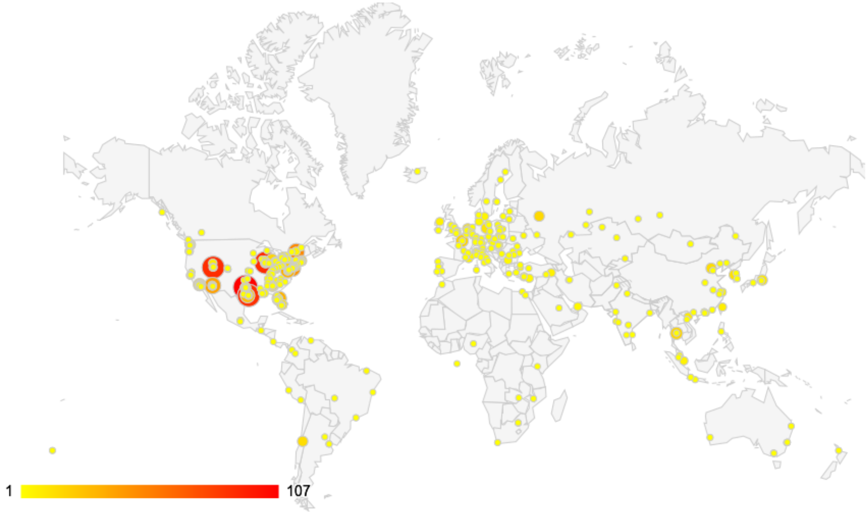


Figure 4.8: Top 400 Phishing City-Based BadHoods

#	CC	City	Sources	Population	Ratio
1	IN	New Delhi	297,638	10,928,270	27,235.60
2	PK	Islamabad	252,576	756,105	334,048.84
3	IN	Bangalore	224,213	4,931,603	45,464.53
4	SA	Riyadh	220,926	3,469,290	63,680.46
5	SA	Jiddah	163,443	2,545,728	64,202.85
6	PE	Lima	122,033	7,646,786	15,958.73
7	IN	Madras	91,308	4,328,416	21,095.01
8	CL	Santiago	90,493	4,837,248	18,707.54
9	IN	Pune	78,053	2,935,968	26,585.10
10	CO	Bogotá	70,211	7,102,602	9,885.25
11	IN	Hyderabad	69,144	3,598,199	19,216.28
12	PK	Karachi	67,563	11,627,378	5,810.68
13	PK	Lahore	65,154	6,312,576	10,321.30
14	BR	São Paulo	62,964	10,021,437	6,282.93
15	CN	Guangzhou	62,654	3,152,825	19,872.34
16	TH	Bangkok	55,047	5,104,475	10,784.07
17	CN	Shenzhen	53,091	1,002,592	52,953.74
18	IN	Calcutta	49,666	4,631,819	10,722.79
19	BR	Rio De Janeiro	46,432	6,023,742	7,708.17
20	TR	Istanbul	42,448	9,797,536	4,332.52

Table 4.9: Top 20 Spamming Cities (Absolute)

Houston (TX), all in the United States (in red). These cities all have data centers within their borders.

Table 4.12 presents the cities having most phishing hosts per million inhabitants. Cities in bold can also be found in Table 4.11. As can be seen, the majority of cities are still in the United States. One surprise was the city number 1, Road Town, located at British Virgin Islands, a small island in the Caribbean area.

4.6 Related Work

The work presented in this chapter was inspired in a previous work conducted in our research group [124]. In this work, we have carried out a study on the most “evil” cities on the Internet – that is, cities that originated most of the observed attacks. For that work, we have employed information obtained from Quarantainenet [125], a Dutch company that develops network management and security tools and provides admission control and malware detection for their customers, including more than 50% of the Dutch universities. Quarantainenet

#	CC	City	Sources	Population	Ratio
1	IN	Nagari	31,618	25,936	1,219,077.72
2	PH	Bagumbayan	3,639	2,990	1,217,056.85
3	RU	Ural	1,218	2,057	592,124.45
4	IN	Kannur	3,547	7,625	465,180.32
5	PK	Islamabad	252,576	756,105	334,048.84
6	RU	Volga	928	3,469	267,512.25
7	RS	Guca	483	2,014	239,821.25
8	RO	Moacsa	491	2,274	215,919,80
9	RU	Kushnarënkovo	1,762	10,650	165,446.00
10	IN	Gurgaon	32,551	197,353	164,937.95
11	IN	Palampur	680	4,131	164,609.50
12	RO	Balotesti	1,000	6,795	147,167.03
13	PH	Liberty	335	2,323	144,210.07
14	AM	Armavir	374	2,752	135,901.16
15	RU	Spassk	210	1,556	134,961.43
16	CO	Cogua	593	4,755	124,710.83
17	IN	Udaipur	3,766	30,266	124,430.05
18	RO	Polovragi	351	2,911	120,577.12
19	BM	Hamilton	107	902	118,625.27
20	LU	Baschleiden	21	185	113,513.51

Table 4.10: Top 20 Spamming Cities (Proportional)

has a honeypot infrastructure which is distributed mostly over the Netherlands. In total, 125 machines are used for this purpose. In this chapter, however, we use different data sets for a different monitoring period. In addition, we have not only aggregated hosts according to their geographical location, but also according to their AS and organization.

Another research work that relates to the one presented in this chapter was carried out by Shin *et al.* [47]. In that paper, the authors analyze infection data for three botnets (Conficker, MegaD, and Srizbi), and carry out a comparative analysis from each of them. They have shown how the botnets are distributed over the IP address space and also the countries in which most of the bots are located. In our work, we determine the number of malicious hosts per country according to its population, while in [47] the authors employ the number of IP addresses allocated to each country.

Most of the current research works focus on geographical location at the country level. For example, Jiang *et al.* [128] propose a spam filtering technique that uses country-level geographical information, which leads to a reduction of 13.9% in their experiments. Even though they were able to reduce the number

#	CC	City	Sources	Population	Ratio (%)
1	US	Dallas	107	1,211,704	88.31
2	US	Chicago	89	2,841,952	31.32
3	US	Provo	88	105,764	832.04
4	US	Houston	87	2,027,712	42.91
5	CA	Montreal	50	3,268,513	15.30
6	US	Scottsdale	40	225,796	177.15
7	US	Orlando	32	207,970	153.87
8	US	Brea	26	39,542	657.53
9	US	Atlanta	23	422,908	54.39
10	FR	Paris	19	2,110,694	9.00
11	US	Scranton	18	73,206	245.88
12	US	Lansing	18	117,691	152.94
13	TH	Bangkok	18	5,104,475	3.53
14	US	San Francisco	17	732,072	23.22
15	US	Atlanta	17	422,908	40.20
16	RU	Moscow	17	10,381,288	1.64
17	US	Burlington	16	22,739	703.64
18	US	Culver City	15	40,129	373.79
19	CL	Santiago	15	4,837,248	3.10
20	US	Columbus	14	736,836	19.00

Table 4.11: Top 20 Phishing cities (Absolute)

of spam messages, the authors do not describe what could happen if city-level information would be used instead of country level for filtering spam.

Sobel *et al.* [101], on the other hand, hold a U.S. patent for use of geolocation data for spam detection. It is stated in the patent that “the geolocation data may be any type of geographical information such as city, country, state or presence within a pre-selected radius of a geographical point”. As a patent, the method is only described while its effectiveness is not addressed.

Other reports on the number of attacks per country also exist. For example, the Internet hosting company Akamai provides a quartely report named ‘The State of the Internet’ [129], which is obtained from the analysis of users that access Akamai servers (many sites, such as Hulu, BBC iPlayer and the White House use the Akamai content distribution network). In their latest report, they have observed attacks from 209 countries/regions, with the U.S. being the premier one, in terms of traffic (12%). However, only 10 countries are mentioned in the report, and they do not provide an analysis at city level. Quarantainenet also provides a daily map of the countries that have attacked their honeypot infrastructure [130].

#	CC	City	Sources	Population	Ratio
1	VG	Road Town	9	8,449	1065.21
2	**US**	**Provo**	**88**	**105,764**	**832.04**
3	US	Secaucus	12	15,534	772.49
4	**US**	**Burlington**	**16**	**22,739**	**703.63**
5	**US**	**Brea**	**26**	**39,542**	**657.52**
6	**US**	**Culver City**	**15**	**40,129**	**373.79**
7	RO	Dobroesti	2	6,599	303.07
8	US	Redmond	12	47,264	253.89
9	CA	Golden	1	4,038	247.64
10	**US**	**Scranton**	**18**	**73,206**	**245.88**
11	US	Garden City	5	21,887	228.44
12	CZ	Hluboka Nad Vltavou	1	4,514	221.53
13	US	Walnut	7	31,720	220.68
14	US	Mountain View	13	68,268	190.42
15	**US**	**Scottsdale**	40	**225,796**	**177.15**
16	**US**	**Orlando**	**32**	**207,970**	**153.86**
17	**US**	**Lansing**	**18**	**117,691**	**152.94**
18	US	Herndon	3	21,499	139.54
19	US	Phoenixville	2	14,660	136.42
20	US	Arlington Heights	9	74,995	120.00

Table 4.12: Top 20 Phishing cities (Proportional)

Finally, Koike *et al.* [131] perform data visualization on the origin of attacks at IP block level or country level. And Muir *et al.* [98] present a survey on the current Internet geolocation methods.

4.7 Conclusions

In this chapter we have evaluated the three possible underlying assumptions behind the Internet Bad Neighborhood concept. We have employed real world data sets for spam and phishing in our analysis.

To investigate the assumptions, we proposed two research questions. In RQ 4.1, we have asked *how malicious hosts are distributed over ISPs?* To answer that, we have aggregated individual addresses into both ASes and organizations and compared the results. We found that the top 20 ASes *concentrate almost 50% of all spamming IP addresses* observed in our data sets, from a total of 42,201 active ASes at the moment of our analysis.

To make sure the malicious IPs announced by the ASes were not from other organizations, we also investigated organization-based BadHoods, and found a strong correlation between the AS-based BadHoods and organization-based BadHoods. Therefore, the *main contribution of this chapter is that AS-based aggregation is an efficient approach for Bad Neighborhoods*. This information can be used by network engineers to develop tools to filter traffic based on the ASes BadHoods. In addition, this finding also confirms the two first assumptions behind the BadHood concept – that some ISPs neglect malicious traffic and malware propagation in their networks.

In RQ 4.2 we wondered *how malicious hosts are distributed over geographical areas (countries and cities)*. We found that the results depend on the application: spam IP addresses are located in countries and cities distributed all over the world (concentrated in Asia), while *phishing* BadHoods are concentrated in few of countries and cities (mostly in developed nations). This shows that one can not generalize assumptions for BadHoods, and the application in question should be considered (we provide a detailed comparison of BadHoods and the different applications in Chapter 7). The reason for that is that phishing requires a reliable available web server in which forged websites can be hosted; therefore, malicious users choose to host them in data centers – which are concentrated in very few countries/cities. In addition, for spam, we found that the top 20 countries were responsible for concentrating more than 75% of all spamming IP addresses, which shows how evident is the existence of BadHoods at country-level.

In addition, we found that the BRIC countries are among the countries with the highest number of spamming IP addresses. Given their current economic growth, we can expect a significant increase in the number of malicious hosts in these countries *if measures are not taken to improve the security in such networks.* For example, if India would have the same Internet penetration rate as the United States (79%), it would have, alone, 20 million spamming IPs – twice as much of as what is observed today for *the entire world*. One might wonder if this is not the case of a silent ticking bomb. Moreover, we have found that 6 countries among the top 20 in terms of spamming hosts employ Internet censorship measures.

Therefore, the results from RQ 4.2 confirms our last assumption about the existence of Internet BadHoods: that non-technical local factors play a role in the BadHoods: phishing IP addresses are located in advanced economies, while spamming IP addresses are distributed all over the world but concentrated in Asia.

Only a free and unrestrained press can effectively
expose deception in government.

Justice Hugo Black, 1979
In: Journal of Politics, Vol 41, Issue 4.

CHAPTER 5

Case Study: Spamming Bad Neighborhoods[1]

AFTER presenting general characteristics of Bad Neighborhoods in previous chapters, in this chapter we present a specific case focusing on spamming Bad Neighborhoods.

We focus on Spamming Bad Neighborhoods (BadHoods) due to the impact caused by worldwide spam. Spam comprises approximately 90 to 95 percent of all e-mail traffic on the Internet nowadays [132, 133]. To deal with all this unsolicited e-mail, companies have to spend computer and network resources, and human labor hours, which cause economic losses. It is estimated that worldwidespam causes losses from $10 billion to $87 billion [20] yearly, and are used by cyber gangs to advertise illegal and counterfeit pharmaceutical operations [18].

In this chapter, we therefore raise the following research questions (RQ) to reveal the specifics of Spamming BadHoods:

- RQ 5.1: *What are the worst protected netblocks* (or prefixes) *on the Internet?* – that is, netblocks containing a significant number of spamming hosts.
- RQ 5.2: *What are the the most "spam-friendly" providers?*, *i.e.*, providers that "turn a blind eye" [27] to massive spammers in their networks.
- RQ 5.3: *Do Spamming BadHoods with many spammers also send many spam messages?*
- RQ 5.4: *How much data do we need to identify Spamming BadHoods?*

[1] This chapter is based on the following publication: Moura, G. C. M., Sadre, R., Pras, A.: *Internet Bad Neighborhoods: the Spam Case*. In: 7th International Conference on Network and Services Management (CNSM 2011), Paris, France, 24-28 October 2011.

These research questions have led us to four refined definitions for Spamming BadHoods. The first two take into account the two types of spammers observed on the Internet: Low-Volume Spammers (LVS) and High-Volume Spammers (HVS) [122]. The first type describes hosts “working under a central provision, each typically spamming with a low volume", while the second one consists of “dedicated spam sources, which are brute force spammers, each spamming in an enormous number every day". Since most of the LVS tend to be part of botnets [122, 123], concentrations of LVS reveal what are the “most infected networks” in the Internet – and, consequently, the worst protected ones. Therefore, this first definition addresses RQ 5.1 by identifying LVS BadHoods, *i.e.*, netblocks containing low-volume spammers.

To answer RQ 5.2, we introduce a second definition: HVS BadHoods. This type of BadHood allows us to identify providers that ignore or tolerate dedicated spammers that spam at high volume in their networks and, therefore, can be considered as “spam-friendly”. RQ 5.3, on the other hand, leads to our next definition – Spamming BadHoods Firepower – which considers the total number of spam messages sent by each netblock. By comparing the number of spam messages and spammers per BadHood, we can determine if there is any correlation between those two. Finally, RQ 5.4 leads to our last definition – All Spamming BadHoods –, in which we compare the BadHoods obtained from different data sources.

The rest of this chapter is structured as follows. Section 5.1 details the four definitions of Spamming BadHoods. Next, Section 5.2 presents the datasets employed in our analysis. Following that, Section 5.3 shows the experimental results. Related work is discussed in Section 5.4. Finally, Section 5.5 contains our conclusions and proposes future work.

5.1 Four definitions for Spamming Bad Neighborhoods

In this section, we propose four definitions that allow us to gain more insight into the behavior of spammers and the networks hosting them. We begin with a brief discussion of the possible data sources to evaluate Spamming BadHoods in Section 5.1.1. Then we present the four definitions of Spamming BadHoods from Sections 5.1.2 to 5.1.5.

5.1.1 Possible Data Sources for BadHood Analysis

In order to identify and analyze Spamming BadHoods, we need to obtain the IP addresses of the spamming hosts. Several data sources can be employed for that. Next we present an overview of those sources.

DNS Blacklists

One approach to identify IP addresses of spammers is to set up *spamtraps*, which are specialized honeypots to collect spam. By definition, every message that reaches a trap is considered spam, since it was unsolicited in the first place. Their source IP addresses can be used to build blacklists, usually in real time. The term “DNS Blacklist” comes from the fact that many blacklist maintainers allow queries to be made to their blacklists in a similar way DNS queries are performed. A comparison of blacklists can be found at [134] and [135]. Blacklists do not necessarily list the full IP address of every single spammer. In fact, some lists only provide aggregated information on whole subnetworks. Even though DNS blacklists list many IPs, they do not provide the information on *how many spam* messages a spammer has sent – they only tell that a certain IP has sent spam.

Mail Server Logs

Most of the spam is currently detected on mail servers, where incoming messages are processed and filtered. Mail filters, such as SpamAssassin [61], are configured to perform a series of checks on every e-mail message. These tests can include header and text analysis, Bayesian filtering, and even take input from DNS blacklists. Depending on the outcome of the tests, each message is classified as “spam" or “ham”. Differently from DNS blacklists, it is possible to determine how many spam messages were sent by each IP address using mail server logs.

Mail Client Logs

Spam mails can also be detected by the mail client itself. This is usually the last resort against spam because it does not avoid the increased bandwidth usage caused by unsolicited mails. Similar to the mail filters used by mail servers, the mail client, such as Thunderbird [136], can perform a series of tests in order to classify the mails.

Network Flows

According to the IETF, a *network flow* is defined as a “set of IP packets passing an observation point in the network during a certain time interval that share the same properties” [53]. Typically, these properties include the source and destination port and address of the packets, as well as other IP header fields, like the protocol number. Flow probes monitor the packets in a network and export so-called *flow records*, which contain summarized information on the identified flows, such as the number of exchanged packets.

Flow monitoring can be used to detect spam [137, 138, 139]. However, since flow records only provide an aggregated view to the network traffic, the validation of the detection results is much harder and has to be based on statistical arguments. Hence, we do not employ this data source in our analysis.

5.1.2 First Definition: LVS BadHoods

Low-Volume Spammers (LVS) are hosts that spam at a low volume to avoid being blacklisted. Typically, they are operated under a central provision, usually as part of a botnet [122]. The latter obviously requires that the host has to be firstly infected. Hence, a concentration of a high number of LVS in a subnetwork indicates that the particular subnetwork is poorly protected or managed, and that the responsible ISP migth neglect the malware propagation in their networks. The goal of this definition for Spamming BadHoods is to detect the worse protected (or infected) subnetworks by identifying the netblocks with many LVS. We refer to such BadHoods as LVS BadHoods.

The first step is to classify spammers according to the number of messages each of them has sent during the observation period. Note that this information is not provided by blacklists, so we have to rely on the other data sources. Since spammers can behave differently across different domains, we combine the data obtained from several observation points. After that, we need to establish a threshold θ that we apply to the number of sent messages in order to separate LVS from other spammers. We define

$$\theta = d \times s \times m \tag{5.1}$$

where d is the length of the analyzed data trace in days, s is the number of different domains being monitored, and m is the maximum number of messages that a spammer can send to a single domain per day in order to be considered a LVS. As described in [122, 123], a LVS usually contacts the same mail server once or twice a day. Hosts spamming under this threshold are classified as LVS.

After classifying each host, we count the number of LVS per /24 netblock. For example, if the following set of /32 IP addresses were listed `1.1.1.5, 1.1.1.4, 1.1.1.64`, the block `1.1.1.0/24` would have a count of three. The blocks are then ranked by their count (score). The maximum possible score per each /24 block is 254, since usually the addresses with `.0` and `.255` suffix are reserved for network identification and broadcasting, respectively. Aggregating data on /24 level is preferable because this is the smallest prefix length that can be delegated on the Internet [31].

5.1.3 Second Definition: HVS BadHoods

Complementary to LVS, High-Volume Spammers (HVS) are those hosts that spam in high volumes, *i.e.*, that send more messages during the observation period than specified by the threshold θ in Equation 5.1. As for LVS, blacklists cannot be used to identify HVS.

HVS are usually dedicated spamming hosts operated by professional spammers. Therefore, a high concentration of HVS in a particular subnetwork indicates that the ISP tolerates them. To identify HVS BadHoods, we follow the same approach employed in the first definition, that is, we count how many spam messages each spammer has sent during the observation period. Spammers above the threshold θ are considered HVS. After classifying each host, we count the number of HVS per /24 netblock and rank them according to that number. As in the first definition, the maximum score for a netblock is 254.

5.1.4 Third Definition: Spamming BadHood Firepower

The third definition for Spamming BadHoods focuses on evaluating the “firepower” of each netblock. Therefore, we identify the most spamming BadHoods on the Internet in terms of the number of sent spam messages — not on the number of spamming hosts. As for the two previous definitions, we have to rely only on mail server and clients logs as data sources, since DNS blacklists do not provide all needed information. The first step is to count how many spam messages each spammer has sent. Next, for each /24 netblock, we calculate the total number of spam messages sent by all the spammers located in the block. The final step is to rank the blocks according to that number.

Blacklist	Aggregation Level	# of entries (04/21/2010)
CBL	/32	8,334,895
PSBL	/32	2,445,270
UCEPROTECT-1	/32	3,350,417
UCEPROTECT-2	Various	30,143
UCEPROTECT-3	ASN	867
SBL	Various	10,535

Table 5.1: DNS blacklists obtained

5.1.5 Fourth Definition: All Spamming BadHoods

The goal of the last definition for Spamming BadHoods is to identify all spamming netblocks, independently of the spammers' behavior. Therefore, we need only the IP addresses of spammers, which allow us to include the information provided by DNS blacklists into our analysis. From each data source, we extract the IP addresses of the spammers. Then, all these IP are combined and duplicate entries are removed. Next, we count the number of spammers in each /24 block and rank the blocks according to the count.

5.2 Evaluated Datasets

In this section we present the data we have used in our experiments. We have obtained data from DNS blacklists, mail server logs, and mail client logs from various sources over a period of one week (April 19-26th, 2010). Next we describe in more detail the collected data.

5.2.1 DNS Blacklists (DNSBL)

Table 5.1 shows the DNS blacklists we have obtained. We have chosen these since they have been previously investigated by academic communities and they are employed in production mail filters. The first one, Composite Blocking List (CBL) [60], maintains four large spamtrap infrastructures from where the source IP addresses of spammers are harvested. To give an idea of the size of their spamtraps, one of the four traps they maintain has received, on average, 2831 spams per second over a period of one year [140]. As shown in Table 5.1, on April 21st more than 8 million unique IP addresses were listed on CBL.

Another blacklist we have obtained is the Passive Spam Block List (PSBL) [75]. PSBL is built using their own spamtraps, which capture around 500 thousands spam messages per day [141]. More than 2 million unique IP addresses were listed by PSBL on April 21st, 2010, as shown in Table 5.1.

The next blacklists are from UCEPROTECT-Network [142]. They have three blacklists: The Level 1 blacklist lists only single IP addresses (/32) of spammers that have contacted their spamtraps. For example, on April 21st, 2010, more than 3 million unique IP addresses were listed on the Level 1 blacklist. The Level 2 blacklist, on the other hand, is automatically generated based on Level 1. In that list, netblocks are entirely blacklisted according to a scoring procedure. The following entry is present on April 21st, 2010 on the Level 2 blacklist: *"86.99.128.0/17 is UCEPROTECT-Level2 listed because 267 abusers are hosted by EMIRATES-INTERNET Emirates Internet/AS5384 there.*". Finally, the Level 3 blacklist lists all IP addresses from an autonomous system (except those whitelisted at ips.whitelisted.org) if "more than 100 IPs, but also a minimum of 0.2% of all IPs allocated to this ASN got Level 1 listed within the last 7 days" [142]..

Finally, the Spamhaus Block List (SBL) [143] lists IP addresses at different aggregation levels "from which Spamhaus does not recommend the acceptance of electronic mail" [143]. SBL contains single IP addresses as well as entire network blocks. As can be seen in Table 5.1, more than 10 thousand entries are present on SBL on April 21st, 2010.

Since blacklists are usually built using a large number of honeypots, they have higher probability to be reached by many different spammers than a single mail server. For example, UT/EWI mail servers have been spammed by 71,754 different IP addresses on April 21st, 2010, while CBL spamtraps lists more than 8 million unique IP addresses. However, DNS blacklists list only the IP addresses of spammers, while mail server logs list every single spam message – which allows to compute how many spam each spammer has sent.

5.2.2 Mail Servers Logs

Table 5.2 shows the mail servers from which we have obtained data. Provider A is a large hosting provider located in the Netherlands. Almost 7 million messages from more than 1.5 million different IP addresses were tagged as spam for the monitoring period (1 week).

Next, we have analyzed data from the mail server of the Electrical Engineering, Mathematics, and Computer Science Faculty at the University of Twente

Domain	Country	# of Spam Msgs	# of distinct IPs
Provider A	The Netherlands	6,981,415	1,668,205
UT/EWI	The Netherlands	1,707,367	458,495
CAIS/RNP	Brazil	84,295	36,938
Provider B	France	1,160	975
Total	–	8,774,237	1,847,874

Table 5.2: Mail servers log files analyzed

(UT/EWI)[2]. In total, more than 1.7 million messages were logged.

Since we did not want to limit this research to data from mail servers from the Netherlands, we have also obtained data from the mail servers of the Security Incident Response Team of the Brazilian Research Network (CAIS/RNP) [144]. More than 80 thousand spam messages were obtained for the monitoring period. Finally, we have obtained data from a small mail server hosted in France, denoted as Provider B, from which we got 1,160 spam messages from 975 distinct senders. In total, we have obtained more than 8.7 million messages from more than 1.8 million different IP addresses from mail servers.

5.2.3 Mail Client Logs

Finally, the last type of data collected was mail client spam logs. For this work we have obtained 1,321 spam messages from 15 mail accounts from various countries. These messages, in turn, came from 763 different senders. Since this dataset is not as representative in comparison to the previous ones, we did not further employ it in our analysis.

5.3 Experimental results

In Section 5.1 we have introduced four definitions for Spamming BadHoods. In this section we apply the different definitions to the datasets presented in Section 5.2 and discuss the results.

[2] UT/EWI has, in fact, a primary and a secondary mail server. We have combined the data from both as a single source. However, we should point out that spammers have targeted much more the secondary one. We assume that spammers believe that secondary servers are not as well secured as primary ones, which is not the case for UT/EWI servers.

	# of IPs	# of Spam	Spam/IP
$x = 1$	867,422 (46.94%)	867,422 (9.80%)	1
$1 < x \leq 10$	821,472 (44.46%)	3,189,391 (36.35%)	3.88
$10 < x \leq 56$	145,648 (7.80%)	30,53,351 (34.80%)	20.96
$x > 56$	13,081 (0.70%)	1,662,913 (18.95%)	127.12

Table 5.3: Distribution of Spam Messages from Mail Server Logs (1 week)

5.3.1 LVS BadHoods

For this definition, we have combined mail server logs from four different domains ($s = 4$): Provider A, UT/EWI, CAIS/RNP, and Provider B. By combining those log files, we increase the chances of observing the same spammer on different mail servers which allows us to better classify it. The logs cover a period of seven days ($d = 7$).

Table 5.3 shows the distribution of the number of spam messages per spamming hosts for the combined mail server logs. For a given number x of spam messages sent by a single spammer, the table gives the number of spammers (second column) that match it, followed by the total number of spams sent by those spammers (third column) and the average number of spams per spammer (fourth column). The numbers in parentheses give the percentages of the total number of spammers and spam, respectively, found in the dataset.

We choose $m = 2$ as the maximum number of mails sent by a LVS to a domain per day, as described in Section 5.1.2. Hence, Equation 5.1 leads to a threshold θ of 56 spam messages, meaning that spamming hosts sending at most 56 messages are classified as LVS while the others are HVS. By employing $\theta = 56$, one can observe that most of the spammers (99.3%) are classified as LVS (first to third rows in Table 5.3) and that they are responsible for around 80% of all spam our mail servers have received. Since most of LVS are believed to be bots [122, 123], we can conclude from our results that probably most of the spam nowadays comes from botnets. In addition, we observe that nearly 50% of all spammers have only sent one message (first row), which confirms the tactic adopted by LVS to spam at very low volume to avoid being detected [122].

Figure 5.1 shows the distribution of LVS BadHoods over the IP address space. The x-axis gives the /8 prefix of the IP address; the y-axis gives the number of spamming LVS hosts per /24 block. Each point in the plot stands for one /24 block. The horizontal line shows the maximum possible number of hosts in a block, that is 254. We observe that there is a high concentration of spamming hosts on certain ranges, such as between 60-100, 110-125, and finally 180-200.

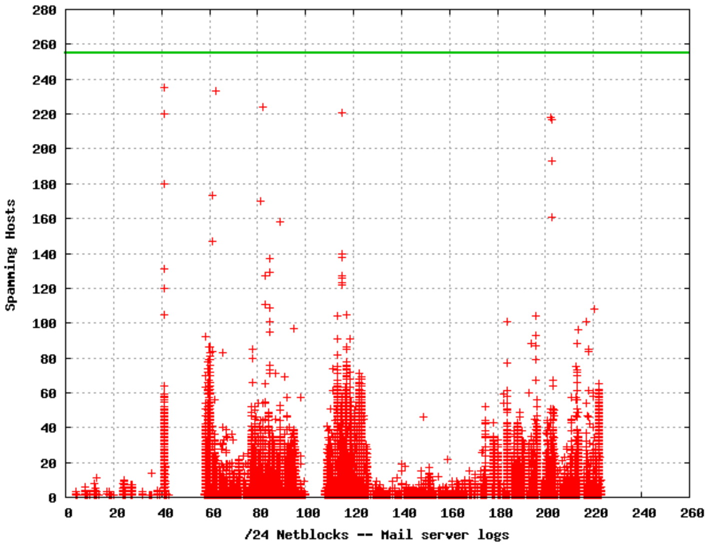


Figure 5.1: LVS BadHoods – Number of Spamming Hosts per /24 prefix

There are also IP ranges where we did not observe a single spammer. This includes the following ranges: 28-31, 44-49, 101-107. The reasons vary: some blocks were not allocated by IANA when we collected the data [67] (31,49,101-106). Others were legacy blocks (28-30, 45-47, 49), which were blocks usually assigned by the central Internet Registry (IR) prior to the Regional Internet Registries (RIRs). However, these blocks are managed by individual RIRs. Finally, the 46 block was allocated in September 2009, while 107 was allocated in February 2010.

For the blocks with the highest number of LVS spammers, we have identified the corresponding ISP. In the middle column of Table 5.4, we show the top 20 providers that manage the /24 most malicious LVS BadHoods i.e., the high scores in Figure 5.1. Analyzing this table, we observe that some LVS BadHoods are made up almost exclusively of spamming hosts; by evaluating only four different mail server domains, we were still able to find networks such as the case of Libyan Telecom having 235 spamming hosts in a single /24 netblock.

Rank	LVS BadHoods	HVS BadHoods
1	Libyan Telecom (235)	PTK Serbia (27)
2	Omantel Tech (233)	Digitel Venezuela (16)
3	Omantel Tech (224)	Maroc Telecom (13)
4	Digitel Philippines(221)	Smart Indonesia (13)
5	Libyan Telecom (220)	Vodafone Romenia (11)
6	Excelcomindo Philippines (218)	Smart Indonesia (11)
7	Smart Telecom Indonesia (217)	Digitel Venezuela (11)
8	Smart Telecom Indonesia (193)	Yahoo! Europe (10)
9	Libyan Telecom (180)	Telefonica Chile (10)
10	CAT Wireless Bangkok (173)	Maroc Telecom (8)
11	Maroc Telecom (170)	BSNLNET India (8)
12	Smart Telecom Indonesia (161)	Korea Telecom (8)
13	TATTELECOM Russia (158)	Chinanet (8)
14	CAT Wireless Bangkok (147)	Orange Romenia (8)
15	Digitel Philippines(140)	Orange Romenia (8)
16	Digitel Philippines (138)	OJSC MegaFon Russia (7)
17	OJSC MegaFon Russia (137)	Kazan Russia (7)
18	OJSC MegaFon Russia (137)	KORNET Korea (7)
19	Libyan Telecom (131)	KPN Netherlands (7)
20	OJSC MegaFon Russia (129)	CODETEL-Dominican Rep (7)

Table 5.4: Providers of the Top 20 Most Malicious /24 Networks (number of hosts between parentheses)

Since most of LVS are believed to be bots, these results show how some ISPs neglect the propagation of bots and malicious activities that the hosts in their networks carry. This also confirms the facts that some DNS blacklists block entirely /24 netblocks, as SBL [143].

As explained in Section 5.1.2, we can assume that the most malicious LVS BadHoods are also the worst protected and, consequently, the most infected ones. The presented results can be used by providers to raise awareness about the security of their networks and to improve it. In addition, LVS Bad Neighborhoods can also be employed to track and detect botnets [123].

5.3.2 HVS BadHoods

HVS BadHoods are determined in a similar way as the LVS BadHoods. We have employed the same datasets, period of time and threshold. As shown in Table 5.3, only 13,081 (0.70%) IP addresses have been classified as HVS. The distribution of the HVS over the IP address space is visualized in Figure 5.2. Each point gives the number of HVS in one /24 block. We observe that most

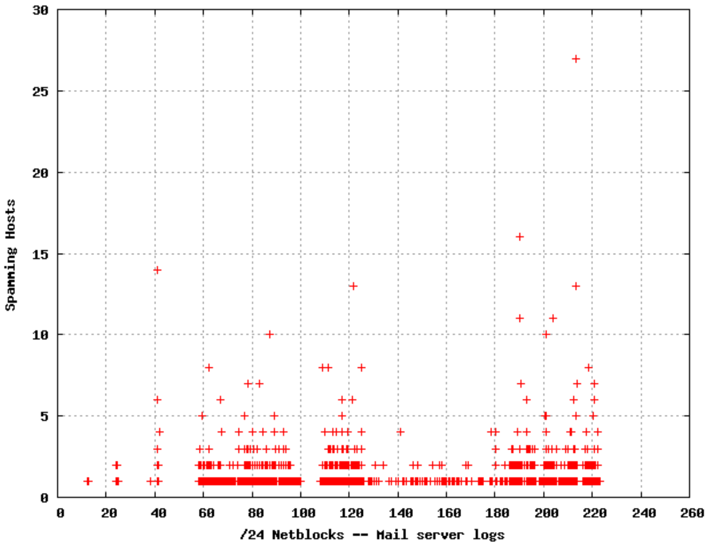


Figure 5.2: HVS BadHoods – Number of Spamming Hosts per /24 prefix

BadHoods host less than three HVS. Remarkably, some blocks contain up to 27 HVS – which is far less than for LVS cases, as shown in Table 5.4.

In the most right column of Table 5.4, we show the top 20 “spam-friendly” providers. Differently from LVS BadHoods, we can find providers for HVS from Europe, Africa, Asia, Russia, and South America among the top 20. Even though the European Union has a directive that regulates spam [49], each member state is responsible for “*taking appropriate measures to ensure that [...] unsolicited communications for purposes of direct marketing [...] are not allowed either without the consent of the subscribers*”. Our results show that 5 of the top 20 HVS BadHoods are located within the EU borders, which raises doubts on the effectiveness of such directive.

Another interesting fact to observe is that Yahoo! Europe ranks number 8 in the Top HVS list. Checking manually the 10 IPs, we found out they are, in fact, mail servers from Yahoo! Mail located in the UK. This might be due to account hijack, in which spammers hijack legitimate accounts to send spam [145, 146].

To conclude, HVS BadHoods show in which blocks HVS are located, thus allowing us to identify “spam-friendly” providers. However, a complete analysis on that is provided in Chapter 4. These results can be used to raise awareness

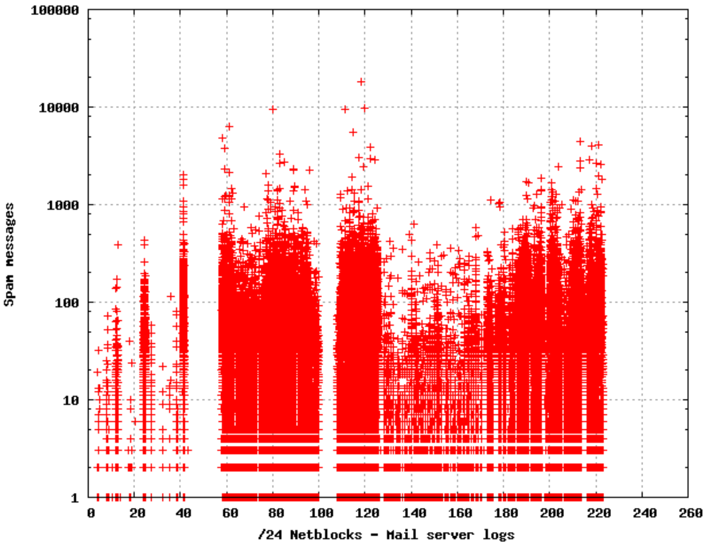


Figure 5.3: Spamming BadHoods Firepower

about those providers and, in some cases, alert to the number of hijacked mail accounts. This could be used by mail filters to appropriately rank mail from such netblocks and by ISPs.

5.3.3 Spamming BadHood Firepower

So far, we have analyzed BadHoods based on the number of spamming hosts per netblock. In this section we evaluate neighborhoods in terms of their firepower, *i.e.*, their impact measured in number of spam messages they have sent. Again, we rely on the mail server logs to calculate the total number of spams sent per /24 netblock. The result is shown in Figure 5.3. Each point represents one /24 block. Note the logarithmic scale of the y-axis.

In the beginning of this chapter, we have raised the question whether the Spamming BadHoods with most spammers are also responsible for most of the spam. In Figure 5.4, we show the number of spams sent by a /24 block as func-

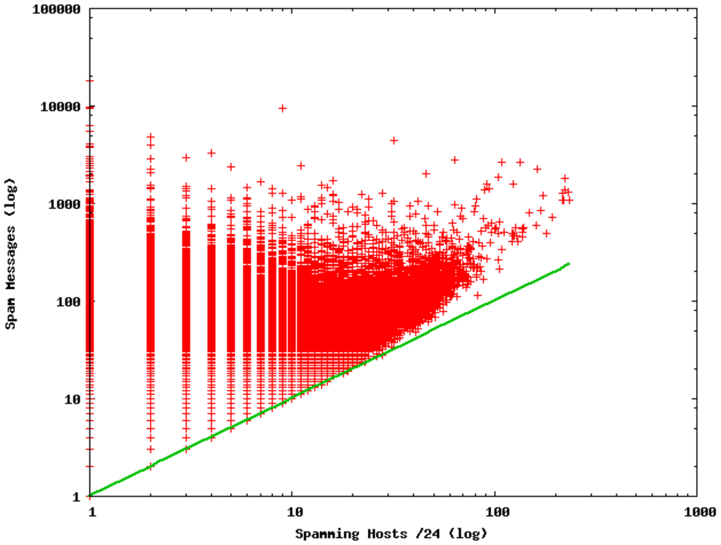


Figure 5.4: Number of Spam Messages versus Number of Spamming Hosts per /24 block

tion of the number of spamming hosts in that block. Each point represents one /24 block. The diagonal line in the plot visualizes the minimum number of spam messages that a netblock can send (which is equal to the number of spammers). We notice a large variation in the number of spam messages per netblock. The figure also shows that a higher number of spammers does not necessarily implies that a BadHood also sends more spams. Especially for blocks with up to ten spammers, there seems to be even a reverse relationship. Consequently, the Pearson correlation [147] between the number of spamming hosts and the number of spam messages per netblock is quite weak with a coefficient of only 0.32.

Our analysis also reveals how spam is distributed according to the Spamming BadHoods. Figure 5.5 presents the cumulative distribution function for the spam messages, where the BadHoods were ordered according to their firepower. As one can see, the majority of spam comes from a small fraction of all BadHoods. In fact, 10 % of the BadHoods were responsible for 54.87% of all spam. This suggests that, just by fighting a small subset of the malicious BadHoods, we should be able to block the majority of spam.

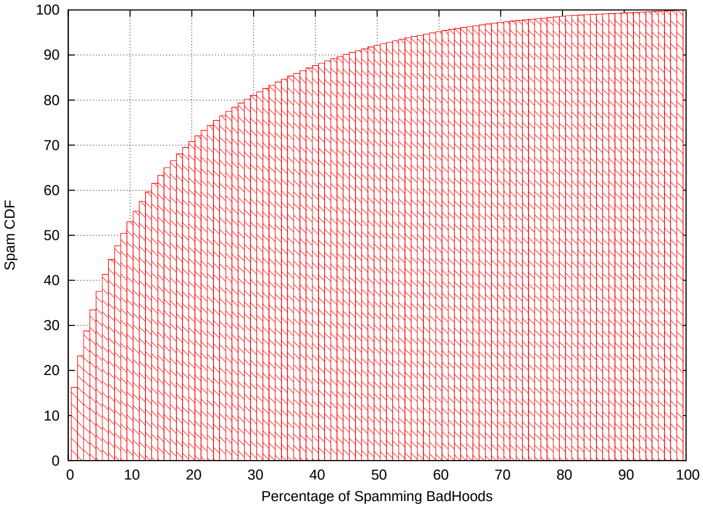


Figure 5.5: Spam CDF

As a matter of fact, these results show the strength of the Bad Neighborhood concept. In Table 5.3, we observed that 46.94% of spammers (/32 hosts) generate only 9.8% of the total amount of spam – which poses a major challenge for DNS blacklists-based spam detection. However, by employing the Bad Neighborhood concept (/24 netblocks), we are able to invert this situation: we found that 10% of Spamming BadHoods were responsible for 54.87% of all spam.

Given the threshold of $\theta = 56$ spam messages, when ranking badhoods according to their firepower, many of them that have a small number of spammers must, therefore, contain HVS. In fact, among the top 10 worse Spamming BadHoods, six have only one spamming host, two have two spamming hosts, one has nine, and the last one has 32 spamming hosts.

This is also shown in Figure 5.4. We can see that the most severe Spamming BadHoods are, in terms of number of spam messages sent (or firepower), HVS BadHoods. However, most of BadHoods are classified as LVS. Even though the HVS BadHoods are a minority, what we can learn from these results is to not underestimate the HVS BadHoods firepower and the damage that they can incur. On the other hand, the average firepower of the individual LVS BadHoods is far lower. LVS BadHoods become powerful through their sheer number.

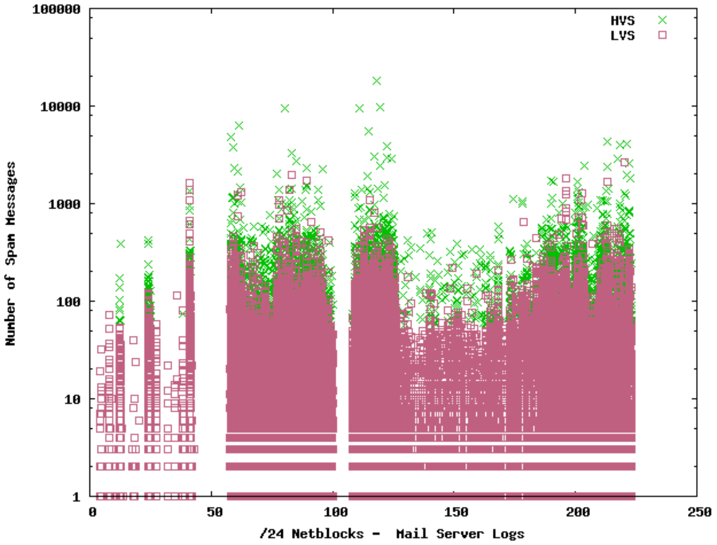


Figure 5.6: All Spamming BadHoods

5.3.4 All Spamming BadHoods

As explained in Section 5.1.5, the goal of the fourth definition for Spamming BadHoods is to identify all spamming netblocks, independently of the spammers' behavior. For our analysis, we used the data from the /32 blacklists (CBL, PSBL, UCEPROTECT-1) and the mail server logs (Provider A, UT, CAIS/RNP, Provider B) covering the period of April 19-26, 2010. This resulted in a list of more than 124 million entries with 15 million unique IP addresses. We aggregated the data by counting the number of spammers for each /24 block. By doing this, we found 1,205,888 /24 netblocks with at least one spammer. Figure 5.6 shows the distribution of the spammers over the IP addressing space. Each point gives the number of spammers of one /24 block. The x-axis specifies the /8 prefix of the blocks.

Our main motivation for this definition is the question, how much data is actually needed to identify all Spamming BadHoods. Comparing Figure 5.6 with Figure 5.1, which has been generated only using mail server logs, we observe a similarity between the results. In both figures, we can identify the same regions with high, respectively low, activity. In total, mail server logs have allowed us to identify 571,389 Spamming BadHoods, while DNS blacklists combined together with mail server logs allowed us to detect 1,205,932 Spamming BadHoods. The difference of 634,543 between the two shows how much extra BadHoods have been identified by using the additional information from DNS blacklists.

However, we should put this into perspective: the blacklists we have used in our analysis have provided more than 115 million entries, while the mail servers have provided us only 8.7 million (a factor of 13). But, by using the blacklists, we were able to identify only 2.11 times more BadHoods. We can conclude that even though blacklist provide much more data, mail server logs perform quite well when finding Spamming BadHoods.

5.4 Related Work

Several research works have suggested that malicious hosts are concentrated on some subnetworks on the Internet. The network level behavior of spammers was analyzed in [27]. The authors have collected spam from a spam sinkhole for more than one year. They have shown that most of spam comes from a few concentrated parts of IP address space. In our work, however, we have obtained spam from mail servers from production networks – which have been running legitimate mail servers for years – and DNS blacklists [33]. DNS blacklists, such as [60, 75, 143], list malicious IP addresses at different aggregation levels, suggesting concentration on some sub-networks. In [28], the authors have defined the concept of uncleanliness, that “works as an indicator for how likely the network is to contain compromised hosts".

In another work [122], the authors have set up an open relay for e-mail and proposed a classifcation for spammers we have employed in this work: low-volume spammers (LVS) and high-volume spammer (HVS). We have extended this concept to BadHoods and employed it in our analysis. In [148], the authors have conducted an experiment by becoming part of a spamming botnet. They were able to observe a spam campaign over a period of week, in which 400 million spam messages were sent.

Late in November 2012, Pitsillidis *et al.* have scrutinized more than *one billion* spam messages [149]. They have extracted the malicious uniform resource locators (URLs) presented in the messages and analyzed the spamming domains listed in the messages (e.g., `pharma-cheap.biz`). Our work differs from theirs since ours considers the IP addresses of the spam senders instead of the URLs of the spamming hosts.

The Bad Neighborhood concept was introduced in [31]. In that study, the authors collected data from several DNS blacklists and counted the number of spamming hosts per /24 prefix. The resulting count was then transformed into a score for the /24 netblock. Together with other data, they employed this score to determine whether an e-mail would be spam or not, based on its sender's IP. If the message originated from a "bad neighborhood", *i.e.*, from a /24 netblock with a high score, it was rated as spam, even if the particular sender IP was never observed as spammer before. Our work goes beyond this work, by defining and analyzing four types of Spamming BadHoods. The results provided in our work can be used to further develop current detection algorithms.

5.5 Conclusions

In this chapter we have refined the Internet Bad Neighborhood definition presented in Chapter 1, using as a case study Spamming BadHoods. We have raised four research questions that led us to four definitions for Spamming BadHoods.

The RQ 5.1 was "*What are the worst protected netblocks*". We have defined and evaluated LVS BadHoods to answer this question. The results have shown that some ISPs seem to completely neglect the malware propagation in their networks. We can use these results to raise awareness on the most infected networks and to improve security levels on those networks. Such BadHoods can also be employed to characterize botnets.

RQ 5.2 focused on "*what are the most spam-friendly providers*". We have defined HVS BadHoods to answer this question, and identified such friendly providers. The presented results can be used to raise awareness about those ISPs and to question the effectiveness of Spam legislation, such as the EU directive 2002/58 [49]. The HVS information could be used by mail filters to block, or at least appropriately rank, mails from such BadHoods.

RQ 5.3 question was whether "*Spamming BadHoods with many spammers also send many spam messages?*". Our analysis revealed that *this is not the case*. In fact, the top 10 Spamming BadHoods had no more than 32 spamming hosts (and 6 of them had only one, including the most active BadHood). In addition, we have shown that most spam comes from a fraction of all BadHoods. The lesson we can learn is that we should not underestimate HVS BadHood firepower. And that the list of HVS providers should be used to raise awareness and improve security of such providers.

Finally, RQ 5.4 addressed "*how much data do we need to find Spamming BadHoods*". We have shown that DNS blacklists help to obtain twice as much BadHoods than when only relying on our mail server logs. However, they have listed 13 times more IPs. We can conclude that even though blacklists provide more data, mail server logs perform quite well when finding Spamming BadHoods.

For more than 15 years, the Internet community has been fighting spam, and the problem is still far from being solved. In this chapter, we have provided an insight on Spamming BadHoods, taking into account spammers behavior and firepower. The definitions and results presented can be used to refine current solutions to fight spam.

Part III

Defending Against Bad Neighborhoods

The most effective way to restrict democracy is to transfer decision-making from the public arena to unaccountable institutions: kings and princes, priestly castes, military juntas, party dictatorships, or modern corporations.

Noam Chomsky, 1998
In: Z Magazine

CHAPTER 6

Bad Neighborhood Blacklists from other Sources[1]

INTERNET blacklists (lists of malicious /32 IP addresses) can be obtained from *public sources* and *peer sources*. Public sources are those that generate blacklists based on the incoming traffic observed by their honeypots/traps and make them publicly available on the Internet. For example, the Passive Spam Block List (PSBL) [75] is a public spam blacklist, among many available on the Internet. Peer sources, on the other hand, are sources that generate blacklists and make them available only to few and by means of *private agreements*. When a blacklist (/32) is aggregate into a bad neighborhood, we refer to that as Bad Neigbhorhood blacklist (BadHood blacklist).

In this chapter, however, the goal is to determine *how much a network manager can rely on others' BadHood blacklists* to secure a host that he/she maintains (we refer to this host as *target* in the rest of this chapter). *Others*, in this context, refers to external sources from which we can obtain blacklists.

The motivation to undertake this study is the following: if most of the BadHoods attacking the (to be secured) target are also listed on BadHood blacklists from other sources, the network administrator could then *effectively* employ such blacklists to feed BadHood-based defense mechanisms – such as network/application firewall and IDS – in order to protect the target. Moreover, that eliminates the need to carry out local measurements to generate the target's own blacklist (target worst offender list (TWOL)), since blacklists obtained from other sources could be employed. Since attackers may check the IP addresses under their control against publicly available blacklists – and re-

[1] This chapter is based on the following publication: Moura, G. C. M., Sperotto, A., Sadre, R., Pras, A.: Evaluating Third-Party Bad Neighborhood Blacklists for Spam Detection. In: IFIP/IEEE International Symposium on Integrated Network Management (IM 2013), Ghent, Belgium, 27-31 May 2013 (*to appear*).

frain from using such blacklists addresses – we are interested in finding whether locally generated blacklists improved detection rates. In the context of this dissertation, we focus on blacklist-based defense mechanisms, such as spam filter rules employed by van Wanrooij and Pras [31].

Therefore, the main research question investigated in this chapter is the following: *how much can a network administrator rely on BadHood blacklists obtained from other sources to protect a target?*

One of the advantages of public blacklists is that they are usually generated using a large number of honeypots/traps distributed over many networks, which increases the probability of blacklisting more sources. Peer sources blacklists, on the other hand, are usually generated based on the incoming traffic to one or few targets. However, there are some disadvantages of employing blacklists from public sources. The main one is the fact that the dependability of the security solution designed to protect the target is put at stake, since it relies on the *availability* of third-party no-warranty freely distributed blacklists. Such blacklist sources might fail for various reasons – a disruption in the service can occur (e.g, PSBL users experienced a 4 day period outage in November 2011 due to bad weather conditions [75]), the public source might become victim of DDoS attacks, or change their business model and charge for access, or even stop providing blacklists overnight. Therefore, in this chapter we also evaluate blacklists from peer sources, supposedly more trustworthy, since they are endorsed by a private agreement.

Taking into account the source of the blacklists, we divide the previously raised research question into two sub-questions:

1. RQ. 6.1: How much can a network administrator rely on *public* BadHood blacklists to protect a target?
2. RQ. 6.2: How much can a network administrator rely on *peer* BadHood blacklists to protect a target?

Figure 6.1 presents a summary on the usage of both public and peer sources for protection of a target. In this figure, the target is protected from the Internet by a BadHood-based firewall, which employs public or peer sources as input.

To answers the sub research questions, we first identify both public and peer sources from which we can obtain blacklists. Next, for each source, we obtain daily blacklists for the same monitoring period (1 week). Then, we compare the public and peer BadHood blacklists to the target worst offender list (TWOL), as shown in Figure 6.1. The idea is to determine if the attacks occurred in a similar way – meaning the same number of attacking BadHoods, the same instances of

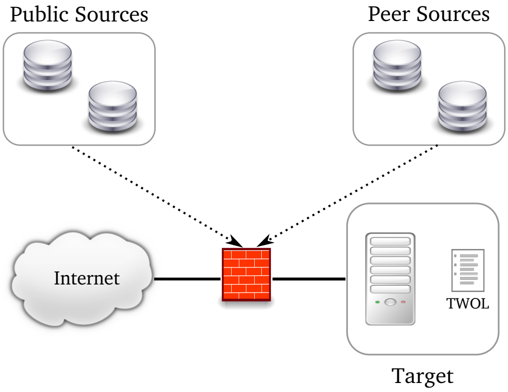


Figure 6.1: Blacklist Sources for Target Protection

these BadHoods, and the same intensity. This, ultimately, allow us to answer the main research question.

The rest of this chapter is organized as follows: in Section 6.1, we present the public and peer sources from which we have obtained blacklists. After that, we introduce in Section 6.2 the methods employed to compare the BadHood blacklists. Next, we address RQ. 6.1 in Section 6.3, in which our results and analysis are detailed. Later, RQ. 6.2 is investigated in Section 6.4. Finally, the concluding remarks are presented in Section 6.5.

6.1 Blacklist Sources

In Section 6.1.1 we present the public sources from which we have obtained blacklists. After that, in Section 6.1.2, we present our blacklist peers sources, while in Section 6.1.3 we present the targets we aim to protect, by comparing the target TWOL to both public and peer BadHood blacklists. Finally, in Section 6.1.4, we describe the measurement period and how we preprocess the data before carry out the analysis.

6.1.1 Public Blacklist Sources

There are many blacklists available on the Internet. Some web-sites, like Unified eMail [150], for example, even provide an interface that allows one to query a single IP address against more than two hundred different spam blacklists at once.

We have therefore to choose what blacklists to evaluate. To do it so, we have employed the following criteria:

1. **Monitored application**: the most popular type of blacklists are the spam ones, but we also wanted to observe results for other applications, such as SSH, so we can determine if the same results hold.
2. **Prior usage on both academic and/or Internet security communities**: in order to filter out blacklists of questionable reputation, we only consider blacklists that have been employed by Internet security systems and/or previously investigated by the research community.
3. **Method of access**: we only evaluated blacklists that could be obtained as a bulk single file. The method of access to blacklist varies, but mostly it is provided in a DNS-like fashion [33], in which one queries the blacklist server if a certain IP is blacklisted or not. In our case, blacklist that only provide DNS-like access are disregarded, because, by this mean, we are not able to effectively obtain all the blacklisted IP addresses.

Taking these criteria into account, we have obtained the following blacklists:

- Composite Block List (CBL) : CBL is operated by “a group of computer security, spam and virus professionals, dedicated to developing and maintaining an anti-spam and anti-virus DNSBL (DNS blacklist) of the highest possible quality and reliability, that large organizations can use with confidence” [60]. It lists /32 IP addresses that have reached their spamtraps. The number of traps and their location is not disclosed, but it is distributed over different networks and countries. CBL has been employed in a number of studies, including [62, 63, 64, 65, 66].
- The Passive Spam Block List (PSBL) [75]: as CBL, PSBL also maintains a blacklist of /32 IP addresses that have spammed PSBL distributed traps. They do not provide more details about their infrastructure and the organization behind it. PSBL has been also investigated by our research group in two papers [65, 66].
- DShield.org (Dshield) [151]: DShield is a community shared firewall log system. Volunteers submit their firewall logs from more than 600 contributors, which encompasses more than “500,000 IP addresses (firewalls) in

over 50 countries" [152]. It is maintained by the SANS Institute [153], and contains security logs from many applications. As for Spam blacklists, the blacklists IP addresses are aggregated from many different sources. The DShield dataset has been investigated by the research community in papers including [154, 76, 155].

- SSH Blacklist (SSHBL): SSHBL maintains "19 hosts distributed over Europe, US, Australia, and China, and receives an average of 90 SSH brute-force attacks per day" [156]. As the other public sources, SSHBL also does not provide more information about its infrastructure.

6.1.2 Peer Blacklist Sources

Differently from the public sources, the peer sources for blacklists are those that one might have private agreements to share data with. We have obtained blacklists from the following sources:

- Provider A: Provider A is a major hosting provider in the Netherlands. We have obtained spam blacklists generated after processing their mail filter log files. As specified in the private agreement, we are not allowed to reveal the real name of the provider or disclose more information.

- University of Twente/EWI (UT/EWI) [157]: As for provider A, we have obtained the IP addresses of spammers that have reached the mail server of the Electrical Engineering, Mathematics, and Computer Science Faculty of the University of Twente, in the Netherlands.

- *S*ecurity Incident Response Team of the Brazilian Research Network [144] (CAIS/RNP): Differently from the previous sources, this one is located in Brazil. We have obtained the malicious IP addresses from their mail server log files.

- QuarantaineNet (QNET) [125]: QuarantaineNet is a Dutch company that develops network security and management tools and provides admission control and malware control to their customers. They maintain a honeypot infrastructure with 125 traps distributed mostly over the Netherlands. In this data set, each individual trap is seen as a single blacklist source. The monitored types of attack include SSH, MySQL, and Windows vulnerabilities.

6.1.3 Target Blacklist Sources

As shown in Figure 6.1, to answer the research questions raised in this chapter, we need as input (*i*) public and peer BadHood blacklists, and (*ii*) targets to be protected. These targets should be, preferably, real world production servers, to reflect what is observed on the Internet. In addition, such targets should be application servers or honeypots for at least one of the applications we have obtained blacklists.

In this chapter, we have considered situations in which the network manager of Provider A, UT/EWI, and CAIS/RNP tries to protect his/hers targets. For Provider A, UT/EWI, and CAIS/RNP, the targets are the mail servers behind each source, while for QNET, the targets are the individual honeypots in their honeypot infrastructure.

For each target, we generated a TWOL blacklist based on the history of incoming traffic (in the case of honeypot) or in the application server log files (for the mail servers). Then, we compare it with the public sources and the peer sources, excluding the cases in which both target and peer blacklists are generated from the same host.

6.1.4 Blacklists Collection and Pre-processing

For all blacklists sources, we have chosen a common monitoring period. This is necessary to create the same comparison conditions for the different sources. For all sources, we have collected data for a period of one week, which is long enough to observe a significant number of events. In addition, in this chapter we only compare blacklists that belong to a *same* application (we compare blacklists from multiple applications on Chapter 7).

For the Spam experiments, the monitoring period was from April 19th to April 26th, 2010. For the SSH experiments, the monitoring period was also one week, but from November 11th to 18th, 2011. Even though we have two different monitoring periods, this does not impact our results, since we only compare blacklists belonging to the same time frame.

After obtaining the blacklists, we have to parse them (using customized Java software and Linux shell tools) and convert them to a common BadHood blacklist format, expressed by the tuple: $\langle /24netblock, \#ofHosts\rangle$. In this format, a */24 netblock* refers to /24 IP address of the BadHood and *# of Hosts* refers to the number of observed malicious host for that particular BadHood ($0 <$ # of Hosts < 256)[2].

[2] In [158] and in Appendix E, we show how this can employed as a criteria in filtering spam.

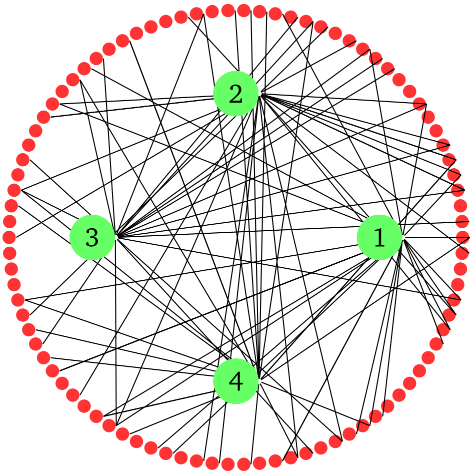


Figure 6.2: BadHoods Attacking Blacklists Sources

In Chapter 3, we have addressed the issue of the BadHood granularity – and how to aggregate BadHoods to smaller prefixes. As we have presented, the aggregation to prefixes smaller than /24 incurs error on the odds of a host belonging to a certain netblock be malicious or not. We have chose therefore /24 in this chapter in order to minimize such errors.

After aggregating the original blacklists to /24 BadHood blacklists, we can proceed with the BadHood blacklists analysis, employing the comparison methods described in Section 6.2.

6.2 BadHood Blacklist Comparison Methods

The idea behind comparing BadHood blacklists is to tell how similar attacks on the Internet are perceived by distinct blacklists sources. Consider as an example the case of Figure 6.2. In this figure, four blacklists sources (1-4) are attacked by different BadHoods (outer circles), being the attacks represented by a line connecting a BadHood to a blacklist source. As can be observed, not all the BadHoods attack the same sources, and some BadHoods attack only one or two targets.

In order to answer RQ. 6.1 and RQ. 6.2, we therefore compare how two different blacklists sources experience the attacks, by answering the following questions:

1. Are two different blacklists sources attacked by the same *number* of BadHoods (that is, the same number of outer circles in Figure 6.2)?
2. Are two different blacklists sources attacked by the *same* BadHoods (that is, exactly the same outer circles in Figure 6.2)?
3. If a BadHood is found attacking two different blacklists sources, is this attack conducted with the same *intensity*, that is, the same number of hosts?

We propose three comparison methods to investigate these questions. We cover each them in the following.

6.2.1 First Method: BadHoods Distribution

The first comparison method focuses on analyzing the BadHoods distribution on each BadHood blacklist. By employing this method to two different blacklists, we can tell if they are attacked by the same number of BadHoods. We therefore compute the following metrics:

- number of source BadHoods (# of distinct /24)
- minimum number of malicious hosts per BadHood (min)
- maximum number of malicious hosts per BadHood (max)
- mean number of malicious hosts per BadHood (mean)
- standard deviation of the number of malicious hosts per BadHood (sdev)

To calculate these metrics, we have employed the software environment for statistical computing R [159].

6.2.2 Second Method: Intersecting BadHoods

The second method focuses on telling if two different blacklists sources are attacked by the same BadHoods. To answer this question, we perform an *intersection* operation between the BadHood set of each source, as shown in Figure 6.3 for two sources ($S1 \cap S2$).

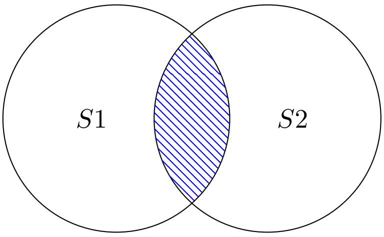


Figure 6.3: Intersecting BadHoods between two Blacklist Sources

The intersection is performed by comparing the each /24 netblock from the BadHood tuple $\langle /24netblock, \#ofHosts\rangle$ from each blacklist source. If there is a match, we then store this information in what we call *BadHood intersection tuple*: $\langle /24netblock, \#ofHostsD_1, \#ofHostsD_2\rangle$.

In the literature, the total number of resulting matching entries is also referred to as *hit count* [76, 160], which we use in the rest of this chapter. We refer to as "irrelevant entries" to the blacklist source 1, in Figure 6.3, to the BadHoods that attacked exclusively source 2 and are not observed by source 1, therefore being irrelevant to source 1. In set notation, this is equal to $S2 - (S2 \cap S1)$.

To single out the matching BadHoods, we have developed a small program using the Java programming language.

6.2.3 Third Method: Correlation

The final comparison method focuses on the following: if a BadHood is found attacking two blacklists sources, does it employ the same number of hosts (intensity)? We answer this question in two different ways:

- Scatter plot: we plot in a point for each BadHood that matches the two compared data sets. The coordinates (x, y) from each point is given by the $\#ofHostsD_1$, $\#ofHostsD_2$ from the BadHood intersecting tuple. The scatter plot allows us to observe, visually, if there is a relationship between the number of hosts from each data set.

- Δ analysis: for each intersecting BadHood (n), we calculate:

  $$\Delta_n = \#ofHostsD_1 - \#ofHostsD_2 \tag{6.1}$$

  Each Δ_n represents the difference between the number of hosts a particular BadHood employed to attack two different targets. We calculate

Δ_n for every matching BadHood between to sources and, after that, we analyze the distribution of delta values.

If most of Δ values are equal to zero, we can conclude that most of matching BadHoods attack both targets using the same number of hosts. If most of Δ values are larger than zero, than the BadHoods attack the first target (D_1) using more hosts than when attacking D_2 (the same analogy applies to negative values).

To calculate the Δ values, we have employed a small program using the Java programming language.

6.3 Public BadHood Blacklists Evaluation

In this section we investigate RQ. 6.1 : "How much can a network administrator rely on *public* BadHood blacklists to protect a target?". As described in Section 6.1.4, we have obtained various blacklists and then aggregated then into /24 BadHoods. After that, we have employed the comparison criteria to answer our research questions. In Section 6.3.1, we presents the results and analysis for the first comparison method (BadHoods Distribution), whereas in Section 6.3.2 we show the results for the second comparison method (BadHoods Intersection), and, finally, in Section 6.3.3 we show the results for the BadHoods correlation.

6.3.1 Method #1: BadHoods Distribution

In this section we present the results of the BadHood distributions. We first present results for Spam and then for SSH blacklists.

Spam BadHoods

When comparing BadHood blacklists from public sources to targets, one could expect that public sources are likely to have a significantly higher number of BadHoods than individual targets, since public sources, typically, aggregate data from multiple hosts to generate a single blacklist. The intuition is that more monitoring hosts increase the chances of observing attacks from different hosts, ultimately increasing the total number of observed BadHoods. Thus, public sources should be able to capture a significantly higher number BadHoods than individual targets.

	Public Sources			Targets		
Metric	**CBL**	**PSBL**	**DShield-Spam**	**Prov. A**	**UT/EWI**	**CAIS/RNP**
# of BadHoods	1,140,005	732,731	37,051	548,866	248,947	34,096
Min(hosts)	1	1	1	1	1	1
Max(hosts)	256	248	110	227	101	28
Mean(hosts)	12.58	4.81	1.23	3.03	1.72	1.08
Sdev	29.32	9.44	1.36	4.81	1.77	0.44

Table 6.1: Spamming BadHoods Distribution

Table 6.1 shows the results for Spam BadHoods. In each column, we have the BadHood blacklists obtained from the public sources and the targets, and in each line we have the results for the metrics described in Section 6.2.1.

Analyzing the number of BadHoods in the table, we see that, with the exception of DShield-Spam[3], the difference between *public BadHood blacklists and Targets blacklists* is relatively not so big. For example, the ratio between CBL and Provider A – data sets that have more entries in each category – is roughly 2. Comparing CBL to the second target in terms of entries (UT/EWI), this ratio is 4.5. The difference is more significant when comparing CAIS/RNP, a small target, to the public sources.

In the case of DShield-Spam, which is a public source, we can observe that it was attacked by a smaller number of BadHoods in comparison to the targets Provider A and UT/EWI. The reason for that might be due to the fact that, for Provider A and UT/EWI, the blacklists are generated based on the mail server logs, while DShield blacklists, on the other hand, are generated based on firewalls/IDS logs [161], which are not specifically designed to detect spam.

We can observe that the ratio between the number of BadHoods observed by public sources and individual targets is only between 2 and 4 (excluding DShield-Spam and CAIS/RNP). An interesting observation is that large data sets, like CBL, have observed 1,140,005 /24 BadHoods out of the roughly 16 million theoretical maximum. This is a revealing number: *it means that at least 6.79% of all /24 neighborhoods on the Internet are involved in Spam.* Provider A, on the other hand, as a single target, has been attacked by 3.2% of all neighborhoods on the Internet.

In addition, we can observe in Table 6.1 that there is a correlation between the number of BadHoods and the mean number of hosts per neighborhood: the more BadHoods a source observes, the more hosts per neighborhood, in average, the source observes (Pearson correlation coefficient [147] of $\rho = 0.92$).

[3] DShield-Spam is a subset of DShield – only attacks on TCP ports 25 and 465.

	Public Sources		Targets				
Metric	**SSH-BL**	**DShield-SSH**	**QNET-1**	**QNET-2**	**QNET-3**	**QNET-4**	**QNET-5**
# of BadHoods	17,637	3,740	110	85	56	31	24
Min(nHost)	1	1	1	1	1	1	1
Max(nHost)	4	113	2	1	1	1	1
Mean(nHost)	1.02	1.09	1.02	1	1	1	1
Sdev	0.19	1.87	0.16	0	0	0	0

Table 6.2: SSH BadHoods Distribution

From the point of view of the network administrator, however, the main concern is that public sources observe more BadHoods than the target itself.

SSH BadHoods

Table 6.2 shows the results for SSH BadHoods. For this application, we have used SSH-BL and DShield-SSH (a subset of DShield dataset, listing IP addresses that have attacked port 22). The targets, on the other hand, were obtained from QNET blacklists. For the monitoring period, 16 of their 125 honeypots have observed SSH attacks. Out of those, we have chosen the top five in terms of number of attacks (QNET-1 – QNET-5) and performed our analysis.

As for Spamming BadHoods, we can observe that for SSH, public sources also observe more BadHoods than targets themselves. However, the difference between public sources and targets, in this case, is quite significant: a ratio of 160.3 when dividing the number of BadHoods of SSH-BL by the number of BadHoods of QNET-1.

In addition, the total number of BadHoods observed for SSH is also very low compared to those observed for Spam. The reason for that is that SSH attacks are far less common than Spam. Therefore, within the same monitoring period of one week, individual targets observe only very few attacks, usually from a single host per /24 BadHood. As expected, public sources detect much more BadHoods than targets. We compare BadHoods from different applications in more details in Chapter 7.

For both Spam and SSH BadHoods, we can conclude that *public sources are more likely to observe more BadHoods than targets* due to the larger number and distribution of their monitoring probes. This behavior, in turn, strengthens the idea that public BadHood blacklists can be employed to protect targets on the Internet.

Targets	Public Sources		
	CBL	PSBL	DShield-Spam
Provider A	98.74% (541,967)	88.03% (483,179)	2.79% (15,336)
UT/EWI	97.95% (243,855)	93.13% (231,856)	4.49% (11,189)
CAIS/RNP	99.12% (33,799)	96.88% (33,034)	6.78% (2,315)

Table 6.3: Spam BadHoods Intersection (% related to the target's BadHoods)

6.3.2 Method # 2: BadHoods Intersection

In Section 6.3.1, we have shown that public sources are likely to observe more BadHoods than the targets. However, we wonder if this implies that the target BadHood blacklists are, consequently, a subset of the public sources BadHood blacklists. This leads us to our second comparison method. First we present the results for Spam BadHoods and then for SSH BadHoods.

Spam BadHoods

For Spam BadHoods, as observed in Table 6.1, CBL BadHood Blacklist comprises 6.79% of the maximum theoretical /24 prefixes, while provider A covers 3.2% of this value. We might expect a significant intersection between these two sources (and the others as well). Otherwise, the implications would be alarming: if both sources are attacked by distinct BadHood sets, that would mean that at least 10% of all /24 neighborhoods on the Internet are involved in malicious activity. Not only that, if the same behavior holds for other blacklist sources, we could end up having the majority of the Internet Neighborhoods (/24) being classified as "bad".

The results the intersection between the Spam Blacklist sources are shown in Table 6.3. In this table, we show the percentage of BadHoods from each target (in rows) captured by the public sources (columns) – ($targets \cap public$) and the absolute number in parenthesis. With exception of DShield-Spam, the public sources, indeed, capture most of the BadHoods that attack individual targets (from 88.03% to 98.74%). From the point of view of the network administrator, this is a very satisfactory result, since it confirms that one can rely upon public spam blacklist sources to protect the network.

Targets	CBL	PSBL	DShield-Spam
Provider A	108.95% (598,038)	45.46% (249,552)	3.95% (21,715)
UT/EWI	359.97% (896,150)	201.19% (500,875)	10.38% (25,862)
CAIS/RNP	3,244.38% (1,106,206)	2052.13% (699,697)	101.87% (34,376)

Table 6.4: Non-Intersecting Spamming BadHoods (% w.r.t. lines)

Irrelevant Entries

Even though public BadHood blacklists can be used to protect a specific target, there is a drawback: even though CBL captures 98.74% of Provider A BadHoods (Table 6.3), CBL has still 598,038 BadHoods that did not spam Provider A mail servers – which is twice the size of Provider A BadHood blacklist, as shown in Table 6.4. Such BadHoods are actually irrelevant to the targets – they represent BadHoods that have not attacked the target in the monitoring period.

This existence of a significant number of irrelevant entries has been suggested by Zhang *et al.*, in which they say that GWOL lists such as PSBL and CBL "have the potential to exhaust the subscribers' firewall filter sets with addresses that will simply never be encountered" [154] This may incur a problem if such BadHood blacklist is employed in memory constrained devices – such as switches and routers (CBL non-matching list in relation to Provider A requires 12MB of storage in plain-text format, but also lookup time plays a role). The same reasoning can be applied for SSH-Badhoods. To address this issue, one could aggregate the BadHood blacklists into smaller prefixes (e.g., /23, /22), as proposed and evaluated in Chapter 3.

Understanding the Intersection

Figure 6.4 shows the intersection between CBL and Provider A. As can be seen, Provider A is almost a subset of CBL (CBL ∩ A, as shown in Table 6.3). However, as shown in Table 6.4, CBL was attacked by 598,038 BadHoods that did not attack Provider A (CBL-(CBL∩ A)). This number represents 52.45% of all BadHoods attacking CBL.

In this subsection, we compare the distribution of the number of hosts of these two subsets (we have carried out the same analysis for UT/EWI and PSBL, and the same conclusions hold). Intuitively, one could think that there would be no particular reasons for this distribution to differ from one another.

As shown in Section 6.2.2, each intersecting BadHood is stored in the follow-

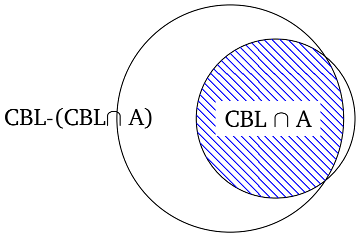


Figure 6.4: CBL and Provider A Intersecting BadHoods

Metric	CBL - Intersection	CBL - Complement
# of BadHoods (/24)	541,967	598,038
Min(nHost)	1	1
Max(nHost)	256	256
Mean(nHost)	22.43	3.66
Sdev	39.17	9.11

Table 6.5: Distribution of malicious hosts

ing tuple: $\langle /24netblock, \#ofHostsD_1, \#ofHostsD_2\rangle$, in wich $HostsD_n$ refers to the number of hosts observed for the compared data sets. For this case, D_1 is the CBL data set, while D_2 is Provider A. Table 6.5 presents the results of the distribution of hosts. In the second column, we present the distribution of the number of hosts per BadHood for CBL (D_1), while in the third column we show the results for complement part of CBL (CBL -(CBL ∩ Provider A)).

Intuitively, one would expect that the BadHoods having more malicious hosts are more likely to attack different targets (CBL and Provider A, in this case), since more malicious hosts would increase the capacity of attack of the BadHood. And this is exaclty what we observe. Analyzing Table 6.5, we can observe that the CBL part that intersects with Provider A has a bigger number of average hosts.

This confirms our intuition: *BadHoods having more malicious hosts attacking a public source are more likely to attack a target*. Figures 6.5 and 6.6 show these results graphically, by showing the distribution of number of hosts per BadHood from both sets.

We believe that the reason why public sources capture, on average, more malicious hosts per BadHood than individual targets is that *they employ multiple targets and aggregate data from those*, increasing the probability to observe

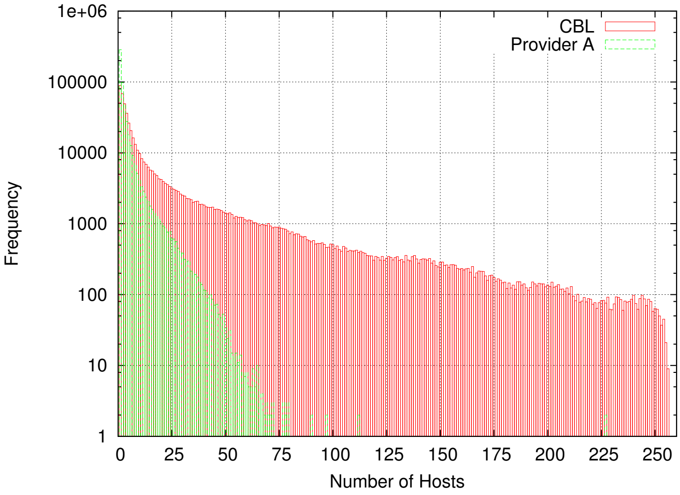


Figure 6.5: Distribution of Hosts for CBL in (CBL ∩ Provider A)

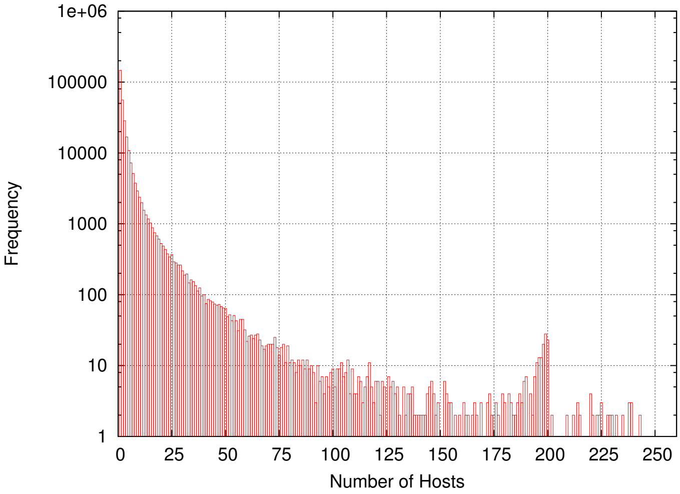


Figure 6.6: Distribution of Hosts for CBL in (CBL - (CBL ∩ Provider A))

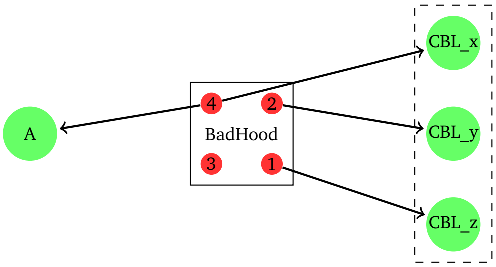


Figure 6.7: Distributed Targets Versus Single Target

Targets	**Public Sources**	
	SSH-BL	**DShield-SSH**
QNET-1	22.72 % (25)	94.54 % (104)
QNET-2	19.58 % (9)	98.82 % (84)
QNET-3	64.28 % (36)	89.28 % (50)
QNET-4	45.16 % (14)	96.77 % (30)
QNET-5	41.66 % (10)	94.83 % (23)

Table 6.6: SSH BadHoods Intersection (% w.r.t. target)

attacks. To illustrate this, consider Figure 6.7. In this figure, target A is attacked by only one host (4) belonging to the particular BadHood, while CBL sources (x, y, and z) are attacked by hosts 1, 2, and 4.

SSH BadHoods

Table 6.6 shows the results for SSH BadHoods. Even though SSH-BL lists much more BadHoods than DShield (17,637 against 3,740, as shown in Table 6.2), surprisingly, more DShield BadHoods intersect with QNET individual targets. This reflects an interesting observation: BadHood blacklists having more entries do not necessarily lead to more matching BadHoods to a particular target. Since we do not know exactly the details of the infrastructure behind each data source, we can only speculate the reasons for this behavior – for example, it might be the case that SSH attackers prefer to choose their targets more strategically than spammers [29].

For both Spam and SSH BadHoods, we can conclude from this comparison

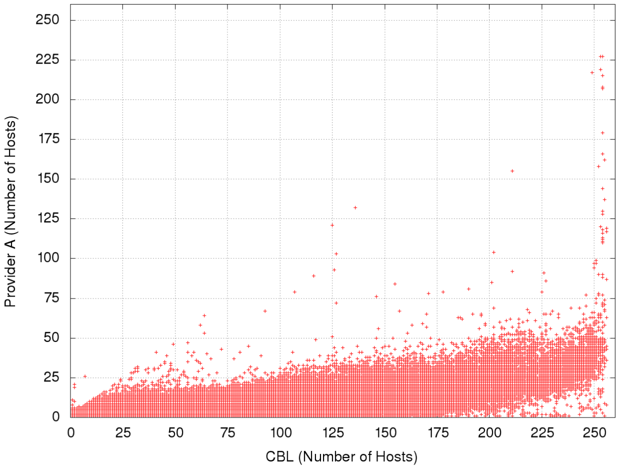


Figure 6.8: Scatter Plot - Provider A - CBL

method that some public sources are likely to capture most of the BadHoods that attack individual targets. The choice of which public source provides the best results cannot always be answered easily. Therefore, we recommend that organizations carry out experiments to assess the quality of the BadHood blacklists they consider to employ in relation to what is observed by the targets.

6.3.3 Method # 3: Correlation

The last comparison method allows us to tell if BadHoods attack different targets with the same intensity, that is, the same number of hosts. To answer this, we first generate a scatter plot using as coordinates the number of malicious hosts from the intersecting BadHoods, as described in Section 6.2.3.

Figure 6.8 shows for all BadHoods in the intersection $I_{CBL \cap ProviderA}$ the number of spamming hosts monitored by CBL (x-axis) and monitored by Provider A (y-axis). As expected, the much larger CBL blacklist sees more hosts for the same BadHood than Provider A. We believe the reasons for that are the same discussed in Section 6.3.2 and shown in Figure 6.7: public sources are more likely to observe more hosts per BadHood due to the large number of monitoring hosts they employ.

To quantify this difference, we calculate the Δ values between the number

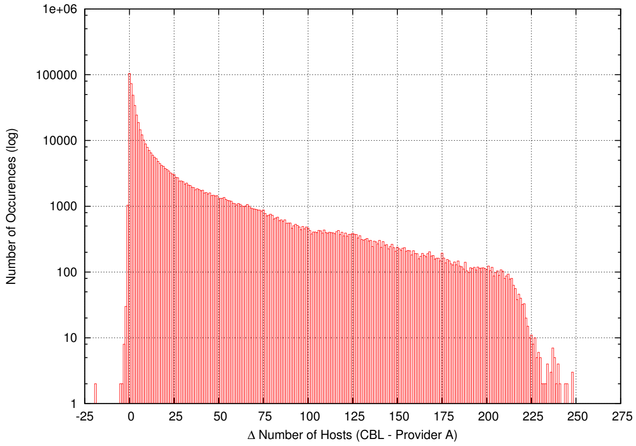


Figure 6.9: Difference Between The Number of Spamming Hosts - CBL and Provider A

of hosts attacking CBL and Provider A from each BadHood. Then, we analyze the distribution of the Δ values, as shown in Figure 6.9. In this figure, on the y axis we show the number of occurrences – that, the number of Δ values, in log scale, while on the x axis we show the values for Δ.

As can be seen, the majority of Δ are positive – meaning that CBL was attacked by more hosts than Provider A (only in few cases Provider A observed more BadHoods – $\Delta < 0$). In addition, we can observe that most of Δ values are located within the interval [0–25] – that is, most of the BadHoods employed between [0–25] more hosts to attack CBL. We have also carried the same analysis PSBL and UT/EWI, and the former results holds.

We have executed the same analysis for SSH BadHoods. However, since the number of intersecting BadHoods is very small in relation to spam (e.g., SSH-QNET1 has 104 BadHoods intersecting with DShield-SSH) we do not present the graphical analysis. Out of the 104 intersecting BadHoods between SSH-QNET1 and DShield-SSH, 95 attacked both with only one host, 2 attacked both with 2 hosts, 6 attacked DShield-SSH with 2 hosts while only one to SSH-QNET, and 1 BadHood attacked DShield with 3 hosts and 2 hosts to SSH-QNET.

Therefore, what we can learn from both Spam and SSH cases is that the way a particular BadHood attacks different targets (or set of) depends on the application. For Spam, it is more likely that BadHoods always attack public sources with more hosts since public sources employ more monitoring hosts than targets typically. This information can be employed, for example, in mail filters, when estimating the probability of a message being spam or not: if the source IP address has been found in a BadHood blacklist generated from public sources, the higher the probability of this message being spam. In the case of SSH, there were few attacks to draw similar conclusions.

6.4 Peer BadHood Blacklists Evaluation

In this section we address RQ. 6.2: "How much can a network administrator rely on *peer* BadHood blacklists to protect a target?". As described in Section 6.1.3, we assume the point of view of a network administrator from Provider A, UT/EWI, CAIS/RNP, and QuarantaineNet.

In this section, we present our analysis for both Spam and SSH BadHoods. In Section 6.4.1, we show the results for the first comparison method, while in Section 6.4.2 and 6.4.3 we show the results for both second and third methods.

6.4.1 Method # 1: BadHoods Distribution

In this section, we show the results for the distribution of Bad Neighborhoods. First we cover the Spam BadHoods and then SSH BadHoods.

Spam BadHoods

When comparing peer BadHood blacklists to targets, one could think that the number of BadHoods would be similar – since, in our case, peer sources and targets are individual hosts. Table 6.7 shows the results for the Spam BadHood blacklists. Analyzing this table, we can observe that there is a significant difference in the number of BadHoods that attack each peer/target (# of BadHoods (/24)): Provider A has observed 2.2 times more BadHoods than UT/EWI, and 16.09 times more than CAIS/RNP.

These result shed some light on the *modus operandi* of spammers: they spam more targets with more users. Even though we do not know the precise number of users per source, we know that Provider A has more e-mail users, followed by UT/EWI, while CAIS/RNP has the smallest number.

	Peer Sources/Targets		
Metric	**Provider A**	**UT/EWI**	**CAIS/RNP**
# of BadHoods (/24)	548,866	248,947	34,096
Min(nHost)	1	1	1
Max(nHost)	227	101	28
Mean(nHost)	3.03	1.72	1.08
Sdev	4.81	1.77	0.44

Table 6.7: Distribution of Malicious Hosts - Peer Sources and Targets

	Peer Sources/Targets				
Metric	**QNET-1**	**QNET-2**	**QNET-3**	**QNET-4**	**QNET-5**
# of BadHoods	110	85	56	31	24
Min(nHost)	1	1	1	1	1
Max(nHost)	2	1	1	1	1
Mean(nHost)	1.02	1	1	1	1
Sdev	0.16	0	0	0	0

Table 6.8: SSH Peer BadHoods Distribution

SSH BadHoods

In Section 6.3.2, we have shown that public SSH BadHood blacklists observe list much more BadHoods than individual targets. Since SSH attack are not that frequent as Spam, one could expect that peer sources and targets consisting of one monitoring host would observe a small but similar number of BadHoods.

This intuition is confirmed by the results shown in Table 6.8. In this table, each Target/Source is an individual honeypot of QuarantaineNet honeynet. Each target (QNET-1 – QNET-5) is located in a different network. As can be seen, the number of attacking BadHoods is small (24–110) and, as shown by the mean value, only one host per BadHood was observing attacking these targets.

6.4.2 Method # 2: BadHoods Intersection

In Section 6.4.1, we have shown that the number of observed BadHoods varies according to the target/peer source. Even though these numbers vary, we still want to know how many (if any) BadHoods were observed by both targets and peer sources ($peer \cap targets$). To answer this question, we determine the

Targets	**Peer Sources**		
	Provider A	**UT/EWI**	**CAIS/RNP**
Provider A	–	41.68% (228,777)	5.8% (32,347)
UT/EWI	91.89% (228,777)	–	9.3% (23,316)
CAIS/RNP	94.8% (32,347)	68.3% (23,316)	–

Table 6.9: Peer Spam BadHoods Intersection – % in relation to the target's BadHood blacklist

intersection between the BadHoods observed by the two data sets. First we present the results for Spam BadHoods, and then for SSH BadHoods.

Spam BadHoods

Table 6.9 shows the results for the Spam BadHoods. In this table, we show the percentage of the target's BadHoods (rows) that were captured by the peer sources (columns), while the absolute number is shown between parentheses. For example, if Provider A were to employ BadHood blacklist from UT/EWI, that would be enough to detect only 41.68% of the Spamming BadHoods it observes.

Analyzing Table 6.9, we can observe that the best results are obtained only when UT/EWI and CAIS/RNP employ Provider A's BadHood blacklist. The reason for that is also related to the number of entries each peer source observes: as shown in Table 6.7, Provider A has observed 2.2 and 16.09 times more BadHoods than UT/EWI and CAIS/RNP, respectively.

Irrelevant Entries

Table 6.10 presents the number of BadHoods that are irrelevant to the targets – that is, they have not been observed attacking the targets, but only the peer sources. If the target UT/EWI were to use Provider A's BadHood blacklist, it would be able to match 91.89% of the BadHoods (as shown in Table 6.9), but UT/EWI would not observe 319,889 BadHoods that only attacked Provider A, as shown in Table 6.10, which is equal to 128,49% of Provider A's own observed blacklist. The same reasoning applies to the other targets.

Target/Peer Source	Provider A	UT/EWI	CAIS/RNP
Provider A	–	20,710 (3.77%)	1,749 (0.31%)
UT/EWI	319,889 (128,49%)	–	10,780 (4.33%)
CAIS/RNP	516,519 (1,514.89%)	225,631 (661,75%)	–

Table 6.10: Non-Intersecting Spamming BadHoods - Peer Sources – % in relation to the target's BadHood blacklist

	Peer Sources				
Target	QNET-1	QNET-2	QNET-3	QNET-4	QNET-5
QNET-1	–	0	3	1	1
QNET-2		–	0	0	0
QNET-3	0	0	–	0	0
QNET-4	0	0	0	–	0
QNET-5	0	0	0	0	–

Table 6.11: Peer SSH Peer BadHoods Intersection

SSH BadHoods

Table 6.11 shows the results for the peer sources for SSH applications. For the monitored sources, the number of matching BadHoods is very small if not zero. For those SSH peer sources, we can conclude that it is not worth to employ other peer's blacklists.

Taking these results in consideration, we can conclude that for some cases it might be worth to employ peer sources to generate BadHood blacklists, taking into account the number of observed BadHoods per target and application.

6.4.3 Method # 3: Correlation

This comparison method allows us to tell whether a peer source and a target are attacked with the same intensity – that is, the same number of malicious hosts per matching BadHood. To perform this, we first generate a scatter plot using as coordinates the number of malicious hosts from the intersecting BadHoods, as described in Section 6.2.3.

The scatter plot for Spam BadHoods is shown in Figure 6.10. Each point represents an intersecting BadHood, where the tuple (x, y) represents the number of hosts used by then BadHood to attack Provide A and UT/EWI, respectively. The green line (Ratio=1) shows where the number of hosts is equal for both

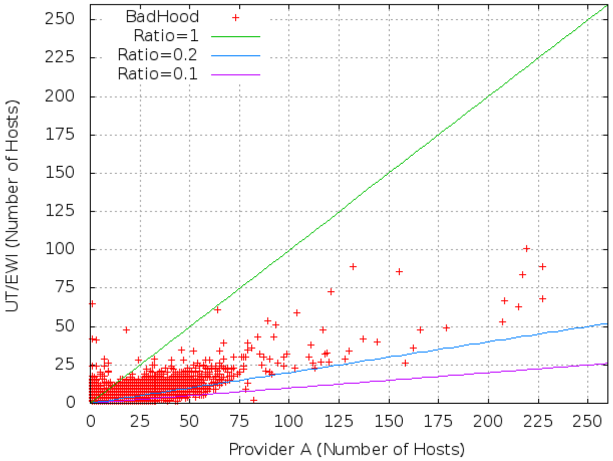


Figure 6.10: Scatter Plot - Provider A - UT/EWI

datasets. As can be seen, most of the BadHoods have attacked Provider A using more hosts than when they attacked UT/EWI (all points below the green line). This shows that not only more BadHoods attack targets having more valid mail users (as shown in Section 6.4.3), *but that they use more hosts to carry out the attacks.*

Figure 6.11 shows the distribution of the difference values (Δ) between the number of hosts attacking Provider A and UT/EWI, for the intersecting BadHoods. As can be see, most of difference values are located within the interval [0–25] – that is, most of the BadHoods employed at most 25 more hosts to attack Provider A than UT/EWI, and the number of occurrences decreases as Δ values increase. There are some few cases in which UT/EWI has observed more attacking hosts than Provider A, but they represent a minority of the cases.

In the case of SSH BadHoods for peer sources, the number of intersecting BadHoods is very small, therefore we have not proceeded with the analysis.

Taking the results from both Spam and SSH, we can conclude that, depending on the application, peer sources observe different number of attacking BadHoods, as we have observed when comparing Public source Blacklists BadHoods to individual targets.

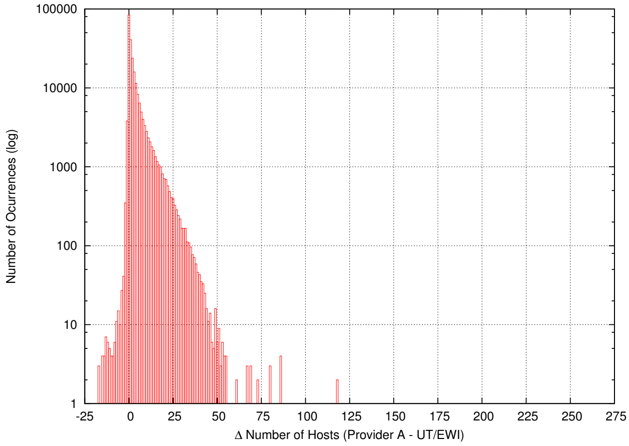


Figure 6.11: Difference Between The Number of Spamming Hosts - Provider A and UT/EWI

6.5 Conclusions

In this chapter we have investigated the following research question: "*how much a network administrator can rely on BadHood blacklists obtained from other sources to protect a target?*". The answer to this question is that, indeed, *network administrators can rely upon public or peer sources to protect a target and have a high BadHood hit count rate*. However, not all BadHood blacklists sources can be relied upon. Therefore, we recommend network administrators to carry out an evaluation (as presented in this chapter) comparing to public and peer BadHood blacklists.

In RQ. 6.1, we asked "*how much a network administrator can rely on BadHood blacklists obtained from public sources to protect a target?*". We have evaluated both Spam and SSH BadHoods, and the results were the following:

- **Spam:** We observe a significant overlap between the public BadHood blacklists. This is particularly true for large, public blacklists such as CBL and PSBL, which cover up to 99% of the BadHoods spamming the targets. Our research also shows that this intersection captures the most aggressive BadHoods (in terms of number of malicious hosts). However, we also found that large blacklists also contain a large number of "irrelevant" entries, which are BadHoods not observed by the target itself. Such extra entries might impose a burden if used in resource-restricted security mechanisms, such as firewalls. These results were presented in Section 6.3.

- **SSH:** public SSH blacklist sources are more likely to observe much more BadHoods than individual targets (up to 734.8 times more), and one of the public sources we have evaluated was able to capture up to 98.82% of BadHoods observed by targets (as shown in Section 6.3).

In RQ 6.2, we investigated "*how much a network administrator can rely on BadHood blacklists obtained from peer sources to protect a target?*'. We have also evaluated Spam and SSH BadHood blacklists. We found:

- **Spam:** not all peer sources observe more BadHoods than individual targets – which is condition required to employ BadHood blacklist from other sources. For the cases in which the peer source observes more BadHoods, we found that peer sources we able to capture up to 94.8% of BadHoods observed by individual targets, as shown In Section 6.4.
- **SSH:** the monitored targets have observed a small number of BadHoods, and only in 3 cases (out of 16) there were BadHood matches, as we covered in Section 6.4.

Comparing the results obtained from above question, we can conclude that public sources can be relied upon to protect individual targets, for both Spam and SSH. Peer sources, on the other hand, can be used for Spam only if the peer source observes more attacks than the target itself, whereas for SSH we found that it is not worthy using peer sources to protect targets. For both cases (public and peer sources), we recommend network administrators to evaluate the Blacklist sources before using them.

It has certainly been true in the past that what we call intelligence and scientific discovery have conveyed a survival advantage. It is not so clear that this is still the case: our scientific discoveries may well destroy us all [...].

Stephen Hawking , 1988
In: A Brief History of Time

CHAPTER 7

Bad Neighborhood Blacklists from Different Applications

In this chapter, our goal is to determine if different types of Internet attacks are originated by the *same set of Internet BadHoods*. The motivation for conducting this study is similar to the motivation of Chapter 6: to avoid carrying out unnecessary network measurements. If we find that the same set of Internet BadHoods are responsible for different type of attacks (e.g, spam, SSH scans, etc), a network administrator could avoid having to generate application-tailored BadHood blacklists, and employ the currently available ones to protect targets running different applications. For example, one could employ a Spam BadHood blacklists to also protect from SSH or Windows Shares attacks.

We have seen in Chapter 4 and in Chapter 6 that the size of BadHood blacklists varies according to the application (e.g., spam blacklist have usually more entries than SSH and phishing counterparts). However, whether we can find a significant number of BadHoods in different blacklists is still unclear – which is the object of study in this chapter.

Taking this into account, we raise the following research question: "*Are the same BadHoods responsible for carrying out attacks to different applications on the Internet?*".

To answer this research question, we have first chosen data sets containing IP addresses found carrying out attacks employing different applications. After choosing the data sets, we have carried out the measurements, and obtained a daily snapshot of each data source for a week period. Next, we have generated individual blacklists containing the IP addresses of the attackers, and aggregated it into /24 BadHoods. We have chosen /24 because this is the prefix that incurs less aggregation error (as discussed in Chapter 3), and it is also the smallest prefix that can be "routed" on the Internet [78]. After that, we will compare the generated BadHood Blacklists employing the same methodology we have used in Chapter 6 when we compared BadHood blacklist from different sources.

This chapter is further organized as follows. In Section 7.1, we present the application-specific data sets we have used to generate the BadHood blacklists for the various evaluated applications. In Section 7.2, we evaluate the data sets employing the comparison methods described in Section 6.2. Finally, in Section 7.3, we present our conclusions for this chapter.

7.1 Blacklist Sources

In this section, we present the data sets that are specific for spam (CBL: Section 7.1.1), phishing (Phishtank, Section 7.1.2), and firewalls logs having multiple applications (DShield, in Section 7.1.3).

For the three data sets, we have collected data for a one week period (November 11th to 18th, 2011). Then, we have generated a single list of /32 IP addresses for each data set. Subsequently, each blacklist was aggregated into a /24 BadHood blacklists.

For choosing what data sets to evaluate, we have employed the same criteria as for Chapter 6, as described in Section 6.1.1: (*i*) monitored applications, (*ii*) prior usage in both academic or Internet security communities, and (*iii*) method of access. We have focused on data sets that would provide lists for different applications, that have been used in different research works, and that can be obtained in bulk. Next we provide more detail about each data set.

7.1.1 Composite Block List (CBL)

As described in Chapter 6, Composite Block List (CBL) is spam blacklist, which is operated by “a group of computer security, spam and virus professionals, dedicated to developing and maintaining an anti-spam and anti-virus DNSBL of the highest possible quality and reliability, that large organizations can use with confidence” [60]. It lists /32 IP addresses that have reached their spamtraps. The number of traps and their location is not disclosed, but it is distributed over different networks and countries. CBL has been employed in a number studies, including [62, 63, 64, 65, 66].

Listing 2 Sample of DShield Log Files

```
Source IP       Port  Proto     Occur.  Targets   First Seen   Last Seen
062.004.071.237 8080    6       78261    78230     02:48:30     22:59:01
063.076.053.141 8080    6       77402    76985     06:37:10     18:31:29
221.011.023.124 3306    6       66622    66604     01:09:31     19:46:32
074.082.166.196 135     6       65307    64923     04:56:15     23:54:11
061.147.088.223 1433    6       64802    64798     06:30:18     07:20:51
121.008.251.046 22      6       126766   64532     00:27:09     00:32:36
```

7.1.2 Phishtank

As shown in Chapter 4, we have obtained data from Phishtank, which is an open community web site in which anyone can "submit, verify, and track phishing websites" [106]. It provides a blacklist of URLs that contain forged websites. Since we need IP addresses instead of URLs to proceed with our analysis, we have obtained this blacklist and resolved all the URLs to IP addresses using Google Public DNS [107]. In case of a URL was resolved to multiple IP addresses, we have considered all of them.

7.1.3 DShield

As described in Chapter 6, DShield [151] is a community shared firewall log system. Volunteers submit their firewall logs from more than 600 contributors, which encompass more than "500,000 IP addresses (firewalls) in over 50 countries" [152]. It is maintained by the SANS Institute [153], and contains security logs from many applications. The DShield dataset has been investigated by the research community in several research works, including [76, 154, 155].

An additional advantage of using the DShield dataset is that it provides log files for attacks belonging to many applications — differently, for example, from blacklists like CBL [60], that only list spamming IP addresses.

Listing 2 shows a sample of a DShield log file (field names were changed to fit in the page). As can be seen, the file is aggregated over the source IP address of the attacker (`Source IP`) and destination port (`Port`). For example, the first IP address in the list (62.4.71.237) has employed TCP (Proto = 6, from IANA Internet protocol numbers [162] to attack port 8080 for 78,261 times (`Occur.`), on 78,230 distinct targets (`Targets`). The time information is also included, in GMT. The date can be inferred from the file's name (DShield provides one file per day).

Metric	Number
# of distinct /32 IPs/Proto/Port	4,978,729
# of distinct /32 IPs	2,888,099
# of distinct /24 BadHoods/Proto/Port	2,915,394
# of distinct /24 BadHoods	1,259,916
# of distinct Applications (Proto/Port)	125,383

Table 7.1: D-Shield Data Set – Breaking Down

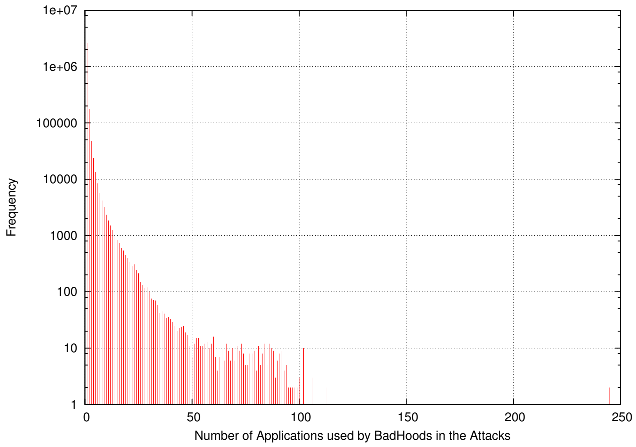
We have then filtered the Dshield files to keep only the information we needed (`Source IP, Port, Proto`). After that, we have filtered out repeated entries and generated unique entries (`Source IP, Port Number, Protocol`). For the monitoring period, we found 4,978,729 different entries.

Table 7.1 provides an overview of the DShield dataset. As can be seen, we have found more than 2.8M single /32 IP addresses carrying out attacks. On average, these individual IP addresses have misused 1.72 different applications ($\frac{row_1}{row_2}$). On the other hand, /24 BadHoods were found misusing 2.31 applications, in average, while having 2.29 malicious ($\frac{row_2}{row_4}$) IP addresses per /24.

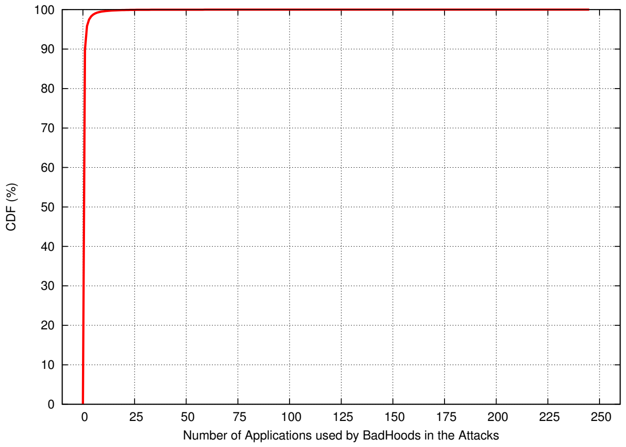
Figure 7.1(a) shows the distribution of the number of *distinct* BadHoods/Proto/Port, while Figure 7.1(b) shows its respective CDF. As can be seen, the vast majority of /24 BadHoods (2,620,153, or 89%) have been observed attacking using a single application only. These results from DShield suggest that *the majority of BadHood are application-specific*; however, in the next section, we will investigate if these findings still hold when comparing BadHoods from different data sources.

Since DShield provides data for more than 100 K types of applications, we chose a subset of these for our analysis. We have ranked the most frequently attacked applications (`Port` and `Proto` fields) in terms of number of attacking IP addresses. Table 7.2 shows the top 20 applications in this list, including their description. As can be seen, most of the attacking IP addresses target Microsoft-DS active directory (Port 445).

In addition, many entries did no list any protocol and others used high port numbers (unassigned). Therefore, we have focused only on attacks on the "well-know ports" (port number < 1024, according to IANA terminology and list [163]) that have the protocol field (`Proto`) different from `NULL`. By filtering out such entries, we filter out potential false positive entries found in DShield data set, and focus on the most repeated ones. Table 7.3 shows the Top 10 ports according to these criteria.



(a) Frequency



(b) CDF

Figure 7.1: DShield /24 BadHoods Distribution According to Application/Protocol

# of /32 IPs	Dst Port	TCP/UDP	Description
553,139	445	TCP	Microsoft-DS AD, Windows/Samba shares
153,535	**5559**	**UDP**	**unassigned**
144,701	6987	TCP	unassigned, Bittorrent (likely)
122,401	445	NULL	–
114,626	50266	UDP	unassigned
102,261	80	NULL	–
101,454	6881	NULL	unassigned, Bittorrent (likely)
94,345	3389	TCP	Microsoft Terminal Server
86,447	30211	UDP	unassigned
85,042	35512	NULL	unassigned
66,946	7827	UDP	unassigned
55,434	5644	UDP	unassigned
44,946	12253	UDP	unassigned
40,498	25	TCP	Simple Mail Transfer Protocol (SMTP)
35,584	28984	TCP	unassigned
32,043	35691	NULL	–
31,255	30806	NULL	–
31,076	51413	NULL	–
29,779	11941	TCP	unassigned
28,293	443	TCP	`https`

Table 7.2: Top 20 Ports - DShield

From the top 10 ports shown in Table 7.3, we have chosen the top 5 ports to carry out our experiments (excluding Telnet[1]), plus a high port having most of the attacks (5559) from Table 7.2. Therefore, six ports from DShield were chosen: TCP 445 (T-445), UDP 5559 (U-5559), TCP 25 (T-25), TCP 443 (T-443), TCP 80 (T-80), and UDP 53 (U-53).

7.2 Experimental Evaluation

In this section we present the results of our evaluation, using the methods described in Chapter 6. In Section 7.2.1, we evaluate the BadHood blacklists according to the their distribution, as described in Section 6.2.1. Then, in Section 7.2.2, we show the BadHoods Intersection, as covered in Section 6.2.2,

[1] We have deliberately excluded Telnet since this application should have already been phased out and replaced by SSH. In addition, it does not make much sense protecting an application that is intrinsically vulnerable, since no encryption is employed and credentials are transmitted in clear. See more in `http://www.networkworld.com/news/2011/012711-hackers-turn-back-the-clock.html`

# of /32 IPs	Dst Port	TCP/UDP	Description
553,139	**445**	**TCP**	**Microsoft-DS AD, Windows/Samba shares**
40,498	**25**	**TCP**	**Simple Mail Transfer Protocol (SMTP)**
28,293	**443**	**TCP**	**https**
16,624	**80**	**TCP**	**Hypertext Transfer Protocol (http)**
11,164	23	TCP	Telnet
8,979	**53**	**UDP**	**Domain Name System (DNS)**
4,517	161	UDP	Simple network management protocol (SNMP)
4,469	137	UDP	NetBIOS Session Service
3,722	22	TCP	Secure Shell (SSH)
3,401	80	UDP	unassigned

Table 7.3: Top 10 Ports < 1024, Protocol “Not Null”

	spam	SMB	–	–	https	http	DNS	phishing
Metric	**CBL**	**T-445**	**U-5559**	**T-25**	**T-443**	**T-80**	**U-53**	**Phishtank**
BadHoods	969,581	240,085	43,233	33,340	24,880	14,484	4,442	1,762
Min(nHost)	1	1	1	1	1	1	1	1
Mean(nHost)	10.84	2.30	3.55	1.21	1.13	1.14	2.02	1.25
Max(nHost)	256	102	113	110	58	121	245	16
Sdev	26.38	2.77	6.67	1.32	1.14	2.31	7.29	1.13

Table 7.4: BadHoods Statistics for Different Applications

and finally, in Section 7.2.3, we show the results and analysis for BadHoods correlation, covered in Section 6.2.3.

7.2.1 BadHoods Distribution

Similar to Section 6.3.1, we start by comparing the number and distribution of malicious hosts over the IP address space for different BadHood blacklists.

Table 7.4 presents the results. In this table, we show, for each application, the number of observed BadHoods and the statistics on the number of hosts per BadHood. Analyzing this table, we can see that the number of observed BadHoods changes considerably according to the application considered. For example, CBL (a spam dataset) has exhibited 550 times more BadHoods than Phishthank (phishing dataset). In addition, even when comparing BadHoods from the the same source (DShield), we can observe port 445 BadHoods (T-445, Windows Shares) were 16.57 times more frequent than http BadHoods (T-80).

In addition, for all the evaluated applications, we observe a correlation between the number of BadHoods and the mean number of hosts per neighborhood: the more BadHoods a source observes, the more hosts per neighborhood, in average, it observes. We found a Pearson correlation coefficient of $\rho = 0.95$ [147]) for the eight applications.

From these results, we conclude that the number of attacking BadHoods varies according to the application being exploited. In the next section we will investigate if there is an overlap between the various BadHoods blacklists.

7.2.2 BadHoods Intersection

This subsection focuses on revealing the percentage of intersecting BadHoods between two BadHood blacklists.

Table 7.5 shows the results. Note that, for two blacklists, we only compare the one which has observed less hosts to the one which has observed more hosts, since we want to compare what is the intersection of a smaller BadHood blacklists to a bigger one. In Table 7.5, we show the number of BadHoods that were found intersecting between two applications BadHoods blacklists; the percentage values refer to the total number of matching BadHoods divided by the number of entries observed by the line source. As an example, consider the second row and second column. It is to be interpreted as follows: of all BadHoods that have attacked using UDP Port 5559 (U-5559), 29.8% were also found attacking TCP 445 application (T-445).

Analyzing this table, we can observe that, for only two cases (U-5559 and T-25, both against CBL) we have an intersection rate above 90% (in relation to U-5559 and T-25 data sets sizes). That means that more than 90% BadHoods that carry out attacks on port 5559 and on port 25 also carry out spam attacks (we would expect such a high rate for T-25, since it monitors the default SMTP port), however, port UDP 5559 is not assigned by IANA, which means no official application is supposed to run in this port.

However, for the rest of the applications, we can see the matching rate between any two data sets is below 51 %, being the majority below 30%. These are very low values if one intends to use BadHood blacklists from one application to secure another application. These numbers also confirm our expectations from what we have found in Section 7.1.3, in which we have broken down the DShield data set according to the application, and found that most of the BadHoods (89.8%) have carried out attacks employing only a single application.

Therefore, what we can conclude is that, for most of the cases, the BadHoods

Target/BadHood SRC	CBL	T-445	U-5559	T-25	T-443	T-80	U-53
T-445	69.4 % (166,789)						
U-5559	91.7% (39,660)	29.8% (12,894)					
T-25	93.0% (31,012)	26.7% (8,928)	19.5% (6,504)				
T-443	51.02% (12,694)	18.2% (4,547)	3.8% (950)	3.5% (884)			
T-80	32.1% (4,658)	11.2 % (1,623)	2.5% (375)	2.6% (387)	9.5% (1,377)		
U-53	28.5% (1,269)	8.2% (368)	3.89% (177)	6.9% (307)	1.8% (84)	3.1% (140)	
Phishtank	23.43% (413)	0.03% (54)	0.01% (2)	2.4% (43)	1.7% (22)	1.7% (23)	0.2% (5)

Table 7.5: BadHoods Intersection for Different Applications (w.r.t. the number of BadHoods of the columns datasets)

attacking two different applications differ, *and therefore it is necessary to carry out measurements for distinct applications*.

7.2.3 Correlation

In this last subsection we compare the number of attacking hosts per intersecting BadHood blacklists. To do this, we have chosen a subset of application blacklists which have presented a intersecting rate of at least 90%, as shown in Table 7.5.

Figure 7.2(a) shows the scatter between CBL and U-5559. In this figure, each point (x, y) represents an intersecting BadHood, while the x coordinate refers to the number of hosts that the particular BadHood has used to attack CBL, while y refers to the number of hosts employed to attack U-5559. As can be seen the majority of points are below the green line (which would indicate that the they employ a similar number of hosts). That means that most of the BadHoods observed in the intersection have attacked CBL using more hosts in comparison to U-5559 – which is confirmed by the Δ values (see Section 6.2.3) analysis shown in Figure 7.2(b). In this figure, we can observe the frequency of the number of hosts of intersecting BadHoods between CBL and U-5559. Figure 7.2(c) shows that the majority of intersecting BadHoods have attacked CBL using a larger number of hosts than U-5559 ($\Delta > 0$).

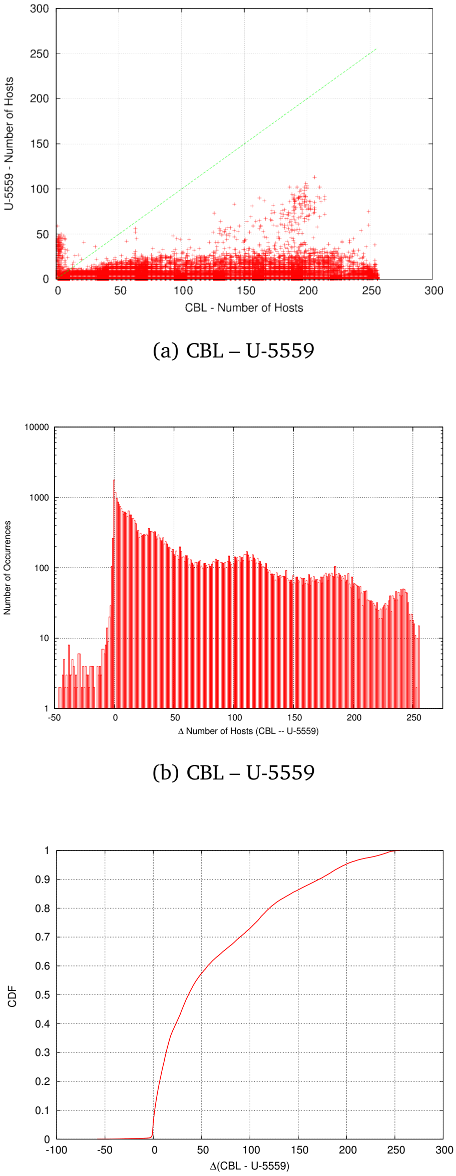
These results may also be influenced by the number of monitored IP addresses each data set source has, since increasing the number of monitored IPs may increase the odds of being attacked by more hosts. However, this information is not provided by those datasets: CBL does not disclose the number of IP addresses they monitor, and DShield provide aggregated information in relation to the source IP addresses (attacking IPs).

The same conclusions can be obtained from analyzing the case of CBL and T-25 – as can be seen in Figures 7.3(a), 7.3(b), and 7.3(c).

7.3 Conclusions

In this chapter we have compared BadHood blacklists obtained from various applications. The goal was to determine if different types of Internet attacks (i.e., using different applications) were carried out by the same set of BadHoods.

To answer this question, we have obtained representative data sets containing offending IP addresses from various applications, for the same monitoring

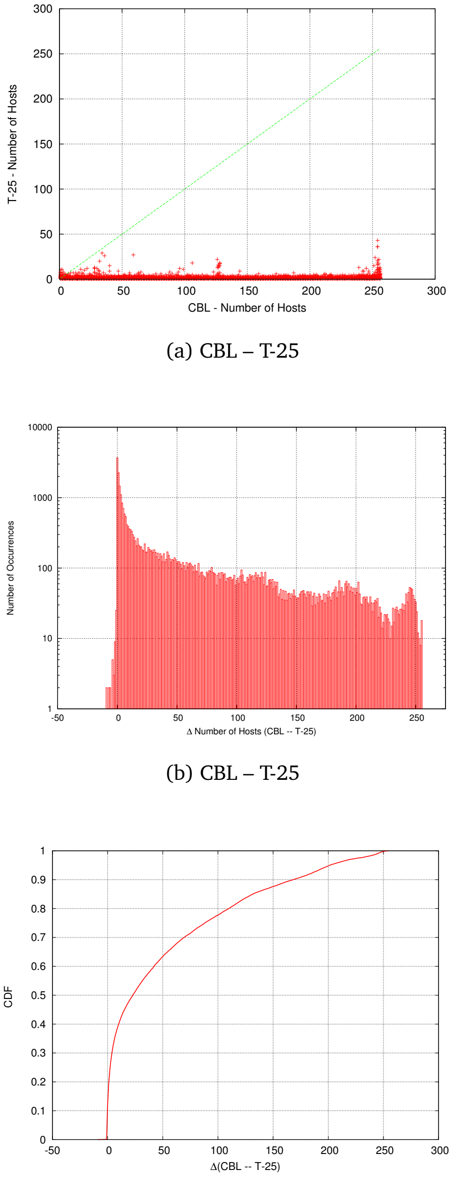


(a) CBL – U-5559

(b) CBL – U-5559

(c) CBL – U-5559

Figure 7.2: Analysis for CBL – U-5559



(a) CBL – T-25

(b) CBL – T-25

(c) CBL – T-25

Figure 7.3: Analysis for CBL – T-25

period. After that, we have generated BadHood blacklists (/24 prefixes) and compared them using the methodology describe in Chapter 6.

In our analysis, we found that the number of offending BadHoods varies significantly according to the application (by a factor of up to 550). The explanation for this variation lies with the specifics of the application being exploited for the attack and its underlying business model. As shown by McCoy *et al.* [18], online sales of counterfeit/illegal pharmaceutical products is heavily based on spam, and it is a market far from being saturated. By analyzing leaked data from two pharmaceutical operations, they have shown that, on average, 1,500 and 3,500 new clients are attracted by spam campaigns every week – which makes spam profitable, and therefore, we can expect it to continue. In contrast, phishing attacks have a different business model that does not rely upon a massive number of individual IP addresses as spam. The difference between the application's business model is therefore reflected in their respective BadHoods.

Moreover, our results have shown that for only two cases (out of 49), we found two BadHood blacklists having an intersecting rate above 90% (w.r.t. the smallest blacklist). The cases were when we compared CBL (a Spam blacklist) to DShield's TCP Port 25 attacks (T-25) and DShield's UDP Port 5559 (U-5559) attacks. For the first case,this could be explained, since both are related to Spam. Since UDP port 5559 is not registered, we cannot tell if there is any relation between this port and spam activity. Regardless, the number of BadHoods (excluding CBL) were below 241,000 – which is equal to 1.4% of the maximum theoretical /24 BadHoods of the IPv4 addressing space. For these two particular cases, we also found that CBL has been attacked more much often by a larger number of hosts than DShield's counterparts, when BadHoods attacking both sources were compared.

The implications of our results is that Internet BadHoods should be application tailored – which supports the BadHood definition we have introduced in Chapter 1 and the findings in Chapter 4. Therefore, we can conclude that a network administrator should employ application-specific BadHood blacklists.

Forgotten were the elementary rules of logic, that
extraordinary claims require extraordinary evidence and
that what can be asserted without evidence can also be
dismissed without evidence.

Christopher Hitchens , 2003
In: “Mommie Dearest”, Slate Magazine

CHAPTER 8

Bad Neighborhoods Temporal Attack Strategies

THE definition of Bad Neighborhood shown in Chapter 1 defines Bad Neighborhoods (BadHoods) according to the number of observed malicious hosts, application exploited, and period of time they are active. In this chapter, we focus on the last of these three parts of the definition. The goal of this chapter is to reveal the temporal attack strategies employed by Internet Bad Neighborhoods. By temporal attack strategies we refer to *when* BadHoods attacks a target, how many attack, and *wether hey strike again*, and if so, *when*.

The motivation for carrying out this analysis is twofold: by scrutinizing temporal attacking strategies employed by BadHoods, a network administrator can determine *how often* BadHoods blacklists should be updated in order to better protect targets. Most important, any observed temporal pattern can be *employed in the design of models and methods* to counterattack attacks (or avoid damage from the attacks).

Therefore, in this chapter we investigate three research questions, as follows:

- RQ 8.1: What is the daily variation in the number of observed BadHoods? The answer to this question allows us to observe how dynamic BadHoods are with relation to a particular data set.

- RQ 8.2: Given a certain monitoring period, in how many days a BadHood is observed carrying out attacks? And on what days do these attacks occur? The answer to this question will show if a network administrator can expect BadHoods to attack again and when that is expected to happen.

- RQ 8.3: Given a single monitoring day, how many BadHoods that carried out attacks in this day can be traced back to previous days (recurrent)?

And how long does it take for most of them to strike again? The answer can be used to develop models that predict attacks from BadHoods, based on historical past.

The rest of this chapter is divided as follows. In Section 8.1, we cover the data sets used in this chapter. Next, in Section 8.2, we address RQ 8.1, while in Section 8.3 we address RQ 8.2. After that, we address RQ 8.3 in Section 8.4 and the conclusions are presented in Section 8.5.

8.1 Evaluated Datasets

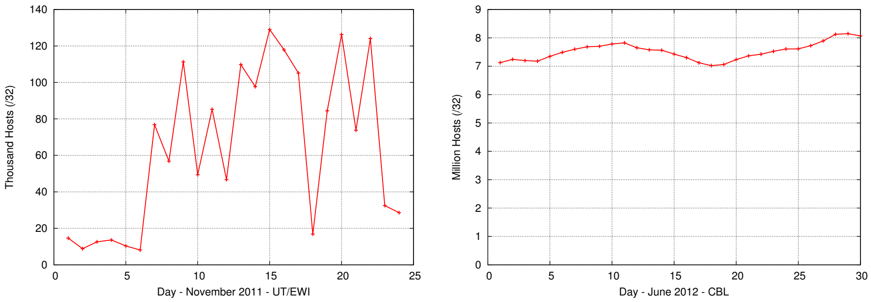
In order to have fair comparison conditions, we have considered datasets obtained for the same monitoring periods, as in the previous chapters. We have therefore considered two monitoring periods:

- April 2010: from 19th to the 26th (8 days)
- November 2011: from November 11th to the 17th (7 days).

For the monitoring period, we have collected data from three data sources. We summarize the data sets here, while more details were previously addressed and can be found in Sections 5.2 and 6.1:

- CBL Spam blacklist (CBL) [60] (See also in Section 3.5.1).
- UT/EWI (UT/EWI): obtained from analyzing the spam filter logs from the mail servers of the Computer Science department of the University of Twente.
- DShield data set [151]: DShield is a community shared firewall log system (See more in Section 7.1.3). We have chosen to focus on two of the most commonly attacked applications:
  - TCP 445 (T-445): Microsoft-DS Active Directory and/or Windows shares [164].
  - TCP 3339 (T-3389): Microsoft Terminal Server (RDP) [165].

After obtained the data sets, we have, for each day and data set, generated a /24 BadHood blacklist. These BadHood blacklists were then employed to answer the research questions presented in the introduction.



(a) Number of /32 Hosts – UT/EWI – November 2011

(b) Number of /32 Hosts – CBL – June 2012

Figure 8.1: Daily Variations (/32 Hosts)

8.2 Daily Number of Bad Neighborhoods

In this section we investigate whether the number of Internet Bad Neighborhoods that a target observes changes for different days (RQ 8.1).

We have several reasons to expect that the BadHoods distribution over different days is far from being static. The main one is a consequence of the behavior that individual hosts (/32) exhibit, trying to be as stealthy as possible (e.g., spamming only once a server and not coming back, – as discussed in Appendix B). For example, Figure 8.1(a) shows the daily number of unique spammers (/32 hosts) for UT/EWI throughout November 2011. As can be seen the values range from less than 20K to more than 120K individual hosts per day, over a period of 24 days. Figure 8.1(b) shows the daily variations for the CBL [60] blacklist, which also exhibits a variation for the monitored days (please notice the difference between the y axis scale of both figures).

Another reason for expecting that the BadHood distribution changes over time is that DNS Blacklists [33], such as CBL [60] and PSBL [75], which contain many malicious /32 IP addresses, have to be constantly updated in order to keep up with the dynamics of individual hosts and be effective in the mail filtering.

Taking these into account, we then proceed to the analysis of the datasets employed in this chapter. Table 8.1 presents the daily number of BadHoods, for each individual dataset. As we expected, for all data sets, the number of BadHoods changes on a daily basis. In addition to that, we observe that:

April 2010				
Day/Dataset	**CBL**	**UT/EWI**	**T-445**	**T-3389**
1st Day	955,036	66,759	141,527	752
2nd Day	958,258	58,344	146,051	817
3rd Day	954,019	61,804	143,379	731
4th Day	954,522	60,045	142,531	759
5th Day	949,167	46,892	142,105	834
6th Day	957,583	48,828	142,422	832
7th Day	961,573	45,351	138,426	773
8th Day	956,410	59,739	141,512	895
Max. Variation:	$\sim$ 1%	$\sim$ 47%	$\sim$5%	$\sim$22%
November 2011				
Day/Dataset	**CBL**	**UT/EWI**	**T-445**	**T-3389**
1st Day	812,217	56,030	79,258	818
2nd Day	809,268	32,612	77,286	25,228
3rd Day	798,345	62,769	76,210	25,331
4th Day	792,098	64,452	77,003	33,319
5th Day	795,763	73,615	78,004	31,065
6th Day	803,126	69,760	79,259	19,742
7th Day	812,598	62,903	77,033	21,331
Max. Variation:	$\sim$ 2%	$\sim$125%	$\sim$4%	$\sim$4,000%

Table 8.1: Number of BadHoods/day

- The variation on the number of daily BadHoods is more significant for UT/EWI and DShield data sets (T-445 and T-3389) than for CBL (`Max Variation` row, which is the ratio between the day having most entries divided by the day having the least entries, or $100 \times Max/Min$.

- Abrupt variations can occur, as can be seen between 1st and 2nd days of T-3389 (November 2011).

We address these observations in details in the next two subsections.

8.2.1 Variation between the Datasets

A reason for the fact that variations were proportionally more significant for UT/EWI and DShield datasets than CBL has to do with the way each original blacklist is generated. UT/EWI and DShield datasets are generated based only on attacks observed on a *single day*, that is, all /32 entries they list correspond to, at least, one attack observed on the very day.

CBL, on the other hand, may list entries on a random day that were not observed in the very day. CBL and other public spam blacklist sources employ large spam traps infrastructures and after blacklisting a certain IP address, they may keep it on the list for many forthcoming days, even though no more spam has been observed from that particular IP. In fact, the CBL's de-list policy is "manual" – that is, the responsible network administrator for the blacklisted IP should go to CBL web site and manually remove the address from the list [166], otherwise it might remain blacklisted for many days, as can be seen for the IP address `221.0.141.106`, as shown in Listing 8.1, which was obtained on CBL's website on October 2nd, 2012. As can be seen, this IP address has been kept in CBL for more than 12 days since it was last observing attacking CBL traps. This, in turn, confirms the fact that entries are expected to remain blacklisted over multiple days even if there was no malicious activity (spam, in this case) observed from its originating sources. Since CBL's aim is to provide better protection against spam in relation to future attacks, keeping it on the blacklists makes sense in case recurrent spammers are frequent.

The combination of both factors (large infrastructure and manual de-listing policy) explains the smaller variation in CBL in comparison to UT/EWI (Table 8.1).

```
IP Address 221.0.141.106 is listed in the CBL. It appears to be infected with a
    spam sending trojan or proxy.

It was last detected at 2012-09-20 08:00 GMT (+/- 30 minutes), approximately 12
    days, 7 hours, 30 minutes ago.

This IP is infected (or NATting for a computer that is infected) with a spambot we
     have not yet been able to identify. For the time being we refer to it as the
     unknown66 spambot.

This IP is infected (or NATting for a computer that is infected) with a spam-
    sending infection. In other words, it's participating in a botnet. If you
    simply remove the listing without ensuring that the infection is removed (or
    the NAT secured), it will probably relist again.
```

Listing 8.1: CBL Lookup Result – October 2nd, 2012

8.2.2 Abrupt Variation in the Number of BadHoods

As shown in Table 8.1, the number of active BadHoods changed significantly over a single day for November 2011 Dshield T-3389 data sets (from 818 to 25,228, as shown in Table 8.1). Since we do not have the full traces, it is not easy to point the exact causes. However, we assume that it relates to the

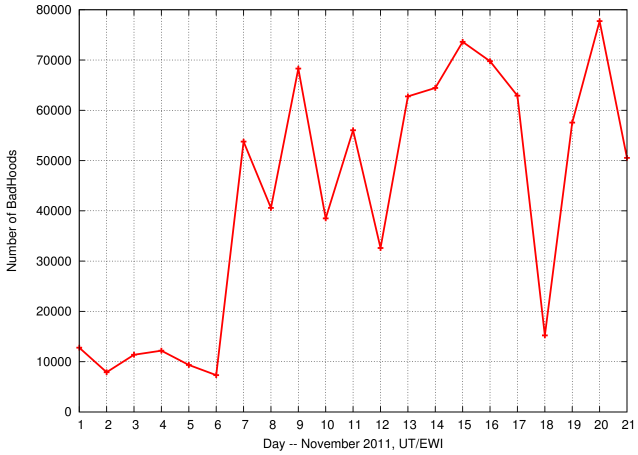


Figure 8.2: Number of BadHoods - UT/EWI

existence of new exploits or seasonal attack behaviors for the particular TCP port.

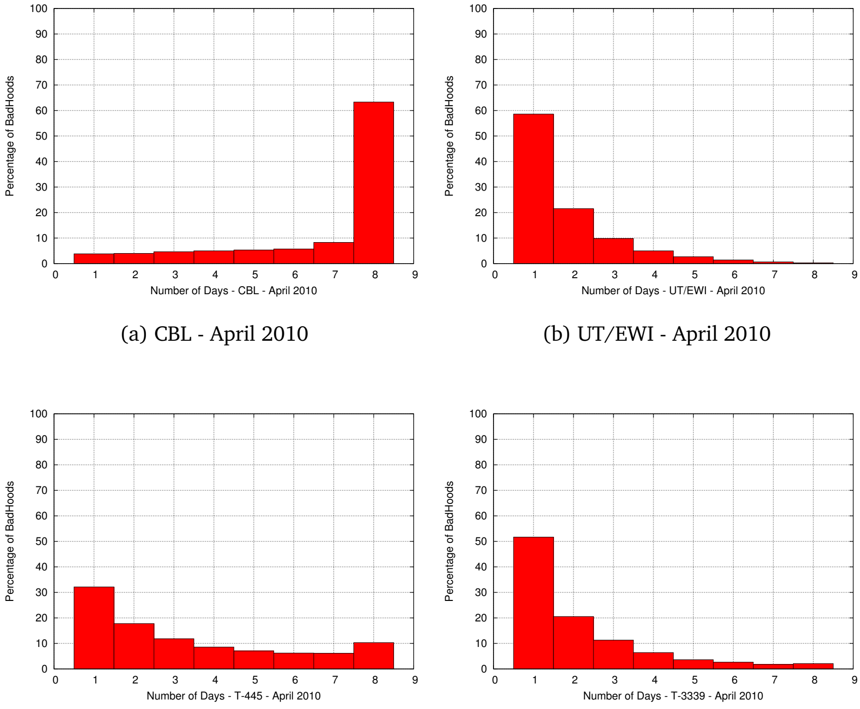
Such behavior is not exclusive for the Dshield data set. Figure 8.2 shows the number of BadHoods for the UT/EWI dataset, for an extended period of time (November 1st -24th, 2011). As can be seen, also for Spam BadHoods, we can observe significant variations from one day to the next (from less than 10K to more than 50K, between November 6th and November 7th).

Taking into account the results presented in this section, we can conclude that the number of active BadHoods a target observes varies on a daily basis. The degree of variation, however, varies according to the dataset. For CBL, however, the variation is smaller compared to the others also because of its delisting policy. This differentiation between data sets should, therefore, play a major role when designing prediction models.

8.3 Bad Neighborhoods Attack Strategy

In this section, we address RQ 8.2 and investigate the temporal attack strategy employed by BadHoods, as perceived by the targets.

We start by observing how many days BadHoods are active, for the two monitoring periods. Figure 8.3 and 8.4 show the distribution of the BadHoods



(a) CBL - April 2010

(b) UT/EWI - April 2010

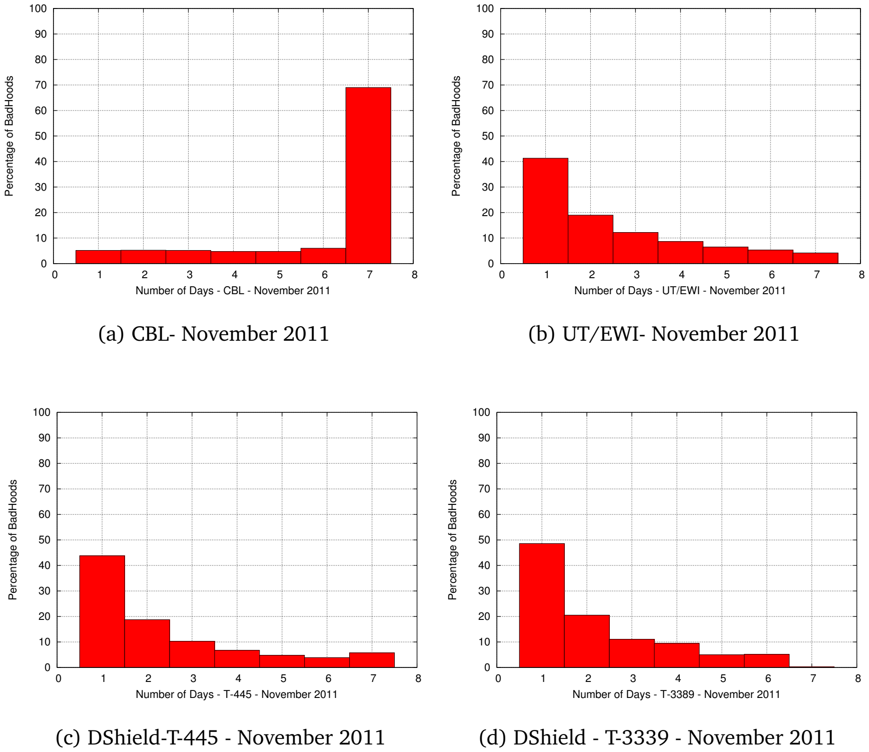
(c) DShield - T-445 - April 2010

(d) DShield - T-3339 - April 2010

Figure 8.3: Number of Days Active - April 2010

taking into account the number of days they are active (not necessarily consecutive days), that is, carrying out attacks. As can be observed in these figures, a significant part of the BadHoods are likely to attack again (for CBL, this is most prominent due to its de-listing policy). This provide good news, implying that using historical data is feasible to predict attacks for a new day.

On the other hand, some datasets have presented a significant percentage of BadHoods that attack a single day out of monitored days (almost 50% for T-3389 data sets). BadHoods that are active for only one day pose a challenge for BadHoods-based security systems, since such BadHoods attack on a single day and are not further observed within a short term period.



(a) CBL- November 2011

(b) UT/EWI- November 2011

(c) DShield-T-445 - November 2011

(d) DShield - T-3339 - November 2011

Figure 8.4: Number of Days Active - November 2011

8.3.1 The Bad Neighborhood Occurrence Score

The results presented in the previous section show the number of days a BadHood is active for the monitoring data sets. However, it does not show *which days* of the monitoring period are chosen by the BadHoods. For example, 2 days could be a combination of *any* 2 random days within the monitoring period.

In order to be able to tell what days (or combination of) the BadHoods are active, we propose in this section the *occurrence score*. For a given data set having n days of data, we define, for each /24 BadHood (B^{24}), an occurrence score as follows:

$$occurrence(B^{24}) = \sum_{i=1}^{n} 2^i \tag{8.1}$$

In this equation, i refers to the day that the particular BadHood (B^{24}) is active. i may vary from 1 (first day of the monitoring day, not necessarily the day of the month) until n, the last day in the observed data set. The final occurrence score is the sum of 2^i for each n day B^{24} is active. In the end, the final number is a single integer number that can be decomposed to reveal which days from the n days a certain BadHoods carried out attacks.

To better illustrate how the occurrence score is calculated and decomposed, consider the April 2010 data set from UT/EWI. Table 8.2 shows an excerpt of the final BadHood score file that was generated after scoring BadHoods for the monitoring period. For each BadHood, an occurrence score is provided, calculated using 8.1. As shown in this table, a score of 96 can be decomposed into two terms. The power of each of them (5 and 6) represents the days the BadHood was active: 5th and 6th of the monitoring period. These, in turn, represent April 23rd and 24th.

BadHood	Score	Decomposed Terms	Days Active
93.105.233/24	96	$2^5 + 2^6$	5th and 6th
94.66.155/24	16	2^4	4th
94.66.154/24	12	$2^2 + 2^3$	2nd and 3rd
93.105.231/24	228	$2^2 + 2^5 + 2^6 + 2^7$	2nd, 5th, 6th, and 7th
71.223.131/24	8	2^3	3rd
94.66.153/24	80	$2^4 + 2^6$	4th and 6th

Table 8.2: Occurrence Scores for UT/EWI BadHoods (April 2010)

An important property of the occurrence score is that any score $2^i < x < 2^{i+1}$ implies that the BadHood is active on the i-th day plus any previous day(s) ($i' < i$), but never on any days $> i$. For example, a score of 32 means that a BadHood is active on the 5th day. However, there is no other combination of days that would yield to a score > 32 and < 64 that would not include the 5th day. For example, if a BadHood is active on days 1–4, it's final score is 30, which is smaller than the occurrence of a single day alone (5th day = 32).

Occurrence Scores Distribution and CDF

Figures 8.5 and 8.6 show both the distribution and the cumulative distribution function (CDF) of the occurrence scores (left and right columns, respectively), for the April 2010 datasets, while Figures 8.7 and 8.8 show the results for the November 2011 data sets.

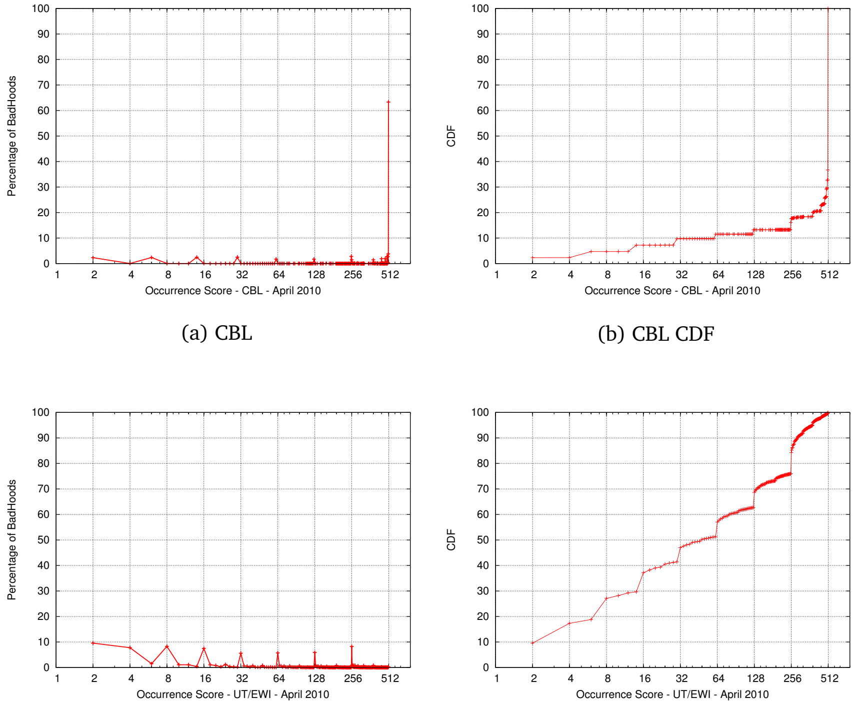
Analyzing the figures, we can observe that, with the exception of CBL, *no occurrence score is significantly more prominently than the others*. In fact, with the exception of CBL, all the other data sets observe small spikes on scores equal to 2^i, which are BadHoods that have only attacked on a single day. CBL, on the other hand, presents a significant spike on score 510 (a score that represents all previous days), as expected from Figure 8.3(a) and 8.4(a), which is due the de-listing policy and the size of their infrastructure.

What we can conclude from our analysis is that, except for CBL, *there is no day or a combination of days that is significantly more recurrent than others.* Therefore, our results show that a network administrator should not expect any pattern or regularity in terms of which days BadHood chose to attack – which makes the task of predicting attacks more complex.

8.4 Tracing Back BadHoods: Time Since Last Attack

From the previous results, we observe that there is no particular combination of days that emerges from the days BadHoods choose to carry out their attacks. Therefore, in this section, we focus on a single day of the monitoring period instead of all the monitored days. We single out the *last day* and scrutinize each observed BadHood, in order to determine *if* they can be traced back to any previous days. After that, we determine *how many days* have passed since the last attack.

To do that, we have carried out a three-step approach. First, we obtain all the /24 BadHoods of the last day of each data set (as covered in Section 8.2). Then, for each of them, look it up on the final occurrence score file generated for the whole monitoring period (as shown in Section 8.3.1). Those BadHoods that have been observed carrying out attacks in the last day in combination with any of the other previous days (in any combination) are filtered. Mathematically, this means that we have only considered BadHoods having an occurrence score larger than the threshold $\varepsilon > 2^i$, in which i is the number of monitoring days for



(a) CBL

(b) CBL CDF

(c) UT/EWI

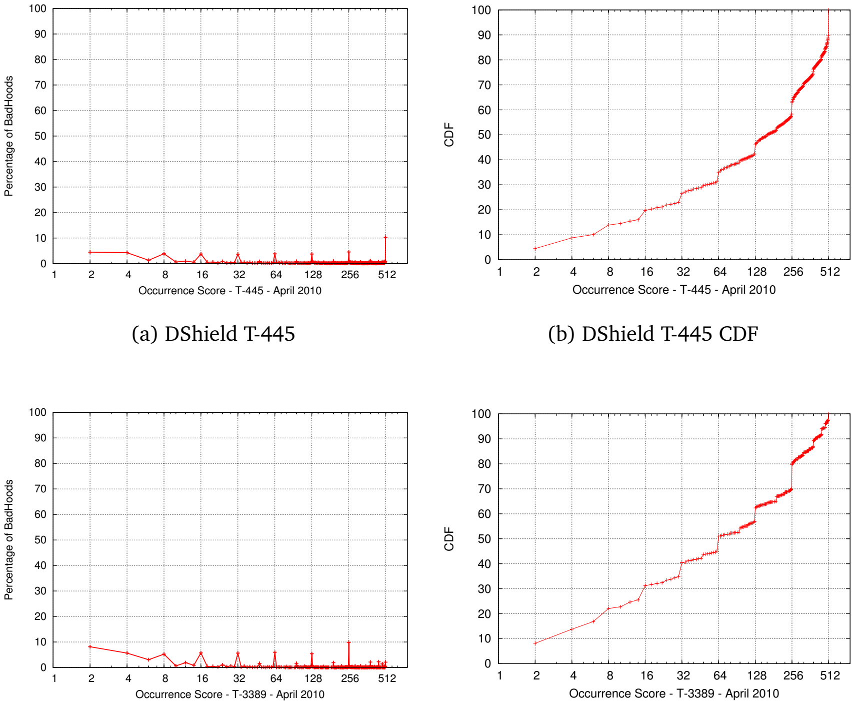
(d) UT/EWI CDF

Figure 8.5: Occurrence Scores – April 2010

each data set. For the April data sets, ε is equal to 256 and 128 for November datasets.

In our case, we are interested in the last i' day that a BadHood X^{24} has been active (the day before the singled out day). To illustrate this, consider that a certain BadHood from UT/EWI (April data set) has a score of 262. By decomposing this number into powers of two, it reveals that this BadHood has been active in days 8, 2, and 1 ($262 = 2^8 + 2^2 + 2^1$). From the days it was active, we compute the *difference between the last day (8) and the day right before it (2)*, which result in 6 days between attacks.

Table 8.3 shows the number of BadHoods on each data set, and the percent-



(a) DShield T-445

(b) DShield T-445 CDF

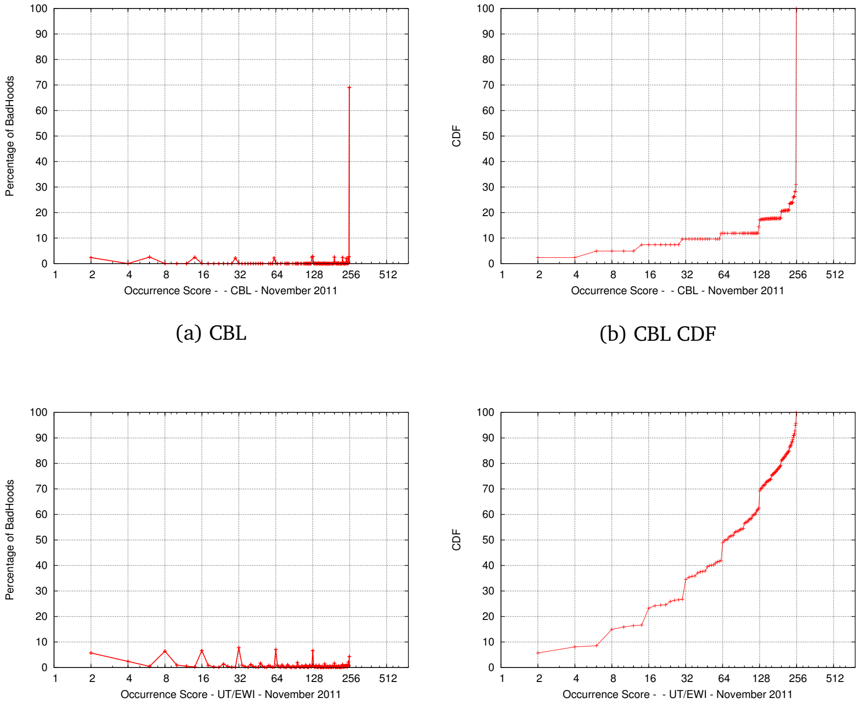
(c) DShield T-3389

(d) DShield T-3389 CDF

Figure 8.6: Occurrence Scores – April 2010

age of the recurrent ones ($OcurrenceScore > \varepsilon$). We can observe that for all the data sets (we have disregarded CBL, due to its removal policy), *the majority of BadHoods that attack a target have also been observed in at least one of the previous days.* For the April 2010 data sets, that means that 65-89% of all BadHoods observed in the last day are likely to have been observed on all previous days (7 days), while for the November 2011 datasets, 73-80% of BadHoods observed on the last day are likely to also have been active on all previous days (6 days).

Then, the next step was to determine when each of the recurrent BadHoods was last observed. Figure 8.9 shows these results as a cumulative distribution function (CDF). As can be seen, for all the data sets, the majority of the recurrent



(a) CBL

(b) CBL CDF

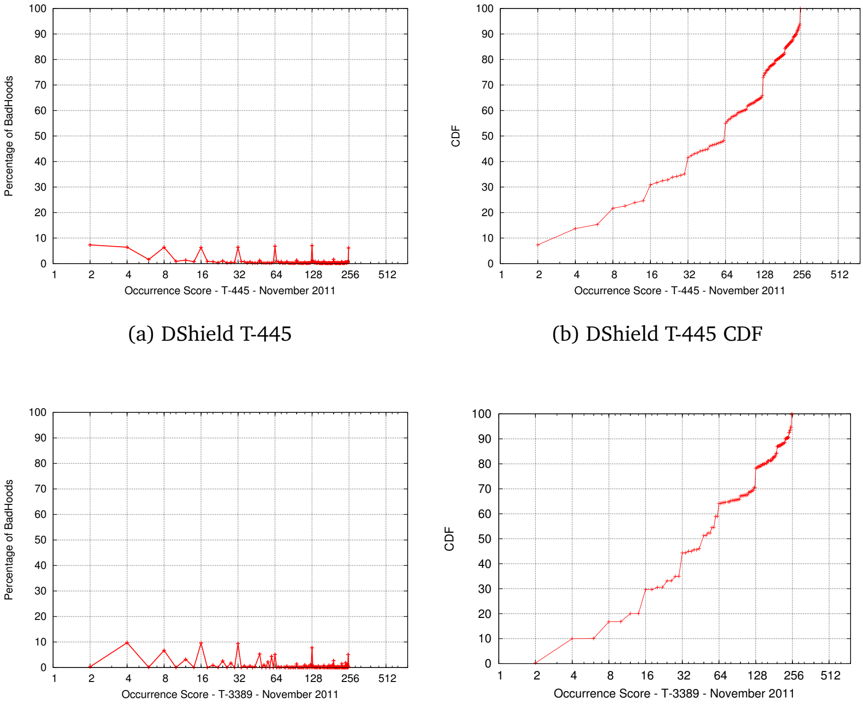
(c) UT/EWI

(d) UT/EWI CDF

Figure 8.7: Occurrence Scores – November 2011

BadHoods return within 5 days (>85%).

Another interesting fact that can be observed from these results is that, for all the data sets, at least 85% of the recurrent BadHoods are observed within the last five days, which is valuable information to determine how many days should be considered to build BadHood attack prediction models.



(a) DShield T-445

(b) DShield T-445 CDF

(c) DShield T-3389

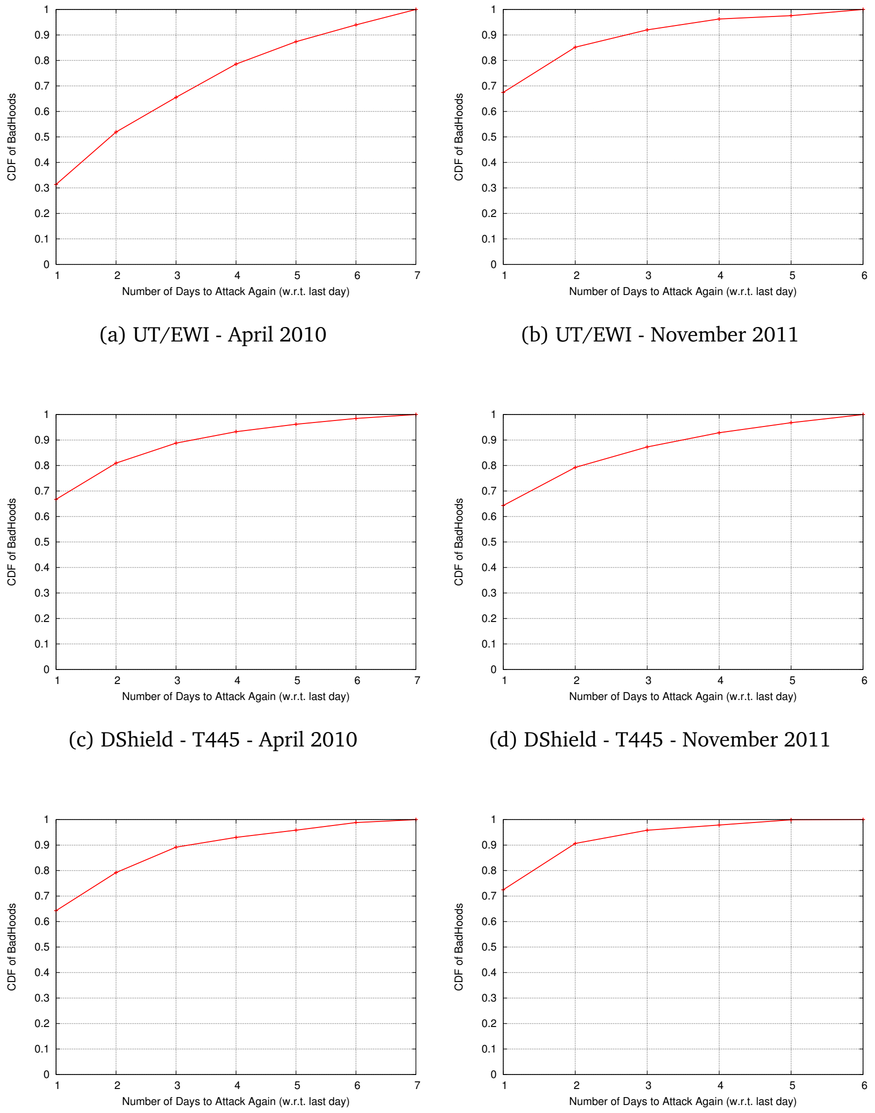
(d) DShield T-3389 CDF

Figure 8.8: Occurrence Scores – November 2011

8.5 Conclusions

In this chapter we have investigated the behavior of Bad Neighborhoods regarding their temporal attack strategy. In this sense, we have proposed three research questions and evaluated different real world data sets.

In RQ 8.1, we asked *"what is the daily variation in the number of observed BadHoods?"*. We found that, for all the evaluated datasets, a variation in the number of active BadHoods on a daily basis. In addition, we also found that the Spam Blacklist CBL shows a much smaller proportional variation than the other data sets, mainly because CBL keeps malicious hosts in their blacklists for



(a) UT/EWI - April 2010

(b) UT/EWI - November 2011

(c) DShield - T445 - April 2010

(d) DShield - T445 - November 2011

(e) DShield - T3389 - April 2010

(f) DShield - T3389 - November 2011

Figure 8.9: Number of Days to Attack Again - CDF

April 2010			
	UT/EWI	**DShield-T445**	**DShield-T3389**
BadHoods (Last Day)	59,739	141,512	895
Recurrent	39,237 (65.68%)	126,057 (89.07%)	602 (67.26%)
November 2011			
	UT/EWI	**DShield-T445**	**DShield-T3389**
BadHoods (Last Day)	62,903	77,033	21,331
Recurrent	51,745 (80.97%)	61,159 (79.39%)	15,732 (73.75%)

Table 8.3: Total and Recurrent BadHoods in Relation to the Last Day

multiple days even though they might not have been active for those days, in order to allow mail filters to protect from such hosts in future attacks. Also, we have shown that the daily variation can be very drastic, which confirms that targets are likely to be attacked by a varying number of BadHoods on a daily basis.

In RQ 8.2, (*"Given a certain monitoring period, in how many days a BadHood is observed carrying out attacks? And on what days do those attacks occur?"*), a significant part of BadHoods (between 40% and 95%, depending on the data set) are likely to attack a same target on multiple days (recurrent BadHoods). This confirms that it is useful to use historical past of BadHoods to predict new attacks. We have also found that there is no particular combination of days that BadHoods chose to attack a target.

Finally, in RQ 8.3 we asked "*Given a single monitoring day, how many BadHoods that carried out attacks in day can be traced back to previous days? And how long does it take for most of them to strike again?*". We found that the majority of the current BadHoods (85%) that attack a particular target are likely to attack it again within a 5 day period.

The findings presented in this chapter provide information that can be used to predict BadHood attacks. We have learned from RQ 8.1 that the daily number of BadHoods attacking a target depends on the data source and application; therefore a prediction model should leverage this and one could not expect an "one-size-fits-all" temporal prediction model. Moreover, the usefulness of the recent historical past has been proved in RQ 8.2, since up to 95% of BadHoods are likely to attack more than one day, which justifies its use to predict future attacks.

Part IV

Conclusions

Now this is not the end. It is not even the beginning of the end. But it is, perhaps, the end of the beginning.
Winston Churchill, 1942

CHAPTER 9
Conclusion

BAD neighborhoods can be found in the real world but also on the Internet. Ultimately, these areas are labeled as “bad” due to higher rates of malicious activities (e.g., robbery in the real world, spam on the Internet) they exhibit compared to the observed average. This dissertation has focused on the Internet counterpart.

In this concluding chapter, we summarize in Section 9.1 the contributions provided in this dissertation. Then, in Section 9.2, we present the main findings along with a discussion on their implications. Next, in Section 9.3, we present some possible steps forward based on the findings and contributions provided in this dissertation. Section 9.4 concludes this dissertation.

9.1 Summary of Contributions

Previous works have addressed the notion that attack sources tend to be concentrated in certain portions of the IP address space [27, 28, 29, 30, 31]. This dissertation, however, presents the first *systematic and multifaceted study on the concentration of malicious hosts, at various aggregation levels* – which is the main contribution of this dissertation.

By first framing such concentration of malicious hosts as *Internet Bad Neighborhoods*, we have put the Bad Neighborhoods (BadHoods) under scrutiny, in a multifaceted way. Figure shows 9.1 shows the BadHoods facets that were investigated in this dissertation, divided according to two main research questions (RQ), raised in Section 1.2.

For RQ 1 (“what are the characteristics of Internet Bad Neighborhoods?”), we have proposed, in Chapter 1, a definition for *what* a Bad Neighborhood is. Next, in Chapter 2, we have presented three assumptions for the occurrence of BadHoods (*“why”* in Figure 9.1). Following that, we have investigated in Chap-

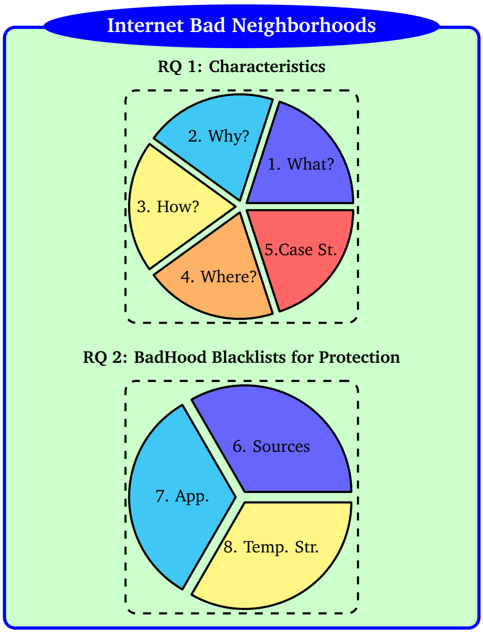


Figure 9.1: Multifaceted Study on Bad Neighborhoods – Research Questions and Chapters

ter 3 *how* malicious IP addresses can be aggregated into network prefixes (/24–/8, in CIDR notation [25]). Later, in Chapter 4, we have then evaluated our assumptions for the existence of BadHoods proposed in Chapter 2 and shown in which Internet Service Providers, countries, and cities BadHoods are *located*. Then, in Chapter 5, we have carried out a *case study* on spamming badHoods, due to the economic impact caused by e-mail spam, estimated to incur losses from $10 billion to $87 billion yearly [20].

In RQ 2 ("Which blacklists should a network administrator choose to protect a network against attacks from Internet Bad Neighborhoods"), in turn, we have assumed the point of view of a network administrator who employs BadHood blacklists to secure networks. We have then investigated, in Chapter 6, what blacklists *sources* a network administrator should use – public, peers, or local measurements. In Chapter 7, we have addressed the question whether a network administrator can employ BadHood blacklist obtained for one *application* (e.g., mail) to protect against attacks to other applications (e.g., ssh). Finally, in Chapter 8, we have investigated the *temporal* attack strategies employed by BadHoods to determine how often blacklists should be updated.

9.2 Main Findings and Implications

In this section we present the main findings of this dissertation, and also a discussion on their respective implications.

9.2.1 Bad Neighborhoods found at several aggregation levels: ISPs, Countries, Prefixes

One of the most important findings of this dissertation is the verification that Internet BadHoods is a real phenomenon, which can be observed not only as network prefixes (e.g., /24), but also at different and coarser aggregation levels, such as Internet Service Providers (ISPs) and countries.

As shown in Chapter 4, the top 20 Autonomous Systems (ASes), which are somehow comparable to ISPs, *concentrate almost 50% of all spamming IP addresses* observed in our data sets, from a total of 42,201 active ASes in our analysis. In the worst case, a single ISP from India (BNSL, AS number 9829) concentrated 7.39% of all the spamming addresses observed for the entire world in our datasets. Moreover, when considering the ratio of malicious IP addresses in an ISP (number of spamming addresses divided by the number of announced addresses), we found that some ISPs have an alarming ratio of up to 62.55% of their announced IPs sending spam (SpectraNet, AS Number 37340, an ISP from Nigeria). These results confirm the *existence of Bad Neighborhoods at the ISP level*.

Also, we found that BadHoods are concentrated in certain countries. For the case of spam, even though we found spamming hosts all over the world, but the BadHoods are more likely in certain countries. Out of 229 countries found having spamming hosts, a single one (India) was found concentrating almost 20% of worldwide spamming IP addresses, followed by Vietnam and Brazil (∼7% each). In total, *the top 20 countries were responsible for 76,31% of all the spamming IP addresses*. These results also confirm that certain countries concentrate most of malicious spamming IP addresses.

These findings advance the state of the art by showing that malicious hosts are concentrated not only in certain portions of the IP address space [27, 28, 29, 30, 31], but more clearly at higher aggregation levels, such as ISPs and countries.

The implications of these findings are twofold: first and foremost, our findings supports that AS-based and country-based BadHoods blacklists can be employed as an *auxiliary* approach to evaluate traffic from unknown sources. Our results do no support, however, that a country or AS should be entirely blacklisted; instead, BadHood blacklists at such coarse levels should be employed to aid existing solutions when scoring unknown traffic. For example, in a mail filter, such as `SpamAssassin` [57], AS-based and Country-based BadHood blacklists can be used to complement current filter rules when scoring the likelihood a message being spam, or even consider network prefixes smaller than /24, using the algorithms proposed in Chapter 3. The main advantage of BadHood-based solutions is that we can avoid looking into the contents of an e-mail message, which is typically employed in Bayesian spam filtering techniques [167, 168, 169]. Such techniques are more CPU intensive than simple IP/ASN lookups.

Another implication of these findings is that it makes it "easier" to tackle the problem of malicious IP addresses on the Internet, by "nipping the problem in the bud". Instead of traditionally blacklisting individual /32 hosts, our results support that a "clean up" on networks in ISPs and countries having higher concentration of malicious IP addresses would be more effective. Such measures can be also supported through specific legislation – similar to the United States' CAN SPAM act [48] and European Union's Directive on Privacy and Electronic Communications (2002/58) [49], even though we have shown in Section 5.5 that legislation alone may be not sufficient (five out of the top twenty high volume spamming BadHoods are located in the European Union).

9.2.2 Bad Neighborhoods May Vary According to the Application Exploited

In the real world, one could intuitively expect that a dangerous area in a city is more likely to have higher concentration/incidence of *various* crimes, such as robbery, car theft, etc. On the Internet, however, that is not the case: BadHoods are *mostly application-specific* and may be located in neighborhoods one would not expect.

In Chapter 4, we have shown that while spam is distributed all over the world (but concentrated in Southern Asia), *phishing* Bad Neighborhoods, on the other hand, are mostly concentrated in the United States and other developed nations. This counterintuitive finding is due to the *context* – in this case the specifics of spam and phishing. Most of spamming hosts are part of an army of “hijacked” malicious hosts (part of botnets), typically at home, schools and businesses with no availability guaranteed. *Phishing* hosts, on the other hand, are “required” by criminals to have a much higher availability than spam bots: the phishing site should be accessible most of the time, so more people can be deceived. If the website is down, the criminal misses the “business opportunity”. The outage of a single spam bot, on the other hand, has a minimal impact on the overall spam capacity of the botnet. Therefore, phishing websites are more likely to be hosted on reliable infrastructures, typically data centers/cloud providers, which, in turn, are mostly located in developed nations, mainly in the United States.

In addition, in Chapter 7, we found that the number of attacks varies significantly according to the application in question, and for most of the applications, the set of attacking BadHoods were almost disjoints. The conclusion is that Bad Neighborhoods are application-specific.

The implications of these findings is that security systems employing BadHood-based techniques should employ application-specific BadHood blacklists. In addition, research work aiming at predicting attack sources, such as the work by Soldo *et al.* [76], can be optimized by taking into account the application employed in the attack (instead of disregarding it).

9.2.3 Bad Neighborhoods are Likely to Attack Again

In Chapter 8, we found that the number of /24 BadHoods (in CIDR notation [25]) that attack individual targets varies daily. After being attacked by a particular BadHood, however, a network administrator might wonder if the same BadHood will return to attack the same target again, and if it will, when can it be expected. We found in Chapter 8 that 40–95% of all BadHoods are likely to strike a target more than once, depending on the application/dataset, within a week period. However, there were no particular combinations of days that /24 BadHoods choose to carry out attacks .

This finding highlights the benefits provided by employing the Bad Neighborhood concept. For example, in Chapter 5, we found that, in a one week period, 46.94% of the individual IP addresses attack *only once*, part of their stealth tactic (“flying under the radar”, as discussed in Appendix B).

We also found in Chapter 8 that, by singling out all /24 BadHoods that attacked a target in a particular day, 65–89% of these /24 BadHoods have also carried out attacks in at least one day of the seven previous days, depending on the target/application in question.

The implications of these results are that network administrators have to constantly update their BadHood blacklists in order to keep up with their dynamics. In addition, our research confirms that historical data of BadHoods attacks should be employed to predict future attacks.

9.2.4 Public Blacklist Sources Allow Better Detection Results

When employing BadHood-based techniques to secure a network, a network administrator can employ blacklists from third-parties or carry out local measurements to generated local blacklists. In Chapter 6, our results have shown that it is better to employ blacklists obtained from public third-party sources, instead of carrying out local measurements.

This is due to fact that specialized public sources typically employ a large number of distributed monitoring points. Consequently, these sources are able to capture more malicious hosts – which typically try to employ a stealth “under the radar” attack strategy – as discussed in Appendix B. Therefore, a network administrator can protect a network by “anticipating” BadHoods blacklisted by public sources. However, it is necessary to carry out an assessment of individual public blacklists in order to verify their quality before applying it.

9.2.5 “Silent Ticking Spam Bomb” in BRIC countries

Another finding in this dissertation is that there might be a “silent ticking spam bomb” in the BRIC countries (Brazil, Russia,India, and China). Currently, these countries have a moderate Internet penetration (Brazil - 40.6%. Russia - 43.0%, India - 7.5%, China - 34.3%, World Average - 35% [110]) that is expected to grow between 9% to 15% yearly, according to a Boston Consulting Group report [111], driven because of their economic growth. The growth, *per se*, is a positive achievement for the countries and their population, since “exclusion from it (the Internet) is one of the most damaging forms of exclusion in our economy and culture” [13], as stated by the sociologist Manuel Castells.

However, a problem might emerge if the ratio of malicious IP addresses in these countries remains stable while the number of Internet users increases. In that case, we can expect a significant increase on the overall number of spam sources. To illustrate this, consider India, a country that ranks first in number of spamming IP addresses. If India would have the same Internet penetration rate as the United States (a developed country comparable in size) while keeping its current ratio of malicious IP addresses, that would cause an increase of 200%

in the total number of malicious spamming addresses observed currently for the whole world.

9.3 Moving Forward from Findings

As described in Section 1.2, the goal of this dissertation was to scrutinize the Bad Neighborhood phenomenon on the Internet to better understand its intrinsic characteristics. The motivation for doing so was to set the theoretical foundations in which BadHood-based security solutions can be build upon.

Therefore, the next natural step is to employ the knowledge provided in this dissertation in security solutions. Traditional /32 IP addresses blacklists try to protect a target from attack sources based on historical pasts. BadHood-based solutions can be seen as a step further in this process, by both *protecting from previous sources* but also by *predicting new sources* (neighbors) of attacks, as we have shown in Section 2.7.

In Appendix E[1], we evaluate a BadHood-based mail filter based on the findings obtained in Chapter 6. We implement an algorithm that uses as parameter the number malicious addresses per BadHood to tell if a message is spam.

However, the major direction we envision for BadHood-based research is to develop algorithms that combine not only findings from a single chapter, but algorithms that build upon the findings presented in the entire dissertation. To mention a few, we have seen in Chapter 6 that employing BadHood blacklists of third-party sources leads to better detection results, while in Chapter 7 we have seen that BadHood blacklists should be application-specific. In addition, we have provided two aggregation algorithms in Chapter 3 and seen that the coarser the aggregation criteria, the larger aggregation errors. Moreover, in Chapter 8, we have seen 40–95% of BadHoods are likely to strike more than once, depending on the application/dataset, within a week period. Also, in Chapter 4, AS-based and country-based BadHood blacklists should be employed in the process.

The next step is therefore to combine all the findings and evaluate the results, in a similar way to what is presented in Appendix E.

In addition, as discussed in Section 1.4, this dissertation has covered IPv4-based BadHoods. With the increasing adoption of IPv6, we can expect more attacks from IPv6 Bad Neighborhoods (currently IPv6 traffic accounts for less

[1] We have not included in Chapter 6 because it addresses contents beyond the scope of the chapter.

than 2% of networks such as Internet2 [41]). As we have covered in Appendix C, an aggregation-style approach like the Internet Bad Neighborhoods approach *is a necessity* (for scalability purposes) when dealing with IPv6 attacks, due to the way that IPv6 addresses are allocated. However, further investigation will be necessary to confirm if the findings provided in this dissertation hold for IPv6 Bad Neighborhoods.

9.4 Concluding Remarks

This dissertation has provided the first multifaceted investigation on the Bad Neighborhood phenomenon in the Internet. We have shown that Internet Bad Neighborhoods can not only be found at certain /24 networks, but that they can also be observed in coarser aggregation levels, such as ISPs and countries.

We have then assumed the point of view of a network administrator whose goal is to protect a network. By putting Bad Neighborhoods under scrutiny, we have investigated how specific they are in relation to the application used in the attacks and the targets they have chosen. Our expectation is that the findings provided in here can serve as a guide for the design of new algorithms and solutions to better secure networks.

APPENDIX A

List of Publications

During the four years of his PhD project, the author of this dissertation has co-authored the following publications:

1. Moura, G. C. M., Sperotto, A., Sadre, R., Pras, A.: Evaluating Third-Party Bad Neighborhood Blacklists for Spam Detection. In: IFIP/IEEE International Symposium on Integrated Network Management (IM 2013), Ghent, Belgium, 27-31 May 2013 IFIP/IEEE International Symposium on Integrated Network Management (IM 2013), Ghent, Belgium, 27-31 May 2013 (*to appear*).

2. Moura, G. C. M., Sadre, R., Sperotto, A., Pras, A.: Internet Bad Neighborhoods Aggregation. In: IEEE/IFIP Network Operations and Management Symposium (NOMS 2012), Maui, Hawaii, USA, 16-20 April 2012.

3. Moura, G. C. M., Sadre, R., Pras, A.: Internet Bad Neighborhoods: the Spam Case. In: 7th International Conference on Network and Services Management (CNSM 2011), Paris, France, 24-28 October 2011.

4. Francois, J., Moura, G. C. M, Pras, A: Cleaning Your House First Shifting the Paradigm on How to Secure Networks . In: 5th International Conference on Autonomous Infrastructure, Management and Security (AIMS 2011), Nancy, France, June 13-17th, 2011 .

5. Hofstede, R., Drago, I., Moura, G. C. M, Pras, A.; Carrier Ethernet OAM: A Survey and Comparison to IP OAM . In: 5th International Conference on Autonomous Infrastructure, Management and Security (AIMS 2011), Nancy, France, June 13-17th, 2011.

6. Van Polen, M., Moura, G. C. M, Pras, A.; Finding and Analyzing Evil Cities on the Internet. In: 5th International Conference on Autonomous Infras-

tructure, Management and Security (AIMS 2011), Nancy, France, June 13-17th, 2011.

7. Pras, A. and Sperotto, A. and M. Moura, G.C. and Drago, I. and Barbosa, R.R.R. and Sadre, R. and de Oliveira Schmidt, R. and Hofstede, R. (2010) Attacks by “Anonymous” WikiLeaks Proponents not Anonymous. Technical Report TR -CTIT-10-41, Centre for Telematics and Information Technology, University of Twente, Enschede. ISSN 1381-3625.

8. de Vries, W., Moura, G. C. M., and Pras, A.: Fighting Spam on the Sender Side: a Lightweight Approach. In: Proceedings of the 16th International EUNICE Workshop - Networked Services and Applications – Engineering, Control and Management (EUNICE 2010), Trondheim, Norway 28-30th June, 2010 .

9. Van den Broek, G. and ten Hoeve, S. and M. Moura, G.C. and Pras, A. (2009) SNMP Trace Analysis: Results of Extra Traces. Technical Report TR-CTIT-10-06, Centre for Telematics and Information Technology, University of Twente, Enschede. ISSN 1381-3625.

10. Moura, G. C. M., Fioreze, T., de Boer, P.-T., and Pras, A.; Optical Switching Impact on TCP Throughput Limited by TCP Buffers. In: Proceedings of the 9th IEEE International Workshop on IP Operations and Management (IPOM 2009), Short Paper Venice, Italy. 26th-30th October 2009.

11. Moura, G. C. M. and Pras, A.; Scalable Detection and Isolation of Phishing. In: Proceedings of the 3rd International Conference on Autonomous Infrastructure, Management and Security (AIMS 2009), PhD Workshop. 30 June - 2nd July, Enschede, The Netherlands.

Media Coverage

In addition, two of the publications have been covered/mentioned by selected international media outlets:

- **Paper**: Van Polen, M., Moura, G. C. M, Pras, A.; Finding and Analyzing Evil Cities on the Internet. In: 5th International Conference on Autonomous Infrastructure, Management and Security (AIMS 2011), Nancy, France, June 13-17th, 2011.
  - Slashdot: `http://it.slashdot.org/story/11/07/10/2219247/zeroing-in-on-the-internets-evil-cities`

  - CNET Buzz Out Loud Podcast 1507: http://www.cnet.com/8301-19709_1-20078494-10/buzz-out-loud-1507-were-out-of-disk-space-too-podcast/
  - Physorg: http://phys.org/news/2011-07-russian-city-cyber-capita.html
  - Tech In Asia: http://www.techinasia.com/most-evil-cities/

- **Technical Report:** Pras, A. and Sperotto, A. and M. Moura, G.C. and Drago, I. and Barbosa, R.R.R. and Sadre, R. and de Oliveira Schmidt, R. and Hofstede, R. (2010) Attacks by "Anonymous" WikiLeaks Proponents not Anonymous. Technical Report TR -CTIT-10-41, Centre for Telematics and Information Technology, University of Twente, Enschede. ISSN 1381-3625.

  - BBC: http://www.bbc.co.uk/news/technology-11983246.
  - New York Times: http://gadgetwise.blogs.nytimes.com/2010/12/13/how-anonymous-shut-down-sites/?src=busln.
  - Le Monde: http://www.lemondeinformatique.fr/actualites/lire-wikileaks-les-attaquants-utilisants-loic-pourraient-etre-facilement-traces-32417.html.
  - Sydney Morning Herald: http://www.smh.com.au/technology/security/wikileaks-cyber-war-proassange-anonymous-v-us-nationalists-20101213-18uuo.html.
  - Stern: http://www.stern.de/digital/online/hacken-fuer-wikileaks-die/-rache-der-vernetzten-1634038.html.
  - Slashdot: http://yro.slashdot.org/story/10/12/11/0228212/Anonymous-WikiLeaks-Proponents-Not-So-Anonymous.
  - PC World:http://www.pcworld.com/businesscenter/article/213395/website_attackers_could_be_easily_traced_researchers_say.html.
  - CNET: http://news.cnet.com/8301-17852_3-20025419-71.html?tag=mncol;2n
  - The Register:http://www.theregister.co.uk/2010/12/13/dutch_hackers_collective_wants_to_re_educate_infowar_hacktivists/.
  - Globo: http://g1.globo.com/tecnologia/noticia/2010/12/ferramenta-de-ataques-pro-wikileaks-revela-identidade-de-usuario.html.

APPENDIX B

The Rise of Botnets

In recent years we have seen a shift in the way malicious activities such as Spam are performed on the Internet. While in the past most of the attacks originated from *single* compromised servers [170, 171], a significant part of current attacks comes from *distributed* compromised machines, part of the so-called *botnets* [21, 22].

In the beginning of the last decade, most spam was sent from dedicated server farms, open relays, or compromised servers [171]. In fact, back then spamming was not even a crime: only in 2003 the CAN-SPAM Act made spamming illegal in the United States [48]. To fight spam, several techniques were developed, and the advent of real-time blacklists containing the IP addresses of those spam sources became effective [19].

Meanwhile, also in the first years of the last decade, broadband technologies such as ADSL increased home broadband adoption all over the world [172]. As a consequence, home computers were left online for more time, while having increased bandwidth in comparsion to the old dial-up access.

In this context, spam gangs realized that they could improve their strategies to carry out spam campaigns. Instead of relying on their spam farms and servers that increasingly became less effective due to the advent of real-time IP blacklists, spammers could "relay their messages through untainted third-party hosts" [19]. Third-party hosts, in this case, were computers with broadband connections at homes, schools, businesses and governments, running vulnerable operation systems [22]. Even though the processing and networking capabilities of each host was not enough to conduct major spam campaigns or Distributed Denial-of-Service (DDoS) attacks, the combined capability of a large set of hosts was, which drove to the creation of modern large-scale botnets. Current botnets, such as BredoLab, were estimated to have a spam capacity of 3.6 billion messages per day, by compromising more than 30 million hosts worldwide [173].

It is important, in the context of this dissertation, to clarify the *relation between botnets and bad neighborhoods*. Even though hosts belonging to a botnet are likely to cause the network to be labeled as a BadHood, this does not imply that all BadHoods are product of a botnet activity. As shown in Chapter 5, for example, we found some Yahoo! Mail servers, located in the UK, sending spam messages. However, this is mostly likely due to account hijack, in which spammers hijack legitimate accounts from users and employ it to send spam [145, 146], benefiting from Yahoo's reputation. However, since most of Spam and DDoS are nowadays carried out by botnets, it is likely to expect that a significant part of BadHoods are labeled as such due to bot activity.

"Fly Under The Radar" Attack Strategy

Botnets can actually be seen as "virtual armies" of compromised hosts [21] distributed all over the world. Figure 1.1 showed the geo-location of a sample of 1,193 zombies belonging to the botnet Hlux2/Kelihos.B [23] (this botnet was later found having more than 100,000 bots). By doing such attacks using zombies, attackers can hide their real identity and amplify the power of the attacks [22], as described in Section 2.3.

As explained by Bailey *et al.*, one of the main problems for botnet herders is to spread their worms into other computers, increasing the size of the botnet army [22]. Many propagation techniques can be employed for this purpose, and current botnets combine many of them to maximize infection. For example, the "SDBot exploits Windows vulnerabilities, P2P networks, and backdoors left by previous worms" [22].

To cope with the fact that bots are distributed all over the world, real-time IP blacklists were developed. These lists contain IP addresses that originated malicious activity and are constantly updated in order to keep up with the dynamism of the sources of attacks. Such blacklists are popular and are used to fight spam, and mail servers are configured to query such blacklists, in real-time, any time a new message arrives [44].

In this cat-and-mouse game, Internet criminals also have developed methods to try to circumvent blacklist-based solutions. Since blacklist-based detection is *reactive* – that is, as soon as an attack is detected, the source is blacklisted and future messages are blocked – spammers found that they could still carry out their spam campaigns by "flying under the radar" – that is, by spamming a mail server using a large number of bots, but sending only one spam per bot.

As shown in Chapter 5 (Table 5.3), we found from mail server logs that 46.94% of the spammers have sent *only 1 spam message* over a period of one week. This, in fact, configures a major problem for source IP blacklist-based detection systems, since the server keeps getting spam while blacklisting sources that will never return.

APPENDIX C

IPv6 Bad Neighborhoods

Standardized by the Internet Engineering Task Force (IETF), The Internet Protocol version 6 (IPv6) [40] is a revision of the IP protocol aimed to succeed the standard version 4 (IPv4). One of the main design requirements for IPv6 was to cope with the well known problem of lack of IPv4 addresses for the current number of devices connected to the Internet. In fact, the last two /8 IPv4 netblocks were allocated by Internet Assigned Numbers Authority (IANA) on February 3rd, 2011 [174] .

Currently, IPv4 still dominates the volume of traffic on the Internet, and IPv6 represents no more than 1% [41] in backbones such as Internet2 or at the Amsterdam Internet Exchange point (AMS-IX)(the weekly average incoming traffic in AMS-IX is 926.845 Gbps, while IPv6 accounts for 2.5 Gbps, or 0.2% of the total [42]). However, we can expect an increase in the volume of IPv6 traffic as more IPv6 addresses are assigned. For example, at the University of Twente, in which we have a fully operational IPv6 network, IPv6 represents 3.2% of the traffic volume.

With the increase adoption of IPv6, we can expect more attacks from IPv6 sources, as the first reported IPv6 DDoS attacks in 2012 [43]. Therefore, in the context of this dissertation, it is important to address what we can expect from IPv6 BadHoods.

As shown in Appendix B, to cope with blacklist-based defense software, attackers started to use more and more intermediary hosts to carry out attacks, so they can relay their attacks through “untainted third-party hosts” [19]. In the IPv4 standard, the IP source address field has a length of 32 bits – which means that theoretically, attackers could use 2^{32} different IP addresses for their attack, or approximately 4.2×10^9 addresses. On IPv6, the source address field was extended to 128 bits, which means that, theoretically, 2^{128} IPv6 are available, or approximately 3.4×10^{38} addresses (or 6.67×10^{27} IPv6 addresses per square meter on Earth). This massive number of IPv6 addresses present a major

n bits	64 -n bits	64 bits
Global Routing Prefix	subnetID	interfaceID

Figure C.1: Aggregation into Bad Neighborhoods

challenge for blacklist-based IPv6 security mechanisms.

C.1 IPv6 Addressing Architecture

The IPv6 addressing architecture is defined in the RFC 4291 [175]. As covered in the RFC, there are three types of IPv6 addresses:

- Unicast: An identifier for a single network interface (equivalent to IPv4 unicast)
- Anycast: An identifier to a set of network interfaces. A packet sent to an anycast addres is delivered to the "nearest" interface of the set, as defined by the "routing protocols' measure distance" [175].
- Multicast: Similar to anycast, but a packet sent to a multicast address is delivered to all interfaces in the set.

In this section, we focus on unicast addresses (a similar analysis can be conducted for anycast and multicast addresses). Figure C.1 shows the format of a IPv6 global unicast address (that is, "routable" on the Internet). In this figure, the 128 bits of the field are divided into three parts. The first two parts (global routing prefix and subnet ID) are used for routing, while interface ID is used to identify the host interface.

Taking this into account, it is likely that providers will assign end sites (e.g., home users) with an /48 IPv6 address (64 bits for global routing prefix, 16 for subnet, and 48 for interface), as described in RFC 6177 [176].

That implies that home users will have at their disposition 2^{48} ($\sim 2.41 \times 10^{14}$) IPv6 addresses to choose from, which is larger than the total IPv4 capacity (2^{32}). As a consequence, a single malicious host can actually carry out attacks (e.g., send a spam message) using a valid IPv6 address and never have to use it twice, since he/she has at least 2^{48} addresses to choose from.

To illustrate this, consider that an IPv6-enable host is part of a botnet and has received an order to spam a single particular mail server. If this spammer is able to send 1 million messages per second to the same mail sever (which is already an absurd number), it will take almost nine years to exhaust all addresses available within the /48 netblock. And that is only one spammer. In this context, standard IPv6 /128 blacklists cannot cope with the massive number of addresses available for attackers to use.

In this sense, the Bad Neighborhood approach to malicious networks is fundamental to cope with the vast number of valid IPv6 addresses available. We propose the following for IPv6 blacklists:

1. Employ /48 as the smallest BadHood aggregation netblock for IP-addressing based BadHoods;

2. Analyze real world traces from IPv6 attacks, and investigate if larger prefixes are necessary to cope with it (e.g., /64)

We believe that some collateral damage may occur. For example, if a single /64 gets blacklisted, maybe a legitimate host within the /64 also has to pay the price. Other approaches might be necessary to avoid this, but blocking individual /128 hosts will not be efficient.

APPENDIX D

Country Codes Employed in Chapter 4

Code	Country
AR	ARGENTINA
AM	ARMENIA
AU	AUSTRALIA
BY	BELARUS
BM	BERMUDA
BR	BRAZIL
CL	CHILE
CN	CHINA
CO	COLOMBIA
CZ	CZECH REPUBLIC
FR	FRANCE
DE	GERMANY
HK	HONG KONG
IN	INDIA
IR	IRAN
IE	IRELAND
IT	ITALY
KZ	KAZAKHSTAN
KR	SOUTH KOREA
LT	LITHUANIA
LU	LUXEMBOURG
MY	MALAYSIA
MU	MAURITIUS
MA	MOROCCO
NL	NETHERLANDS

NG	NIGERIA
PK	PAKISTAN
PS	PALESTINIAN TERRITORY
PE	PERU
PH	PHILIPPINES
PR	PUERTO RICO
RO	ROMANIA
RU	RUSSIAN FEDERATION
SA	SAUDI ARABIA
RS	SERBIA
SI	SLOVENIA
ES	SPAIN
TH	THAILAND
TR	TURKEY
UA	UKRAINE
US	UNITED STATES
UY	URUGUAY
UZ	UZBEKISTAN
VN	VIETNAM
VG	VIRGIN ISLANDS

Table D.1: Country Codes

APPENDIX E

Third-Party Bad Neighborhood Blacklists for Spam Detection

This appendix contains an excerpt of the following paper, accepted for publication at the moment of writing. The publication is based on the results and datasets presented in Chapter 6:

- Moura, G. C. M., Sperotto, A., Sadre, R., Pras, A.: Evaluating Third-Party Bad Neighborhood Blacklists for Spam Detection. In: IFIP/IEEE International Symposium on Integrated Network Management (IM 2013), Ghent, Belgium, 27-31 May 2013 (*to appear*)

E.1 Effectiveness on Detecting Spam

In Chapter 6, we have focused on how specific a BadHood blacklist is to its measurement points. The presented analysis allowed to verify if BadHood Blacklists collected from different sources share the same observation span, i.e., if they observe the same BadHoods.

In this section, we analyze the effectiveness of the third-party BadHood blacklists by measuring the ratio of detected Spam messages. In addition, we investigate whether we can interchangeably use different blacklists to protect a target from Spam.

We begin with a description of the used methodology and the considered scenario in Section E.1.1, followed by a discussion of the achieved results in Section E.1.2.

E.1.1 Methodology and Considered Scenario

In [31], the authors present an approach for Spam filtering which relies on analyzing the origin of e-mail messages as well as the links within the messages to malicious websites. One of the criteria used in their approach is whether the number of malicious hosts in the origin BadHood of the e-mail is above a certain threshold. Similar to our work, the authors used publicly available blacklists to build the list of BadHoods.

With this Spam detection scenario in mind, we investigate here the effectiveness of employing different BadHood blacklists to detect Spam messages. For our experiments, we propose a simple Spam detection system that implements the threshold-based criterion described above.

Consider L_S as the BadHood blacklist to be used for Spam detection. Whenever a new message M arrives, the mail filter extracts the source /24 netblock address of the sender ($M_{/24}$) and checks it against the list L_S. If $M_{/24}$ is found in L_S, then the mail filter will declare the message as Spam if $nHosts(M_{/24}) > \theta$, where θ ($0 \leq \theta < 256$) can be considered a threshold on how malicious a BadHood is. This procedure is summarized in Algorithm 3.

To evaluate the effectiveness of the different BadHood blacklists, we follow the same scenario as described in Figure 6.1: We regard the mail servers of Provider A, UT/EWI, and CAIS/RNP as targets to be protected from Spam. We apply Algorithm 3 to each target T for different values of θ and for the different

Algorithm 3 Spam detection procedure used in the experiments

```
Require: $L_S = \{\langle B_i, nHosts_S(B_i)\rangle, i = 1 \dots n_S\}$
Require: $\theta$
Require: $M_{/24}$
Ensure: true, if spam detected; false, otherwise
if $M_{/24} \in L_S$ and $nHosts_S(M_{/24}) > \theta$ then
    return true
else
    return false
end if
```

blacklists L_S and calculate the achieved Spam detection hitcount $h_{S,T}(\theta)$ by

$$h_{S,T}(\theta) = \frac{\text{number of Spam mails detected}}{\text{total number of Spam mails received by } T} \tag{E.1}$$

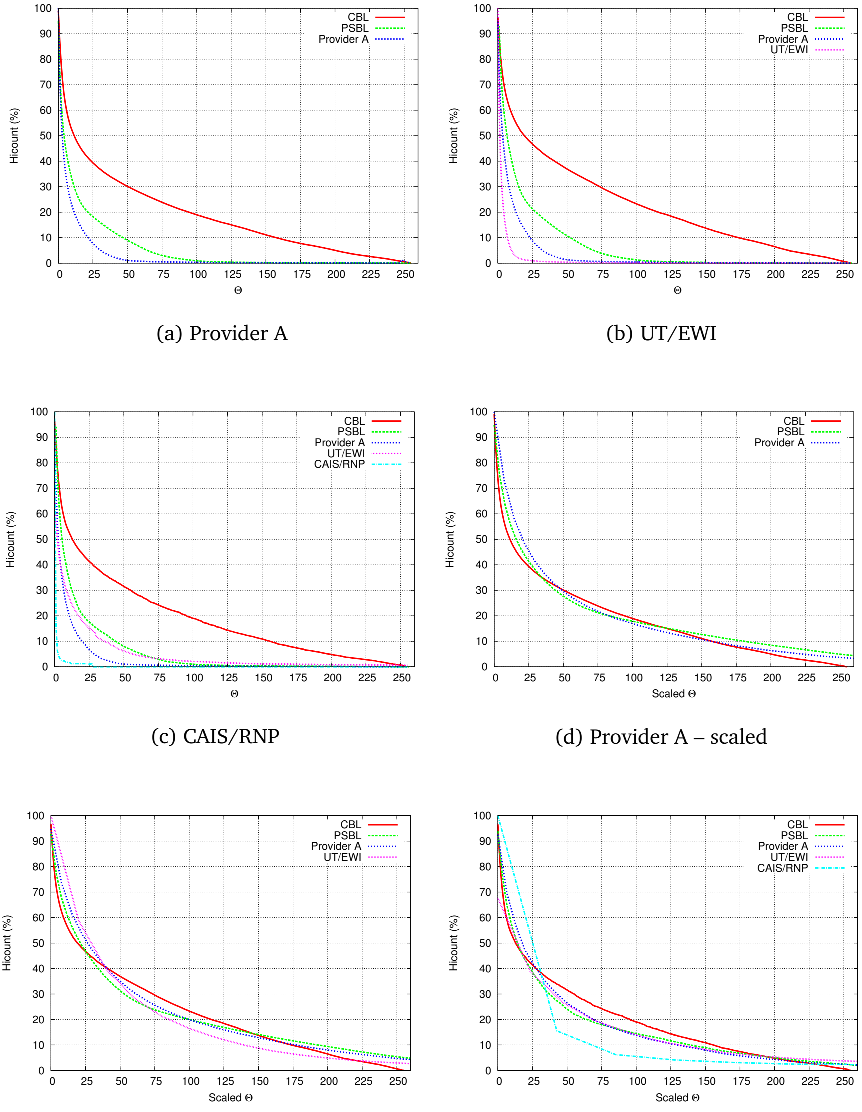
Again, we will only use a blacklist to protect a target if the blacklist is larger than the target's blacklist, i.e., we will not apply the UT/EWI blacklist to the Provider A mail server.

E.1.2 Experimental Results

Figures E.1(a), E.1(b), and E.1(c) show the obtained hitcounts (in percent) for filtering the Spam directed to Provider A, UT/EWI, and CAIS/RNP, respectively, as function of the threshold θ, using the different blacklists. The figures indicate that it is possible to effectively detect Spam messages based on different BadHood blacklists. This is especially true for large blacklists, like CBL, which always provides the highest hitcount. The figures also show that the hitcount decreases fast with increasing values of θ, a fact that most likely is due to the presence of high-volume spammers in the data sets.

A second insight provided by these results is that the value of θ should be adjusted to the considered BadHood blacklist. For the same θ, the hitcount values change considerably among BadHood blacklists. At first sight, this seems to suggest that the best choice for an administrator is the largest BadHood blacklist, just due to the fact that it has observed a higher number of spamming hosts. However, large BadHood Blacklists might suffers of drawbacks like a high number of irrelevant entries, as indicated in Section 6.3.2.

We investigate therefore if smaller BadHoods can still potentially provide similar hitcounts for appropriately chosen values of the threshold θ. Let θ_U



(a) Provider A

(b) UT/EWI

(c) CAIS/RNP

(d) Provider A – scaled

(e) UT/EWI – scaled

(f) CAIS/RNP – scaled

Figure E.1: Spam hitcount for varying values of the threshold θ in Fig. (a)-(c) and the scaled threshold in Fig.(d)-(f)

be the threshold that we have used to calculate the hitcount for the BadHood blacklist L_U. We choose the threshold θ_V for the list L_V as

$$\theta_V = \theta_U \cdot \frac{\sum_{B_i \in I_{U \cap V}} nHosts_V(B_i)}{\sum_{B_i \in I_{U \cap V}} nHosts_U(B_i)} \tag{E.2}$$

In Figures E.1(d), E.1(e), and E.1(f), we present the hitcounts obtained for the different targets using the rescaled θ defined in Eq. (E.2). For the L_{CBL} list, the threshold as indicated on the x-axis is used. For the other lists, we compute the rescaled theta according to Eq. (E.2), choosing $L_U = L_{CBL}$.

The figures show that, once the size factor is removed by using Eq. (E.2), all the considered BadHood blacklists can detect Spam with comparable performance. These results therefore indicate that Eq. (E.2) offers an operational way for choosing values of θ for different blacklists such that the blacklists are similarly effective in identifying Spam. In fact, one may be tempted to conclude from these results that all blacklists perform similarly independently of their size.

However, a different picture is obtained when calculating the number of legitimate mail traffic erroneously flagged as Spam, i.e., the number of false positives. Figure E.2 shows the percentage of legitimate mail messages received by the mail server of UT/EWI[1] that are labeled as Spam for varying values of the scaled threshold θ. While for CBL and PSBL the percentages of blocked Ham is less than 10% and rapidly falls to zero, for UT/EWI and Provider A we observe that up to 60% of legitimate mail would be labeled as Spam if a very low value of θ is chosen. On the other hand, also in the case of Provider A and UT/EWI, the percentage of blocked Ham is decreasing rapidly for increasing values of θ.

Our results highlight therefore a trade-off between (i) the size of the blacklist, (ii) the Spam hitcount and (iii) the percentage of blocked Ham. Very large lists, such as CBL and PSBL, achieve a high Spam hitcount with a low percentage of blocked Ham but contain a large number of irrelevant entries. In contrast, small and mid-sized lists, that is, Provider A and UT/EWI, contain much less irrelevant entries and can achieve Spam hitcounts comparable to those of the larger lists. However, for $\theta_{CBL} < 100$, a relatively high number of false positives can be expected.

[1] Only the UT/EWI data set provides information on Ham messages.

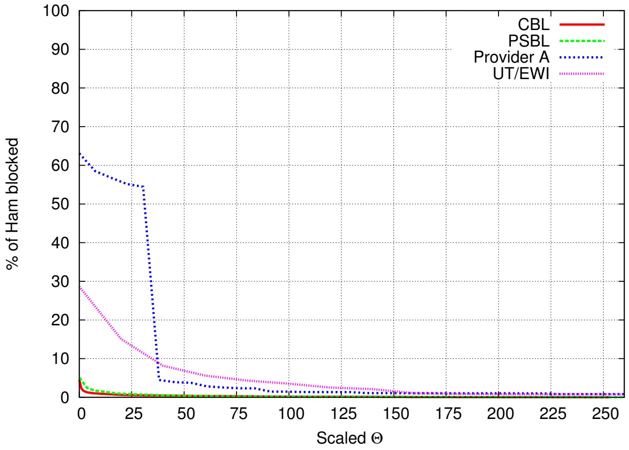


Figure E.2: Percentage of Ham erroneously blocked at UT/EWI, using the scaled θ

Bibliography

[1] National Atlas, “National Altas Home Page,” June 2012. [Online]. Available: http://www.nationalatlas.gov/mapmaker

[2] NYPD, “NYPD: Office of the Chief of Department - Crime Prevention – Crime Statistics,” August 2011. [Online]. Available: http://www.nyc.gov/html/nypd/html/crime_prevention/crime_statistics.shtml

[3] D. A. Wheeler and G. N. Larsen, “Techniques for cyber attack attribution,” Institute for Defense Analyses, Alexandria, VA, USA, Tech. Rep., 2003. [Online]. Available: http://oai.dtic.mil/oai/oai?verb=getRecord&metadataPrefix=html&identifier=ADA468859

[4] B. Krebs, “The Scrap Value of a Hacked PC,” May 2009. [Online]. Available: http://voices.washingtonpost.com/securityfix/2009/05/the_scrap_value_of_a_hacked_pc.html

[5] ——, “The Scrap Value of a Hacked PC, Revisited,” October 2012. [Online]. Available: http://krebsonsecurity.com/2012/10/the-scrap-value-of-a-hacked-pc-revisited/

[6] E. Ogg, “Internet van’ helped drive evolution of the Web,” 2012. [Online]. Available: http://news.cnet.com/Internet-van-helped-drive-evolution-of-the-Web/2100-1033_3-6217511.html

[7] Computer History Museum, “Computer History Museum and Web History Center Celebrate 30th Anniversary of Internet Milestone,” 2012. [Online]. Available: http://www.computerhistory.org/press/30th-anniversary-internet-milestone.html

[8] R. D. Smith, “The Dynamics of Internet Traffic: Self-Similarity, Self-Organization, and Complex Phenomena,” *ArXiv e-prints*, Jul. 2008.

[9] D. Krioukov, M. Kitsak, R. Sinkovits, D. Meyer, D. Rideout, and M. Boguna, “Network Cosmology,” arXiv:1203.2109, Tech. Rep., Mar 2012.

[10] W. Willingham, R. Govindan, S. Jamin, V. Paxson, and S. Shenker, “Scaling Phenomena in the Internet: Critically Examining Criticality,” in *Proceedings of the National Academy of ciences*, 2002.

[11] Central Intelligence Agency, “CIA: The World Factbook,” 2012. [Online]. Available: https://www.cia.gov/library/publications/the-world-factbook/rankorder/2184rank.html

[12] The World Bank, “Internet Users | Data | Table,” 2012. [Online]. Available: http://data.worldbank.org/indicator/IT.NET.USER/countries

[13] M. Castells, *The Internet Galaxy: Reflections on the Internet, Business, and Society*. New York, NY, USA: Oxford University Press, Inc., 2001.

[14] R. A. Clarke and R. Knake, *Cyber War: The Next Threat to National Security and What to Do About It*. New York, NY, USA: HarperCollins Publishers, 2010.

[15] Ian Traynor, “Russia accused of unleashing cyberwar to disable Estonia,” 2007. [Online]. Available: http://www.guardian.co.uk/world/2007/may/17/topstories3.russia

[16] Cisco Systems, “Cisco IronPort SenderBase Security Network,” 05 2012. [Online]. Available: http://www.senderbase.org/home/detail_spam_volume?displayed=last6months&action=&screen=&order=

[17] MAAWG, “Messaging Anti-Abuse Workign Group - E-mail Metrics Program: The Network Operator’s Perpective – Report # 15,” November 2011. [Online]. Available: http://www.maawg.org/sites/maawg/files/news/MAAWG_2011_Q1Q2Q3_Metrics_Report_15.pdf

[18] D. McCoy, A. Pitsillidis, G. Jordan, N. Weaver, C. Kreibich, B. Krebs, G. M. Voelker, S. Savage, and K. Levchenko, “PharmaLeaks: Understanding the Business of Online Pharmaceutical Affiliate Programs ,” in *Proceedings of the 21st USENIX Security Symposium*. Bellevue, Washington, USA: USENIX Association, August 2012.

[19] C. Kanich, C. Kreibich, K. Levchenko, B. Enright, G. M. Voelker, V. Paxson, and S. Savage, “Spamalytics: an empirical analysis of spam marketing conversion,” in *Proceedings of the 15th ACM conference on Computer and communications security*, ser. CCS ’08. New York, NY, USA: ACM, 2008, pp. 3–14.

[20] J. Soma, P. Singer, and J. Hurd, “SPAM Still Pays: The Failure of the CAN-SPAM Act of 2003 and Proposed Legal Solutions,” *Harv. J. on Legis.*, vol. 45, pp. 165–619, 2008.

[21] E. Cooke, F. Jahanian, and D. McPherson, “The zombie roundup: understanding, detecting, and disrupting botnets,” in *Proceedings of the Steps to Reducing Unwanted Traffic on the Internet Workshop*. Berkeley, CA, USA: USENIX Association, 2005, pp. 6–6.

[22] M. Bailey, E. Cooke, F. Jahanian, Y. Xu, and M. Karir, “A Survey of Botnet Technology and Defenses,” in *Proceedings of the 2009 Cybersecurity Applications & Technology Conference for Homeland Security*, ser. CATCH ’09. Washington, DC, USA: IEEE Computer Society, 2009, pp. 299–304.

[23] T. Werner, “Botnet Shutdown Success Story: How Kaspersky Lab Disabled the Hlux/Kelihos Botnet,” September 2011. [Online]. Available: http://www.securelist.com/en/blog/208193137/Botnet_Shutdown_Success_Story_How_Kaspersky_Lab_Disabled_the_Hlux_Kelihos_Botnet

[24] SurfNet, “We make innovation work,” 2012. [Online]. Available: http://www.surfnet.nl/en/Pages/default.aspx

[25] V. Fuller and T. Li, “RFC 4632: Classless Inter-domain Routing (CIDR): The Internet Address Assignment and Aggregation Plan,” August 2006. [Online]. Available: http://tools.ietf.org/html/rfc4632

[26] Wikipedia, the free encyclopedia, “CIDR notation,” June 2012. [Online]. Available: http://en.wikipedia.org/wiki/CIDR_notation

[27] A. Ramachandran and N. Feamster, “Understanding the Network-level Behavior of Spammers,” in *Proceedings of the 2006 Conference on Applications, Technologies, Architectures, and Protocols for Computer Communications*, ser. SIGCOMM ’06. New York, NY, USA: ACM, 2006, pp. 291–302.

[28] M. P. Collins, T. J. Shimeall, S. Faber, J. Janies, R. Weaver, M. De Shon, and J. Kadane, “Using Uncleanliness to Predict Future Botnet Addresses,” in *Proceedings of the 7th ACM SIGCOMM conference on Internet measurement*, ser. IMC ’07. New York, NY, USA: ACM, 2007, pp. 93–104.

[29] Z. Chen and C. Ji, “Optimal worm-scanning method using vulnerable-host distributions,” *Int. J. Secur. Netw.*, vol. 2, pp. 71–80, March 2007.

[30] Z. Chen, C. Ji, and P. Barford, “Spatial-Temporal Characteristics of Internet Malicious Sources,” in *INFOCOM 2008. The 27th Conference on Computer Communications. IEEE*, April 2008, pp. 2306 –2314.

[31] W. van Wanrooij and A. Pras, “Filtering Spam from Bad Neighborhoods,” *International Journal of Network Management*, vol. 20, no. 6, pp. 433–444, November 2010.

[32] M. K. Nozaki and H. F. Tipton, *Information Security Management Handbook, Sixth Edition, Volume V*, 6th ed. AUERBACH, Oct. 2011.

[33] J. Levine, “DNS Blacklists and Whitelists,” RFC 5782 (Informational), Internet Engineering Task Force, Feb. 2010.

[34] BBC News, “Inner-city areas on DHL blacklist ,” June 2005. [Online]. Available: http://news.bbc.co.uk/2/hi/uk_news/england/london/4108470.stm

[35] I. J. Tashevand, J. D. Couckuyt, N. W. Black, J. C. Krummand, R. Panabakerand, and M. L. Seltzer, “Pedestrian Route Production,” American Patent US 8 090 532, January 3rd, 2012.

[36] Chris Matyszczyk, “The joy of Microsoft’s ’avoid ghetto’ GPS patent,” January 2012. [Online]. Available: http://news.cnet.com/8301-17852_3-57354445-71/the-joy-of-microsofts-avoid-ghetto-gps-patent

[37] J. Hawkinson and T. Bates, “Guidelines for creation, selection, and registration of an Autonomous System (AS),” RFC 1930 (Best Current Practice), Internet Engineering Task Force, March 1996.

[38] J. R. Levine, “Experiences with greylisting,” in *In Second Conference on Email and Anti-Spam*, 2005.

[39] D. E. Sanger, “Obama Order Sped Up Wave of Cyberattacks Against Iran,” 2012. [Online]. Available: http://www.nytimes.com/2012/06/01/world/middleeast/obama-ordered-wave-of-cyberattacks-against-iran.html

[40] S. Deering and R. Hinden, “Internet Protocol, Version 6 (IPv6),” RFC 2460 (Specification), Internet Engineering Task Force, Dec. 1998.

[41] Internet2, “Internet2 NetFlow: Weekly Reports: Week of 20100426 ,” October 2010. [Online]. Available: http://netflow.internet2.edu/weekly/20100426/

[42] Amsterdam Internet Exchange, “Traffic Statistics,” July 2012. [Online]. Available: http://www.ams-ix.net/statistics/

[43] Arbor Networks, “7th annual worldwide infrastructure security report,” http://www.arbornetworks.com/report, February 2012.

[44] A. Ramachandran and N. Feamster, “Understanding the network-level behavior of spammers,” in *Proceedings of the 2006 conference on Applications, technologies, architectures, and protocols for computer communications*, vol. 36, no. 4. ACM New York, NY, USA, 2006, pp. 291–302.

[45] Brian Krebs, “Host of Internet Spam Groups Is Cut Off,” 2008. [Online]. Available: http://www.washingtonpost.com/wp-dyn/content/article/2008/11/12/AR2008111200658.html

[46] ——, “Spam Volumes Drop by Two-Thirds After Firm Goes Offline,” 2008. [Online]. Available: http://voices.washingtonpost.com/securityfix/2008/11/spam_volumes_drop_by_23_after.html

[47] S. Shin, R. Lin, and G. Gu, “Cross-Analysis of Botnet Victims: New Insights and Implications,” in *Proceedings of the 14th International Symposium on Recent Advances in Intrusion Detection (RAID’11))*, Sep. 2011.

[48] United States of America, “CAN-SPAM Act of 2003,” http://uscode.house.gov/download/pls/15C103.txt, 2003.

[49] European Union, “Directive on Privacy and Electronic Communications (2002/58),” July 2002. [Online]. Available: http://eur-lex.europa.eu/LexUriServ/LexUriServ.do?uri=CELEX:32002L0058:EN:NOT

[50] I. Asrar, “Android.Counterclank Found in Official Android Market,” http://www.symantec.com/connect/blogs/androidcounterclank-found-official-android-market, January 2012.

[51] S. M. Bellovin, “Security problems in the TCP/IP protocol suite,” *SIGCOMM Comput. Commun. Rev.*, vol. 19, no. 2, pp. 32–48, Apr. 1989.

[52] libpcap, “TCPDUMP/LIBPCAP public repository,” 2012. [Online]. Available: http://www.tcpdump.org/

[53] J. Quittek, T. Zseby, B. Claise, and S. Zander, “Requirements for IP Flow Information Export (IPFIX),” 2004.

[54] Snort, “Snort: A free lightweight network intrusion detection system for UNIX and Windows,” 2011. [Online]. Available: http://www.snort.org

[55] A. Patcha and J.-M. Park, “An overview of anomaly detection techniques: Existing solutions and latest technological trends,” *Comput. Netw.*, vol. 51, no. 12, pp. 3448–3470, Aug. 2007.

[56] A. Sperotto, R. Sadre, P.-T. de Boer, and A. Pras, “Hidden Markov Model Modeling of SSH Brute-Force Attacks,” in *Integrated Management of Systems, Services, Processes and People in IT*, ser. Lecture Notes in Computer Science, C. Bartolini and L. Gaspary, Eds. Springer Berlin / Heidelberg, 2009, vol. 5841, pp. 164–176.

[57] O’Reilly Commons, “SpamAssassin/SpamAssassin Rules,” July 2011. [Online]. Available: http://commons.oreilly.com/wiki/index.php/SpamAssassin/SpamAssassin_Rules

[58] N. Provos, “A virtual honeypot framework,” in *Proceedings of the 13th conference on USENIX Security Symposium - Volume 13*, ser. SSYM’04. Berkeley, CA, USA: USENIX Association, 2004, pp. 1–1.

[59] OSSEC, “Open Source Host-based Intrusion Detection System,” 2012. [Online]. Available: http://www.ossec.net/

[60] CBL, “Composite Blocking List,” 2012. [Online]. Available: http://cbl.abuseat.org/

[61] SpamAssassin, “The SpamAssassin Project,” 2012. [Online]. Available: http://spamassassin.apache.org/

[62] J. P. John, A. Moshchuk, S. D. Gribble, and A. Krishnamurthy, “Studying spamming botnets using botlab,” in *Proceedings of the 6th USENIX symposium on Networked systems design and implementation*, ser. NSDI’09. Berkeley, CA, USA: USENIX Association, 2009, pp. 291–306.

[63] V. Chandra and N. Shrivastava, “Ways to Evade Spam Filters and Machine Learning as a Potential Solution,” in *Communications and Information Technologies, 2006. ISCIT ’06. International Symposium on*, 18 2006-sept. 20 2006, pp. 268 –273.

[64] J. Franklin, V. Paxon, A. Perrig, and S. Savage, “An inquiry into the nature and causes of the wealth of Internet miscreants,” in *Proceedings of the 14th ACM conference on Computer and communications security*, ser. CCS ’07. New York, NY, USA: ACM, 2007, pp. 375–388.

[65] G. Moura, R. Sadre, and A. Pras, “Internet Bad Neighborhoods: The spam case,” in *Network and Service Management (CNSM), 2011 7th International Conference on*, oct. 2011, pp. 1 –8.

[66] G. C. M. Moura, R. Sadre, A. Sperotto, and A. Pras, “Internet Bad Neighborhoods Aggregation,” in *Network Operations and Management Symposium (NOMS), 2012 IEEE*, April 2012, pp. 343 –350.

[67] IANA, “IPv4 Address Space Registry,” 2012. [Online]. Available: http://www.iana.org/assignments/ipv4-address-space/ipv4-address-space.txt

[68] Oracle, “Class Random,” 2012. [Online]. Available: http://docs.oracle.com/javase/7/docs/api/java/util/Random.html#nextInt%28int%29

[69] James Fieser, “Ethics,” 2012. [Online]. Available: http://www.iep.utm.edu/ethics/

[70] L. P. Nathan, B. Friedman, P. Klasnja, S. K. Kane, and J. K. Miller, “Envisioning systemic effects on persons and society throughout interactive system design,” in *Proceedings of the 7th ACM conference on Designing interactive systems*, ser. DIS ’08. New York, NY, USA: ACM, 2008, pp. 1–10.

[71] N. Wiener, *Cybernetics, Second Edition: or the Control and Communication in the Animal and the Machine*. The MIT Press, Mar. 1965.

[72] IEEE, “IEEE Code of Ethics,” October 2012. [Online]. Available: http://www.ieee.org/about/corporate/governance/p7-8.html

[73] E. A. Buchanan and M. Zimmer, “Internet research ethics,” in *The Stanford Encyclopedia of Philosophy*, E. N. Zalta, Ed., 2012.

[74] The Spamhaus Project, 2012. [Online]. Available: http://www.spamhaus.org

[75] Passive Spam Block List, 2011. [Online]. Available: http://psbl.surriel.com/

[76] F. Soldo, A. Le, and A. Markopoulou, “Blacklisting recommendation system: Using spatio-temporal patterns to predict future attacks,” *Selected Areas in Communications, IEEE Journal on*, vol. 29, no. 7, pp. 1423 –1437, august 2011.

[77] S. Floyd and M. Allman, “Comments on the Usefulness of Simple Best-Effort Traffic,” RFC 5290 (Informational), Internet Engineering Task Force, Jul. 2008.

[78] RIPE NCC, “PI Assignment Size ,” July 2006. [Online]. Available: http://www.ripe.net/ripe/policies/proposals/2006-05

[79] V. Fuller, T. Li, J. Yu, and K. Varadhan, “Supernetting: an Address Assignment and Aggregation Strategy,” RFC 1338 (Informational), Internet Engineering Task Force, Jun. 1992.

[80] Y. Rekhter, T. Li, and S. Hares, “A Border Gateway Protocol 4 (BGP-4),” RFC 4271 (Draft Standard), Internet Engineering Task Force, Jan. 2006, updated by RFC 6286.

[81] T. Bates, “CIDR Report,” November 2012. [Online]. Available: http://www.cidr-report.org/as2.0/#General_Status

[82] K. Hubbard, M. Kosters, D. Conrad, D. Karrenberg, and J. Postel, “Internet Registry IP Allocation Guidelines,” RFC 2050 (Best Current Practice), Internet Engineering Task Force, Nov. 1996.

[83] Internet Assigned Numbers Authority, 2012. [Online]. Available: http://www.iana.org/

[84] ICANN, “Internet Corporation for Assigned Names and Numbers,” 2012. [Online]. Available: http://www.icann.org

[85] IANA, “IPv6 Global Unicast Address Assignments,” 2012. [Online]. Available: http://www.iana.org/assignments/ipv6-unicast-address-assignments/ipv6-unicast-address-assignments.txt

[86] African Network Information Centre (AfriNIC), 2012. [Online]. Available: http://www.afrinic.net/

[87] American Registry for Internet Numbers (ARIN), 2012. [Online]. Available: https://www.arin.net/

[88] Asia-Pacific Network Information Centre (APNIC), 2012. [Online]. Available: http://www.apnic.net/

[89] LACNIC - Latin American and Caribbean Internet Addresses Registry, 2012. [Online]. Available: http://www.lacnic.net/

[90] Reseaux IP Europeens Network Coordination Centre (RIPE NCC), 2012. [Online]. Available: http://www.ripe.net/

[91] B. Irwin and N. Pilkington, “High Level Internet Scale Traffic Visualization Using Hilbert Curve Mapping.” in *VizSEC*, ser. Mathematics and Visualization. Springer, 2007, pp. 147–158.

[92] L. Daigle, “WHOIS Protocol Specification,” RFC 3912 (Draft Standard), Internet Engineering Task Force, Sep. 2004.

[93] IANA, “Autonomous System (AS) Numbers,” 2012. [Online]. Available: http://www.iana.org/assignments/as-numbers/as-numbers.txt

[94] Team Cymru Community Services, “Ip to asn mapping,” 2012. [Online]. Available: http://www.team-cymru.org/Services/ip-to-asn.html

[95] Maxmind, “GeoLite Autonomous System Number Database,” 2012. [Online]. Available: http://www.maxmind.com/app/asnum

[96] ——, “GeoIP Organization Database,” 2012. [Online]. Available: http://www.maxmind.com/app/organization

[97] G. Huston, “Opinion: The ISP – The Uncommon Carrier,” *The Internet Protocol Journal*, vol. 5, no. 3, pp. 23–27, 2002. [Online]. Available: http://www.cisco.com/web/about/ac123/ac147/archived_issues/ipj_5-3/uncommon_carrier.html

[98] J. A. Muir and P. C. V. Oorschot, “Internet geolocation: Evasion and counterevasion,” *ACM Comput. Surv.*, vol. 42, pp. 4:1–4:23, December 2009.

[99] I. Poese, S. Uhlig, M. A. Kaafar, B. Donnet, and B. Gueye, “IP Geolocation Databases: Unreliable?” *SIGCOMM Comput. Commun. Rev.*, vol. 41, no. 2, pp. 53–56, Apr. 2011.

[100] Hulu, “Hulu - What your favorites. Anytime. For free.” accessed on February 2011. [Online]. Available: http://www.hulu.com

[101] W. E. Sobel and B. McCorkendale, “Use of Geo-Location Data for Spam Detection,” U.S. Patent #7,366,919 issued Apr. 29, 2008, filed 2003. [Online]. Available: http://www.google.com/patents/US8001598

[102] I. Poese, S. Uhlig, M. A. Kaafar, B. Donnet, and B. Gueye, “IP Geolocation Databases: Unreliable?” Deutsche Telekom Lab./TU Berlin, Germany/INRIA Rhone-Alpes, France//Universite catholique de Louvain, Belgium//Universite Cheikh Anta Diop de Dakar, Senegal, Tech. Rep., 2011. [Online]. Available: http://www.net.t-labs.tu-berlin.de/papers/PKDGU-IGDU-11.pdf

[103] Maxmind, “Maxmind,” 2012. [Online]. Available: http://www.maxmind.com/

[104] ——, “Geolite city accuracy,” accessed on August 2012. [Online]. Available: http://www.maxmind.com/app/geolite_city_accuracy

[105] ——, “GeoIP City Database,” 2012. [Online]. Available: http://www.maxmind.com/app/city

[106] PhishTank, “PhishTank: Join the Fight Against Phishing,” 2012. [Online]. Available: http://www.phishtank.com

[107] Google, “Google Public DNS,” 2012. [Online]. Available: https://developers.google.com/speed/public-dns/

[108] Hurricane Electric, “BGP Toolkit,” 2012. [Online]. Available: http://bgp.he.net/

[109] European Internet Exchange Association, “ASNs present at 10 or more IXPs,” August 2012. [Online]. Available: https://www.euro-ix.net/tools/asn_common

[110] International Telecommunications Unions (ITU), “Percentage of Individuals using the Internet 2000-2010,” 2012. [Online]. Available: http://www.itu.int/ITU-D/ict/statistics/material/excel/2010/IndividualsUsingInternet_00-10.xls

[111] M. Aguiar, V. Boutenko, D. Michael, V. Rastogi, A. Subramanian, and Y. Zhou, “The Internet’s New Billion: Digital Consumers in Brazil, Russia, India, China, and Indonesia,” The Boston Consulting Group, Tech. Rep., 2010. [Online]. Available: http://www.bcg.com/documents/file58645.pdf

[112] United Nations, “World Population Prospects, the 2010 Revision,” 2010. [Online]. Available: http://esa.un.org/unpd/wpp/Excel-Data/population.htm

[113] CIA, “The World Fact Book - Appendix B: International Organizations and Groups,” accessed on August 2012. [Online]. Available: https://www.cia.gov/library/publications/the-world-factbook/appendix/appendix-b.html

[114] OpenNet Initiative, “OpenNet Initiative,” 2012. [Online]. Available: http://opennet.net/

[115] ——, “Saudi Arabia,” 2009. [Online]. Available: http://opennet.net/research/profiles/saudi-arabia

[116] Reporters Without Borders, “Internet Enemies Report 2012,” 2012. [Online]. Available: http:/en.rsf.org/IMG/pdf/rapport-internet2012_ang.pdf

[117] OpenNet Initiative, “China,” 2009. [Online]. Available: http://opennet.net/research/profiles/china

[118] ——, “Belarus,” 2010. [Online]. Available: http://opennet.net/research/profiles/belarus

[119] ——, “Kazakhstan,” 2010. [Online]. Available: http://opennet.net/research/profiles/kazakhstan

[120] ——, “Vietnam,” 2007. [Online]. Available: http://opennet.net/research/profiles/vietnam

[121] ——, “Tunisia,” 2009. [Online]. Available: http://opennet.net/research/profiles/tunisia

[122] A. Pathak, Y. C. Hu, and Z. M. Mao, “Peeking into Spammer Behavior from a Unique Vantage Point,” in *Proceedings of the 1st Usenix Workshop on Large-Scale Exploits and Emergent Threats*. Berkeley, CA, USA: USENIX Association, 2008.

[123] L. Zhuang, J. Dunagan, D. R. Simon, H. J. Wang, and J. D. Tygar, “Characterizing Botnets from Email Spam Records,” in *Proceedings of the 1st Usenix Workshop on Large-Scale Exploits and Emergent Threats*. Berkeley, CA, USA: USENIX Association, 2008.

[124] M. van Polen, G. C. M. Moura, and A. Pras, “Finding and Analyzing Evil Cities on the Internet,” in *Proceedings of the 5th International Conference on Autonomous Infrastructure, Management and Security (AIMS 2011)*, vol. 6734. Nancy, France: Springer Verlag, 2011.

[125] Quarantainenet B.V., “Quarantainenet,” http://quarantainenet.com/, accessed on February 2011.

[126] Google, “Visualization: Geochart,” 2012. [Online]. Available: https://google-developers.appspot.com/chart/interactive/docs/gallery/g%eochart

[127] Maxmind, “MaxMind World Cities with Population,” 2012. [Online]. Available: http://www.maxmind.com/app/worldcities

[128] Y. Jiang, N. Zhang, and B. Fang, "An email geographic Path-Based technique for spam filtering," in *2007 International Conference on Computational Intelligence and Security*, 2007, pp. 750–753. [Online]. Available: 10.1109/CIS.2007.94

[129] Akamai, "The State of the Internet, 3rd Quarter, 2010," Akamai. Available online at: http://www.akamai.com/stateoftheinternet/, accessed on February 2011, Tech. Rep., 2010.

[130] Quarantainenet B.V., "Virus attacks," http://quarantainenet.com/?language=en; page=infections, accessed on February 2011.

[131] H. Koike, K. Ohno, and K. Koizumi, "Visualizing cyber attacks using IP matrix," in *IEEE Workshops on Visualization for Computer Security*, vol. 0. Los Alamitos, CA, USA: IEEE Computer Society, 2005, p. 11.

[132] Sophos, "Only one in 28 emails legitimate," http://www.sophos.com/pressoffice/news/articles/2008/07/dirtydozjul08.html, June 2008.

[133] Spamhaus, "Effective spam filtering," http://www.spamhaus.org/effective_filtering.html, January 2010.

[134] J. Makey, "Blacklists Compared," 2012. [Online]. Available: http://www.sdsc.edu/~jeff/spam/Blacklists_Compared.html

[135] Blacklist Statistics Center, 2011. [Online]. Available: http://stats.dnsbl.com/

[136] Thunderbird, "Junk Mail Controls," 2012. [Online]. Available: http://kb.mozillazine.org/Junk_Mail_Controls

[137] A. Sperotto, G. Vliek, R. Sadre, and A. Pras, "Detecting spam at the network level," in *Proceedings of the 15th Open European Summer School and IFIP TC6.6 Workshop, EUNICE 2009, Barcelona*, ser. Lecture Notes in Computer Science, vol. 5733. Berlin: Springer Verlag, August 2009, pp. 208–216.

[138] A. Sperotto, G. Vliek, R. Sadre, and A. Pras, "Detecting spam at the network level," in *Proceedings of the 15th Open European Summer School and IFIP TC6.6 Workshop on The Internet of the Future*, ser. EUNICE '09. Berlin, Heidelberg: Springer-Verlag, 2009, pp. 208–216.

[139] A. Sperotto, G. Schaffrath, R. Sadre, C. Morariu, A. Pras, and B. Stiller, "An Overview of IP Flow-Based Intrusion Detection," *Communications Surveys Tutorials, IEEE*, vol. 12, no. 3, pp. 343 –356, 2010.

[140] CBL, "Spam Trap Flow Statistics," 2012. [Online]. Available: http://cbl.abuseat.org/totalflow.html

[141] PSBL FAQ, 2011. [Online]. Available: http://psbl.surriel.com/faq/

[142] UCEPROTECT-Network - Germany's first Spam protection database, 2011. [Online]. Available: http://www.uceprotect.net/en/

[143] The Spamhaus Block List, 2011. [Online]. Available: http://www.spamhaus.org/sbl/

[144] CAIS, "Security Incident Response Team (In Portuguese: *Centro de Atendimento a Incidentes de Segurança*," 2012. [Online]. Available: http://www.rnp.br/en/cais/

[145] Comtouch, "July 2011 - Internet Threats Trend Report," July 2011. [Online]. Available: http://www.commtouch.com/download/2085

[146] Dick Craddock, "Hey! My friend's account was hacked!" 2011. [Online]. Available: http://windowsteamblog.com/windows_live/b/windowslive/archive/2011/07/14/hey-my-friend-s-account-was-hacked.aspx

[147] D. C. Montgomery and G. C. Runger, *Applied Statistics and Probability for Engineers, 4th Edition, and JustAsk! Set*, 4th ed. John Wiley & Sons, May 2006.

[148] C. Kreibich, C. Kanich, K. Levchenko, B. Enright, G. M. Voelker, V. Paxson, and S. Savage, "On the Spam Campaign Trail," in *Proceedings of the 1st Usenix Workshop on Large-Scale Exploits and Emergent Threats*. Berkeley, CA, USA: USENIX Association, 2008.

[149] A. Pitsillidis, C. Kanich, G. M. Voelker, K. Levchenko, , and S. Savage, "Taster's Choice: A Comparative Analysis of Spam Feeds," in *ACM SIGCOMM Conference on Internet Measurement (IMC)*, 2012, pp. 427–440.

[150] UnifiedeMail, "Multi-RBL Check ," May 2012. [Online]. Available: http://www.unifiedemail.net/Tools/RBLCheck/Default.aspx

[151] DSHIELD.org, "dshield Home | DShield; Cooperative Network Security Community - Internet Security," May 2012. [Online]. Available: http:/www.dshield.org

[152] ——, "About the Internet Storm Center| DShield; Cooperative Network Security Community - Internet Security," May 2012. [Online]. Available: http://www.dshield.org/about.html

[153] SANS, "SANS Information, Network, Computer Security Training, Research, Resources," May 2012. [Online]. Available: http://www.sans.org

[154] J. Zhang, P. A. Porras, and J. Ullrich, "Highly predictive blacklisting," in *USENIX Security Symposium*, P. C. van Oorschot, Ed. USENIX Association, 2008, pp. 107–122.

[155] F. Soldo, A. Le, and A. Markopoulou, "Predictive blacklisting as an implicit recommendation system," in *Proceedings of the 29th conference on Information communications*, ser. INFOCOM'10. Piscataway, NJ, USA: IEEE Press, 2010, pp. 1640–1648.

[156] SSHBL.org, "sshbl.org - (the SSH blacklist)," 2012. [Online]. Available: http://www.sshbl.org/

[157] UT/EWI, "Faculty of Electrical Engineering, Mathematics and Computer Science," 2012. [Online]. Available: http://www.utwente.nl/en/education/eemcs/

[158] G. C. M. Moura, A. Sperotto, R. Sadre, and A. Pras, “Evaluating Third-Party Bad Neighborhood Blacklists for Spam Detection (to appear),” in *IFIP/IEEE International Symposium on Integrated Network Management (IM 2013)*, Ghent, Belgium, May 2013.

[159] R, “The R Project for Statistical Computing,” 2012. [Online]. Available: http://www.r-project.org/

[160] J. Zhang, P. Porras, and J. Ullrich, “Gaussian Process Learning for Cyber-Attack Early Warning,” *Stat. Anal. Data Min.*, vol. 3, pp. 56–68, February 2010.

[161] DSHIELD.org, “How To Submit Your Firewall Logs To DShield| DShield; Cooperative Network Security Community - Internet Security,” May 2012. [Online]. Available: http://www.dshield.org./howto.html

[162] IANA, “Assigned Internet Protocol Numbers,” 2012. [Online]. Available: http://www.iana.org/assignments/protocol-numbers/protocol-numbers.txt

[163] ——, “ Service Name and Transport Protocol Port Number Registry,” 2012. [Online]. Available: http://www.iana.org/assignments/service-names-port-numbers/service-names-port-numbers.txt

[164] Microsoft, “Active Directory Domain Services,” 2012. [Online]. Available: http://technet.microsoft.com/en-us/library/cc770946%28v=ws.10%29.aspx

[165] ——, “What’s New in Terminal Services for Windows Server 2008,” 2012. [Online]. Available: http://technet.microsoft.com/en-us/library/cc733093%28WS.10%29.aspx

[166] CBL, “The CBL FAQ,” 2012. [Online]. Available: http://cbl.abuseat.org/faq.html

[167] M. Sahami, S. Dumais, D. Heckerman, and E. Horvitz, “A bayesian approach to filtering junk e-mail,” in *AAAI’98 Workshop on Learning for Text Categorization*, 1998.

[168] I. Androutsopoulos, J. Koutsias, K. Chandrinos, G. Paliouras, and C. D. Spyropoulos, “An evaluation of naive bayesian anti-spam filtering,” *CoRR*, vol. cs.CL/0006013, 2000.

[169] I. Androutsopoulos, J. Koutsias, K. V. Chandrinos, and C. D. Spyropoulos, “An experimental comparison of naive bayesian and keyword-based anti-spam filtering with personal e-mail messages,” in *Proceedings of the 23rd annual international ACM SIGIR conference on Research and development in information retrieval*, ser. SIGIR ’00. New York, NY, USA: ACM, 2000, pp. 160–167.

[170] Rik Ferguson, “The History Of The Botnet: Part 1,” Sep. 2010. [Online]. Available: http://www.businesscomputingworld.co.uk/the-history-of-the-botnet-part-i/

[171] ——, “The History Of The Botnet: Part 2,” Sep. 2010. [Online]. Available: http://www.businesscomputingworld.co.uk/the-history-of-the-botnet-part-ii/

[172] R. W. Mark Dutz, Jonathan Orszag, "The Substantial Consumer Benefits of Broadband Connectivity of U.S. Households," http://internetinnovation.org/files/special-reports/CONSUMER_BENEFITS_OF_BROADBAND.pdf, July 2009.

[173] InfoSecurity.com, "BredoLab downed botnet linked with Spamit.com," November 2010. [Online]. Available: http://www.infosecurity-magazine.com/view/13620/bredolab-downed-botnet-linked-with-spamitcom

[174] Number Resource Organization, "Free pool of ipv4 address space depleted," http://www.nro.net/news/ipv4-free-pool-depleted, February 2011.

[175] R. Hinden and S. Deering, "IP Version 6 Addressing Architecture," RFC 4291 (Draft Standard), Internet Engineering Task Force, Feb. 2006, updated by RFCs 5952, 6052.

[176] T. Narten, G. Huston, and L. Roberts, "IPv6 Address Assignment to End Sites," RFC 6177 (Best Current Practice), Internet Engineering Task Force, Mar. 2011.

About the Author

Giovane César Moreira Moura was born in Goiânia, the capital of the state of Goiás, Brazil, in 1981. He has obtained a degree in Computer Engineering from the Federal University of Goiás (UFG), in 2006.

His first steps on the network management field were taken while in the Engineering School, during his internship at the Point-of-Presence (PoP) of the Brazilian Research Network (*Rede Nacional de Ensino e Pesquisa – RNP.br*) at his home state. During the internship, Giovane also worked in a project to evaluate the feasibility of a Internet Exchange Point (IX), by assessing whether there was significant traffic exchange between POP-GO/RNP and the biggest telecom operator in the state, using Cisco's Netflow technology.

Following his graduation, Giovane chose to continue on the network management field, and applied for a master position one at the the Computer Networks group at the Federal University of Rio Grande do Sul (UFRGS), in Porto Alegre – 2000 km far from his home state. After being admitted, Giovane worked under the supervision of Luciano Gaspary, focusing on Web services for network management and the complexity of security procedures on IT infrastructures. During his masters, Giovane co-authored (as first author) two articles in two of the most prestigious network management conferences – IM and NOMS, and presented both of them.

After obtained his master's degree, Giovane then applied for a PhD position at the Design and Analysis of Communication Systems (DACS) group from the University of Twente, to work with Aiko Pras – who had research ties with professors Luciano Gaspary and Lisandro Granville from UFRGS. Giovane then moved from Porto Alegre to Enschede to start his PhD project in September, 2008. He worked in the FP6's European Network of Excellence on Management Solutions for Next Generation Networks (EMANICS) project, focusing on network security. During his first year of the PhD, Giovane also visited Prof. Burkhard Stiller's Communications System Group (CSG) at the University of Zurich, Switzerland, also in the context of the EMANICS project (March-April 2009).

During his PhD project – Internet Bad Neighborhoods – Giovane has co-authored

11 papers (see Appendix A for the complete list), including in the most prestigious network management conferences – CNSM 2011, IM 2013, NOMS 2012, and IPOM 2009. In addition, he has been involved with other projects (optical switching, Ethernet and network management, phishing, etc.) and supervised several bachelor and master students. Also, some of his research work has drawn attention from some media outlets. He has also developed collaboration with industry partners – including Quarantainenet, SurfNet, and RNP. Giovane has also served as a reviewer for several journals and conferences in his field, and, during his spare time, he has also been involved and served as president in the PhD Network of the University of Twente (P-NUT), where he met many of his friends and has learned several non-technical skills.

His current research interests include developing Bad Neighborhood-based security solutions that employ the main findings providing in this. Giovane has also interest in network traffic analysis and network management, and a special interest in the emerging field of cyber-war and its developments. Giovane can be reached at giovane@gmail.com.