Linköping Studies in Science and Technology

Dissertations. No. 1798

Collaborative Network Security

Targeting Wide-area Routing and Edge-network Attacks

by

Rahul Gokulchand Hiran

Department of Computer and Information Science
Linköping University
SE-581 83 Linköping, Sweden

Linköping 2016

Copyright © 2016 Rahul Hiran

ISBN 978-91-7685-662-8

ISSN 0345-7524

Cover art together with Per Lagman

Printed by LiU Tryck 2016

URL: <http://urn.kb.se/resolve?urn=urn:nbn:se:liu:diva-131959>

Abstract

To ensure that services can be delivered reliably and continuously over the Internet, it is important that both Internet routes and edge networks are secured. However, the sophistication and distributed nature of many attacks that target wide-area routing and edge networks make it difficult for an individual network, user, or router to detect these attacks. Therefore collaboration is important. Although the benefits of collaboration between different network entities have been demonstrated, many open questions still remain, including how to best design distributed scalable mechanisms to mitigate attacks on the network infrastructure. This thesis makes several contributions that aim to secure the network infrastructure against attacks targeting wide-area routing and edge networks.

First, we present a characterization of a controversial large-scale routing anomaly, in which a large Telecom operator hijacked a very large number of Internet routes belonging to other networks. We use publicly available data from the time of the incident to understand what can be learned about large-scale routing anomalies and what type of data should be collected in the future to diagnose and detect such anomalies.

Second, we present multiple distributed mechanisms that enable collaboration and information sharing between different network entities that are affected by such attacks. The proposed mechanisms are applied in the contexts of collaborating Autonomous Systems (ASes), users, and servers, and are shown to help raise alerts for various attacks. Using a combination of data-driven analysis and simulations, based on publicly available real network data (including traceroutes, BGP announcements, and network relationship data), we show that our solutions are scalable, incur low communication and processing overhead, and provide attractive tradeoffs between attack detection and false alert rates.

Finally, for a set of previously proposed routing security mechanisms, we consider the impact of regional deployment restrictions, the scale of the collaboration, and the size of the participants deploying the solutions. Although regional deployment can be seen as a restriction and the participation of large networks is often desirable, we find interesting cases where regional deployment can yield better results compared to random global deployment, and where smaller networks can play an important role in achieving better security gains. This study offers new insights towards incremental deployment of different classes of routing security mechanisms.

This work was supported by the Swedish National Graduate School of Computer Science (CUGS) and the Internet Foundation in Sweden (IIS).

Populärvetenskaplig sammanfattning

Internet och dess tjänster är mycket exponerade för attacker. Många av de kritiska protokoll och mekanismer som behövs för att leverera tjänster över internet designades för flera decennier sedan. Dessa protokoll och mekanismer är starkt beroende av tillit mellan olika nätverkskomponenter, såsom routers och servrar. Den explosionsartade tillväxten av internetanvändning har dock lett till att angripare börjat utnyttja denna underförstådda tillit mellan komponenter. Till exempel Border Gateway Protocol, som används för att avgöra vilken väg data på internet skall ta, tillåter att vem som helst kan påstå att en viss väg existerar. Därigenom kan angripare avlyssna och manipulera information som skickas över internet.

För att säkerställa att tjänster kan levereras på ett tillförlitligt sätt över internet är det viktigt både att rutten är korrekt och att slutanvändarnätverket är säkrat. Eftersom angreppen ofta är avancerade och distribuerade är det svårt för en enskild nätverksoperatör att upptäcka angrepp. Samarbete är därför viktigt för att upptäcka och skydda mot sådana angrepp. Trots att fördelar med samarbete mellan nätverksoperatörer har påvisats återstår många utmaningar. En sådan utmaning består i att designa distribuerade skalbara mekanismer för att förhindra angrepp mot nätverksinfrastruktur. I denna avhandling föreslås och utvärderas flera sätt att skydda valet av rutter samt slutanvändarnätverk.

För att förstå omfattning och vilka tekniker som används för att utföra angrepp mot nätverksinfrastruktur presenterar vi först en beskrivande studie av en storskalig incident där China Telecoms nätverkssystem felaktigt påstod att de var slutdestination för en betydande del av trafiken på internet. Detta ledde till att mycket internettrafik omdirigerades till China Telecom, inklusive trafik ämnad för amerikanska försvarsorganisationer. Det som utmärkte denna incident var att China Telecom hade bandbredd nog att i sin tur leverera trafik till den korrekta destinationen, vilket gjorde incidenten svår att upptäcka. Vi har studerat konsekvenserna av denna incident och undersökt vilka förutsättningar som gjorde den storskaliga avledningen av trafik möjlig.

Vi föreslår och utvärderar flera mekanismer som tillåter samarbete mellan olika nätverksoperatörer, och som möjliggör upptäckt av försök att omdirigera rutter. De föreslagna metodernas lämplighet för att upptäcka angrepp mot nätverk undersöks genom omfattande simuleringar. Vi visar på en fördelaktig balans mellan antal rapporterade avvikelser och nödvändiga systemresurser.

Slutligen undersöker vi hur säkerhetsvinsterna av tidigare föreslagna mekanismer påverkas när tillämpningen av dessa mekanismer begränsas, till exempel geografiskt. Genom att jämföra den begränsade tillämpningen med

den obegränsade visar vi hur effektiv en implementation i till exempel endast EU-området skulle kunna vara. Vi undersöker hur antal samarbetande nätverk, samt nätverkens storlek, påverkar möjligheten att upptäcka och förebygga angrepp. Vi undersöker även våra egna protokoll för att demonstrera nyttan av dessa i både stor och liten skala. Vi visar att våra mekanismer kan hjälpa till att skydda användares trafik, inte bara i regionen där de implementeras, utan även globalt.

Acknowledgements

First I would like to thank my primary supervisor, Professor Nahid Shahmehri for giving me the opportunity to do research in the interesting area of security. She has played a vital role in shaping my research work. Not only has she been tireless in planning for my research, she has helped me with smaller but important aspects such as presentation skills, writing, listening, questioning, and improving my research ideas. She has helped me improve my research skills in so many ways.

I would also like to take this opportunity to thank my co-supervisor, Associate Professor Niklas Carlsson. I truly enjoy working with Niklas. He has been an inspiration and a role model as a researcher. He has been instrumental in helping me choose interesting research topics and research questions. Our discussions on research questions and their possible solutions are always interesting. I can ask him, without hesitation, the simplest or the hardest of questions, and always receive helpful tips.

During my PhD studies I have had the opportunity to work with Assistant Professor Phillipa Gill. Working with Phillipa helped me increase my knowledge and experience greatly. I am thankful to her for the collaboration. I would also like to thank Dr. David Byers with whom I worked during the initial period of my PhD. I thank Brittany Shahmehri for thorough proof-reading of the thesis. I would also like to thank Anne Moe for all the help with administrative matters.

I thank all current and former members of ADIT for their friendship, support, and all the valuable comments during numerous ADIT meetings. I would also like to express my gratitude to members of the badminton group at Campushallen and IDA. I enjoyed training with them, which helped improve my badminton skills and kept me healthy.

I express my gratitude to my wife, Vaishali, who has positively impacted my studies. She has been a constant source of support and, when needed, a pleasant distraction.

Finally, I thank my family for supporting and trusting the decisions that I make in life.

Rahul Hiran
September 2016
Linköping, Sweden

Dedicated to

Riddhansh, Shashwati, Divya, Grishma, Sakshi, and Shraddha

Contents

1	Introduction	1
1.1	Cybercrime	1
1.1.1	Incidents of cybercrimes	2
1.1.2	Reasons networks and users are vulnerable to cyber-attacks	4
1.2	Cybercrime classification and thesis focus	5
1.3	Network-centric attacks	7
1.3.1	Incidents of network-centric attacks	8
1.3.2	Factors contributing to network-centric attacks	10
1.4	Problem formulation	11
1.5	Contributions	14
1.5.1	Study of large-scale routing anomaly	14
1.5.2	Collaboration among network entities to detect attacks	14
1.5.3	Effect of scale, size, and locality	15
1.6	Methodology	16
1.6.1	Characterization and empirical observations	16
1.6.2	Evaluation of large-scale systems	16
1.6.3	Working with large datasets	18
1.7	List of publications	20
1.8	Thesis organization	21
2	Background and Related Work	23
2.1	Internet routing	23
2.1.1	Autonomous systems and addressing	24
2.1.2	Routing on the Internet	24
2.1.3	Control-plane and data-plane	25
2.1.4	Intra-domain routing protocol	26
2.2	BGP and inter-domain routing	27
2.2.1	Reachability information	27
2.2.2	Route selection	28
2.2.3	Route aggregation and longest prefix matching routing	31
2.3	Security issues with BGP	32
2.4	Countering threats to BGP	35
2.4.1	Route filtering	35
2.4.2	Crypto-based architectures for BGP security	36
2.4.3	Anomaly detection based techniques	41
2.5	Attacks involving edge networks	43
2.5.1	Spamming	44

2.5.2	Scanning	44
2.6	Peer-to-peer networks	45
2.7	Intrusion detection	47
2.7.1	Centralized IDS	47
2.7.2	Hierarchical IDS	48
2.7.3	Decentralized IDS	49
2.8	Discussion	50
3	A Characterization Study of the China Telecom Incident	53
3.1	Introduction	53
3.1.1	Insecurity of the Internet's routing system	54
3.2	Related work	55
3.3	Methodology	55
3.3.1	China Telecom's network	56
3.3.2	Control-plane measurements	56
3.3.3	Data-plane measurements	57
3.3.4	Limitations	57
3.4	Impact of the China Telecom hijack	58
3.4.1	What is the geographic distribution of the announced prefixes?	58
3.4.2	Which organizations were most impacted?	59
3.4.3	Were any of the announcements subprefix hijacks?	59
3.5	The mechanics of interception	60
3.5.1	How was interception possible?	60
3.5.2	How many ISPs chose to route to China Telecom?	61
3.5.3	Which prefixes were intercepted?	62
3.5.4	Why <i>didn't</i> neighboring ASes route to China Telecom?	63
3.6	Conclusion	64
3.6.1	Key observations	64
3.6.2	Discussion	64
4	PrefiSec: A Distributed Framework for Collaborative Security Information Sharing	67
4.1	Introduction	67
4.2	System Overview	69
4.2.1	Distributed overlay framework	69
4.2.2	PrefiSec services	71
4.2.3	Distributed information sharing and aggregation	71
4.2.4	Local repository and optimizations	72
4.3	Overlay Structures	72
4.3.1	Distributed prefix registry	72
4.3.2	Replication and load balancing	75
4.3.3	Lookup optimizations	76
4.3.4	Overhead analysis	76
4.4	Service implementation	78
4.4.1	Basic services	78

4.4.2	High level services	79
4.4.3	Incentive-based hierarchy	80
4.5	Conclusions	80
5	PrefiSec-based BGP Monitoring	83
5.1	Introduction	83
5.2	Prefix and subprefix hijacks	85
5.2.1	Policy overview	85
5.2.2	Case-based overhead analysis	87
5.3	Interception attack	91
5.3.1	Policy overview	91
5.3.2	AS relationship information sharing	92
5.3.3	Case-based analysis	94
5.4	Prefix-based information sharing	101
5.5	Conclusions	102
6	TRAP: Open Decentralized Distributed Spam Filtering	103
6.1	Introduction	103
6.2	The TRAP system	104
6.2.1	Requirements	105
6.2.2	The TRAP overlay network	106
6.2.3	The TRAP protocol	106
6.2.4	Node identifiers	107
6.2.5	Trust holders	107
6.2.6	Calculation of reputation values	107
6.3	Security of TRAP	110
6.4	Evaluation of the reputation system	111
6.4.1	Method	112
6.4.2	Results	112
6.5	Related Work	114
6.6	Discussion	116
7	Crowd-based Detection of Routing Anomalies on the Internet	119
7.1	Introduction	119
7.2	Targeted routing attacks	121
7.3	System model and detection tradeoffs	122
7.3.1	System model and evaluation framework	123
7.3.2	Single node detection	124
7.3.3	Collaborative detection	126
7.3.4	Evaluation results	127
7.4	Detector selection	129
7.5	Candidate architectures and overhead	132
7.6	Related work	134
7.7	Conclusions	136

8	Does Scale, Size, and Locality Matter? Evaluation of Collaborative BGP Security Mechanisms	137
8.1	Introduction	137
8.2	Collaborative information sharing	139
8.2.1	Hijack prevention using prefix origin	139
8.2.2	Control-plane based anomaly detection	140
8.2.3	Route anomaly detection using passive and active measurements	140
8.3	Evaluating hijack prevention techniques	141
8.3.1	Simulation-based evaluation methodology	142
8.3.2	Global baseline: scale and size	143
8.3.3	Location-based discussion	145
8.4	Evaluating hijack detection mechanisms	146
8.4.1	Methodology and datasets	146
8.4.2	Global baseline	146
8.4.3	Location-based analysis	148
8.5	Interception and imposture detection	150
8.5.1	Methodology and datasets	150
8.5.2	Global and location-based evaluation	151
8.5.3	Scale of collaboration	152
8.6	Related work	153
8.7	Conclusions	155
9	Conclusions and Future Work	157
9.1	Thesis summary	157
9.2	Future work	161

List of Figures

1.1	Structure of the thesis	21
2.1	Routing on the Internet	25
2.2	Spreading of reachability information through BGP	29
2.3	BGP routing policies example, with three different types of paths available for the <i>source</i> to send traffic towards the <i>destination</i> : a provider path, a peer path, and a customer path, each indicated in the figure.	30
2.4	Route aggregation on the Internet	31
2.5	Example attacks targeting BGP, used to attract traffic intended for a victim network.	33
2.6	Example actions taken by the attacker networks and their impact on the victim node.	34
3.1	Interception observed in the traceroute from planet2.pittsburgh.intel-research.net to 125.246.217.1 (DACOM-PUBNETPLUS, KR).	57
3.2	Top 10 countries impacted by the China Telecom incident.	58
3.3	Example topology that allows for interception of traffic following decisions made by the neighbours of China Telecom (AS 4134)	60
3.4	Cumulative distribution function of prefixes each neighboring AS routed towards China Telecom.	61
4.1	High-level PrefiSec architecture.	70
4.2	Overview of the framework, its key components, and structure.	71
4.3	Holder assignment in prefix overlay.	74
4.4	Resolving query and longest-prefix query routing.	75
4.5	Number of Chord lookups and IP-level messages to resolve a query.	77
4.6	Cumulative Distribution Functoin (CDF) of the number of Chord lookups and IP-level messages to resolve a query.	78
5.1	Prefix hijack detection mechanism	86
5.2	Sub-prefix hijack detection mechanism	87
5.3	Cumulative Distribution Function (CDF) of the distance to the closest prefix in the global prefix registry, of newly observed prefixes.	89
5.4	Time-line of anomolous origin reporting.	90
5.5	Impact of alliance size.	90

5.6	Detecting route inconsistencies and potential interception attacks.	91
5.7	Interception detection mechanism	92
5.8	Different legit reasons for anomalies in traceroute AS path and routepath mismatching.	93
5.9	New routes observed during the first week of November, 2012.	95
5.10	ASes involved in the anomaly	98
6.1	Overview of components of TRAP.	105
6.2	Convergence and maximum difference of trust values for lower spam rates.	113
6.3	Convergence and maximum difference of trust values for higher spam rates.	114
6.4	Number of messages processed before the trust values converge for varying message drop rates.	115
7.1	Evaluation scenarios for example interception and imposture attacks.	123
7.2	Tradeoff between the alert rates of individual clients during attack and normal circumstances.	125
7.3	Alert rate during normal circumstances, during interception attacks, and during imposture attacks.	127
7.4	Alert rates as a function of the binomial threshold p_{bin}^* under different circumstances.	128
7.5	Tradeoff between the fraction of detected attacks during interception attacks and the false alert rate during normal circumstances.	129
7.6	Alert rates as function of the relative RTT distances between detector, attacker, and victim.	130
7.7	Detection tradeoff for different skew in the rates at which nodes are affected and detector nodes selected, respectively. Default parameters: $\alpha = 1$, $\beta = 1$, 50% affected nodes, and 40 detector nodes.	131
7.8	Attack detection rates under different attacks, while keeping a fixed alert rate of 0.01. Default parameters: $\alpha = 1$, $\beta = 1$, 50% affected nodes, and 40 detector nodes.	132
8.1	Average percentage improvement in the number of ASes that choose the correct origin when different subsets of the global set of ASes participate.	143
8.2	Impact of the number of participating ASes, when ASes are selected from a particular geographical region or the “rest of the world”.	144
8.3	Impact of the degree threshold of the participating ASes, when all are selected from a geographic region or the “rest of the world”.	145

8.4	Average number of alerts raised when global ASes collaborate the day of the China Telecom incident.	147
8.5	Average number of alerts raised when global ASes collaborate the day before (April 7) and after (April 9) the incident. . . .	148
8.6	Number of alerts during the day of the incident (April 8, 2010) for different sizes of regional collaborations.	149
8.7	Impact of the size of the participating ASes on the number of alerts.	149
8.8	Detection tradeoff, shown as the tradeoff between the detection rate during attack and false alert rate during normal circumstances, for varying percentages of affected nodes. . . .	151
8.9	Detection tradeoff, when keeping the number of detectors fixed. Here, 50% of the nodes are assumed to be affected. . .	152
8.10	Detection tradeoff, shown as the tradeoff between the detection rate during attack and false alert rate during normal circumstances, for varying numbers of detectors. Here, 50% of the nodes are assumed to be affected.	152

List of Tables

1.1	Classification of cybercrimes based on target	6
2.1	Simplified BGP decision process	29
3.1	Summary of control-plane updates matching the attack signature.	56
3.2	Organizations most impacted by the China Telecom incident.	59
3.3	Neighbors that routed the most prefixes to China Telecom. .	61
3.4	Organizations with the most potentially intercepted prefixes.	62
3.5	Why networks chose not to route to China Telecom.	63
5.1	Summary of route announcements statistics for a April 7, 2010.	88
5.2	Summary of route announcements statistics for a April 8, 2010.	88
5.3	Reduction in the number of traceroutes.	95
5.4	Analysis of traceroute paths and AS-PATHs during week two (Nov. 1-7, 2012).	97
5.5	Redundancy in unique triples using different approaches. . . .	98
5.6	Reduction in IP-to-AS mappings (Nov. 1-7, 2012).	100
6.1	Description of notations used for reputation model	108
7.1	Summary of datasets analyzed in chapter.	124
8.1	Examples of systems, the information they share/use, and the attacks they can help detect/prevent.	139

List of Acronyms

AS	Autonomous System
ASN	Autonomous System Number
BGP	Border Gateway Protocol
C2P	Customer-to-Provider
CA	Certificate Authority
DDoS	Distributed Denial of Service
DHT	Distributed Hash Table
DNS	Domain Name System
DNSSEC	Domain Name System Security Extensions
EGP	Exterior Gateway Protocol
FIB	Forward Information Base
FQDN	Fully Qualified Domain Name
IANA	Internet Assigned Number Authority
IDS	Intrusion Detection System
IGP	Interior Gateway Protocol
IGRP	Interior Gateway Routing Protocol
IP	Internet Protocol
IRR	Internet Routing Registry
IS-IS	Intermediate System to Intermediate System
ISP	Internet Service Provider
IXP	Internet eXchange Point
LIR	Local Internet Registry
MOAS	Multiple Origin AS
OSPF	Open Shortest Path First
P2C	Provider-to-Customer
P2P	Peer-to-Peer
PKI	Public Key Infrastructure
RIB	Routing Information Base
RIR	Regional Internet Registries
ROA	Route Origin Authorization
RPKI	Resource Public Key Infrastructure
RPSL	Routing Policy Specification Language
RTT	Round-Trip Time
S-BGP	Secure BGP
SoBGP	Secure Origin BGP
SRO	Secure Route Origin

Chapter 1

Introduction

The Internet has become an important part of human life and is currently providing critical services to billions of people each day. In 2015, approximately 43% of the world's population had Internet access [1], and as the number of people with Internet access increases, it will continue to redefine our social and personal lives. Today, the Internet is increasingly relied upon for end-user services such as eCommerce, entertainment, news, weather forecasting, email, voice over IP, video over IP, social interactions over social networks, business transactions, and information sharing [2], as well as controlling critical infrastructures such as power generation and transmission. Unfortunately, the success of the Internet, its popularity, and its wide usage have also attracted miscreants, and it is increasingly becoming a prime target for people and organizations performing illegal activities.

1.1 Cybercrime

With the increasing number of people dependent on the Internet for critical applications, it is perhaps not surprising that the Internet, the services it provides, and the services that use it in different ways are increasingly targeted by malicious users. Criminal activities that involve computers and networks are referred to as *cybercrime*. Cybercrime may have effects ranging in severity from simple annoyance to devastation for end users relying on the services provided over the Internet. Critical infrastructures such as energy installations, telecommunications, water supply, and transportation could be attacked and left non-functional. Online banking credentials of bank customers could be stolen and used illegally by attackers. The digital lives of users could be compromised or their computers could be used for other criminal activities leading to great financial, personal and legal problems for the user.

The cost of cybercrime is high. According to a 2012 study by Symantec,

the cost of cybercrime was estimated to be approximately \$110 billion [3]. Cybercrimes also carry very high indirect and defense costs for the average citizen. For example, Anderson et al. [4] estimate that the cost of defense against cyber fraud is at least ten times what cyber fraud nets for its perpetrators. In summary, it is in our interest that we design solutions that effectively mitigate the attacks in a way that reduces both direct costs incurred due to successful crimes and indirect costs associated with the mitigation techniques themselves.

1.1.1 Incidents of cybercrimes

At the time of writing, reports of major cybercrimes are commonplace.

- In June 2010, researchers discovered the Stuxnet worm, which seemed to target Iran's nuclear facilities. The worm attacked nuclear centrifuges causing them to self-destruct. The worm had a specialized malware payload that targeted only Siemens supervisory control and data acquisition systems (SCADA). The source of the worm was never discovered. Industrial systems have been targeted before, but the sophistication of the worm led many security researchers and firms to conclude that this worm had the support of a nation state.
- In April 2011, Sony officially announced that its PlayStation Network had been hacked. Personal data such as names, addresses, emails, and credit-card numbers of millions of customers were stolen in this attack. Sony had to shut down the PlayStation network owing to the attack. The outage lasted 24 days. Sony was the target of multiple lawsuits in Canada, the United States, and the United Kingdom following this incident. It was alleged that Sony did not store the user passwords in an encrypted format and failed to have adequate firewall protections, failed to provide adequate and timely warnings to its customers, and took a long time to restore their services. Under the UK's data protection act, Sony was fined £250,000 for failing to protect its customers' personal and financial information.
- In September 2012, the websites of the US-based banks JPMorgan Chase, Bank of America, Citigroup, U.S. Bank, PNC, and Wells Fargo suffered a day-long slowdown and blackouts for customers due to denial of service attacks¹. To carry out the attack, the attackers got hold of thousands of compromised computers and instructed them to target the banks. No clear explanation about the source of attack was revealed. Security researchers could only speculate that the attacks could have had support from a nation state or that attackers rented

---

¹N. Perloth, "Attacks on 6 Banks Frustrate Customers", <http://www.nytimes.com/2012/10/01/business/cyberattacks-on-6-american-banks-frustrate-customers.html>, Sep. 2012

botnets from the black market from the Internet underground. Banks face such attacks often and have some of the best defenses against these kinds of attacks. However, this time the banks were outdone.

- Users and organizations have long been targeted by phishing attacks, in which the attackers use spoofed email messages and attempt to acquire sensitive information such as username, password, and credit card details or to trick people into installing malware on their computers [5]. There are typically several confirmed phishing attacks reported every month.² For example, in 2011, it was reported that phishing attacks originating in China targeted the Gmail accounts of senior US government officials. In this incident, attackers sent crafted email messages that looked like subscription forms that could be activated using Gmail credentials. The message promised the users access to important reports. Once a user entered their credentials for Gmail, these credentials were sent to the attackers, who later used these credentials to monitor the Gmail accounts of the senior government officials.³ Other recent examples from 2016 include Netflix users being targeted for their credentials or students and faculty at various universities being affected by such attacks.^{4,5}
- In April 1997, AS 7007, a network owned by MAI Network Services, received a full route table from one of its customers. MAI did not filter the announcements that it received from its customer. MAI started separating routes into /24 routes (each specifying the originating network and path to reach these 256 IP addresses) and passing them on to its neighboring networks. AS 7007 made announcements as if these individual blocks of IP addresses (prefixes) were originating from itself, rather than the actual origin network. Other networks on the Internet started sending traffic towards the more specific /24 prefixes announced by AS 7007. Even though the router was rebooted within 15 minutes after the incident, the routes had already propagated throughout the Internet and it took many hours before the situation returned to normal. This event is often referred to as the first famous routing incident.⁶ Since then, many more targeted routing attacks have

---

²FraudWatch International, "Phishing alerts: Recently validated phishing attacks", [Phishingalerts:Recentlyvalidatedphishingattacks](#), Aug. 2016

³S. Yin, "Google Gmail Phishing Attacks Continue, China Suspected", <http://www.pcmag.com/article2/0,2817,2391058,00.asp>, Aug. 2011

⁴Symantec, "Netflix malware and phishing campaigns help build emerging black market", <http://www.symantec.com/connect/blogs/netflix-malware-and-phishing-campaigns-help-build-emerging-black-market>, Feb. 2016

⁵The Harvard Crimson, "Students, Staff, and Faculty Targeted by Phishing Scam", <http://www.thecrimson.com/article/2016/2/24/students-admins-phishing-email/>, Feb. 2016

⁶Avi Freedman, "7007: From the horse's mouth", <http://merit.edu/mail.archives/nanog/1997-04/msg00380.html>, Apr. 1997

occurred, including one studied later in this thesis.

- The new emerging *smart grids* are expected to rely on digital and analog communication systems to gather and act on information. Although smart grids are expected to create their own networks to enable distributed power generation, conferring advantages such as improved automated fault detection, increased efficiency, and commerce wherein consumers can also sell energy back to power distribution companies, these networks are also on the radar of cybercriminals and enemy states. Attacks on such electrical power grids can lead to power system blackouts, threats to human safety, and damaged consumer devices [6]. In a concrete example, in December 2015, the Ukrainian national power grid was subjected to a cyberattack which resulted in thousands of people being left without power for several hours. The sophistication of the attack is evident in the fact that after the power cut, a denial of service attack was mounted with the objective of preventing error messages from reaching service personnel [7].

With these examples, we can see how cyber-attacks can target important services and have effects ranging from annoyance to devastation for users on the Internet. Thus, it is important that such attacks are mitigated.

1.1.2 Reasons networks and users are vulnerable to cyberattacks

There are many reasons for the fact that cyberattacks are possible even after spending years of intellectual, computational, and financial resources to combat them. Some of the reasons are listed here:

- **Vulnerable software:** A vulnerability is defined as a weakness of an asset or group of assets that can be exploited by one or more threats [8]. Although many software vulnerabilities have been documented, both new and existing vulnerabilities continue to be identified. Examples of common software vulnerabilities are stack overflows and weak usernames and passwords for systems. These vulnerabilities can sometimes be exploited to get root privileges on the vulnerable systems, which can then be used to perform different attacks.
- **Packaged attacks:** Attackers do not necessarily need the advanced skills required to develop many of the common exploits observed today, as they are readily available on the Internet. Many times prepackaged tools exist that can be used by the attacker to aid in a cybercrime. In fact, there is a thriving digital arms trade that contributes to the ready availability of exploits, and criminals can purchase exploits and tools on illegal online forums or through brokers.⁷ Also, hackers sometimes

---

⁷“The digital arms trade”, <http://www.economist.com/news/business/>

post these tools freely on websites where they can be downloaded and used by attackers.

- **Legacy issues:** Vulnerabilities in network protocols also contribute to attacks on the Internet. Most of the components of the network protocol suite were developed when the Internet had only trusted hosts, and security was not a concern. The designers probably did not anticipate that the Internet would grow to the extent that it has grown today and that it would be a major target of illegal activities. A common example is a prefix hijack attack, where an attacker announces prefixes that are allocated to other networks, which the attacker is not permitted to announce. This attack is possible due to vulnerabilities in the Border Gateway Protocol (BGP), which is used to determine routing paths between different networks, and its inherent reliance on trust between network entities. When BGP was designed there were only a few network operators on the Internet, all of whom could trust each other, so security was not a major concern.
- **Configuration management:** Most of the existing attacks can be mitigated by system administrators who carefully follow security bulletins and apply patches released by vendors on their software as promptly as possible. However, it is not always possible to update all the affected host machines as soon as a new patch is available. Furthermore, sometimes system administrators may refrain from upgrading operating systems or software, as they worry that it could break applications that are critical to the organization.

1.2 Cybercrime classification and thesis focus

In Table 1.1, we broadly classify cybercrimes into four categories, depending on whether the target is an individual, the society as a whole, an end device, or the network infrastructure itself. Crimes that target individuals include identity theft, breaches of personal privacy, cyber stalking, and phishing attacks. Crimes in this category can result in personal financial loss and mental agony for the targeted individual, for example.

At the societal level, we include a wide range of crimes, including trafficking child pornography, copyright infringements, cyber laundering (money laundering using digital currencies), and counterfeiting. For example, it has been reported that cybercriminals launder money using in-game currencies in games such as *Second Life* and *World of Warcraft*.⁸ Media piracy in the form of downloading music and movie files illegally is rampant in many

---

21574478-market-software-helps-hackers-penetrate-computer-systems-digital-arms-trade, Mar. 2013

⁸O. Solon, “Cybercriminals launder money using in-game currencies”, <http://www.wired.co.uk/news/archive/2013-10/21/money-laundering-online>, Oct. 2013

Table 1.1: Classification of cybercrimes based on target

Target	Examples of attacks	Outcome of attacks
Individual	Identity theft, invasion of privacy, cyber stalking, phishing, cyber bullying	Personal financial loss, mental agony
Society	Child pornography, counterfeiting, cyber laundering, forgery, Internet fraud	Personal, corporate, and national financial losses and cost of countermeasures
End devices	Viruses, Trojan horse	Loss of data, stealing of data, annoyance, bot services such as spamming
Network	Routing attacks, botnets, denial of service attacks, scanning, spamming	Intellectual property loss, financial loss, cost of countermeasures, loss of trade and competitiveness, damage to company reputation

countries. Overall, the crimes in this category cause huge losses to governments, targeted individuals (*e.g.*, artists), and industries. It is often difficult to identify and prosecute responsible parties, as such crimes can be launched from anywhere in the world and are often deemed legal (or no laws against such activities are present) in the country where they originate.

The third category include the many crimes that affect user end devices such as computers, laptops, and smartphones. Such attacks often involve the attacker installing viruses or trojan horses on the end devices. Such attacks may lead to theft or loss of important data, and could also lead to end devices unwittingly being involved in bot activities such as spamming and scanning.

In the final category we place the crimes that disrupt the workings of the Internet itself, or that involve a large number of compromised hosts in a single network or spread over multiple networks to achieve their malicious intents. Disrupting the function of the Internet is a serious crime that can threaten national security and the financial system of a country. Attacks such as denial of service can render corporate networks useless. Attacks that use spamming and scanning (leveraging a large number of compromised hosts over the network) may be difficult to detect and mitigate. Attacks on routing protocols on the Internet can result in thousands of customers who are unable to reach their intended network destinations [9].

This thesis focuses on the subset of the attacks in the fourth category (shown in Table 1.1) of cyber-attacks, where the workings of the Internet and its services may be disrupted. We refer to this sub-category as *network-centric* attacks. More detailed examples of this category of attacks are presented in Section 1.3.

Referring back to Table 1.1 and the example attacks for each type of cyber threat, we note that the line between the four classes above can be

quite blurred as an attack from one type can be used to mount an attack of another type. For example, routing attacks can be used to steal traffic, which can be used to eavesdrop the credentials for user accounts if the credentials are not encrypted. Thus, routing attacks can help attackers to perform identity theft. Along similar lines, spamming can be used to make users reveal their credentials and/or perform Internet fraud.

1.3 Network-centric attacks

The Internet is often defined as a network of networks that enables an endless number of distributed systems to be implemented over a wide range of devices and network entities. In this landscape, different entities often play different roles and form services that collectively appear as a single system to the users.

Distributed systems built over the Internet are often designed to be scalable, heterogeneous, open, fault tolerant, secure, transparent to the user, and to enable communication between networks, devices, and users. However, these design properties are often contradictory to each other. For example, the property of systems being open easily compromises the property of such systems being secure, typically making such systems more vulnerable. In contrast, being open and heterogeneous enables growth and innovation in such systems since they are not managed by a single entity. These are important properties since, to provide useful services, different independent participants are often expected to cooperate in such systems. Unfortunately, it is often more difficult to incorporate accountability mechanisms, since a few participants acting maliciously can easily cause significant disruptions in such systems.

We use the term “network-centric attacks” to describe the subset of cyberattacks that target the workings of the Internet or its services with the objective of disrupting it. We further identify two distinct classes of network-centric attacks: (i) wide-area routing attacks targeting the exchange of information between different subnetworks, and (ii) attacks targeting or involving the edge networks closest to the end users. Typically, devices such as computers, smartphones, and laptops connect over an edge network, which relays the traffic of these devices to the rest of the Internet and allows them to communicate with devices in other subnetworks. To route a packet between end hosts located in different subnetworks, the Internet relies on wide-area routing protocols such as the Border Gateway Protocol (BGP). BGP helps spread reachability information between different Autonomous Systems (ASes), where an Autonomous System (AS) is a network under the administrative control of a single organization. This protocol is the target of many of the wide-area routing attacks, whereas attacks on the edge networks cover a much richer set of services. An example of a routing attack is a prefix hijack attack, in which the attacker announces itself as the origin of a prefix with the intention of attracting some of the traffic intended for IP addresses

belonging to this prefix. Examples of attacks involving the edge network are spamming and scanning.

Both of these types of attacks can significantly impact end users and organizations. First, consider the example of routing attacks that hijack traffic destined for a particular subnetwork on the Internet. These attacks often leverage the fact that networks have to collaboratively spread information about who owns which part of the total Internet address space using the Border Gateway Protocol to inject false routes into the network and, in this way, hijack traffic. This can result in significant disruptions for the end users and organizations wanting to access services associated with these hijacked prefixes (blocks of consecutive IP addresses). Second, with edge-based attacks such as spamming, clients and the edge networks that serve these clients can be significantly affected. Here also the relatively open design of the email protocol makes it relatively easy for a malicious attacker to launch new attacks. For example, any network can deploy an email server and start sending and receiving emails by connecting to other email servers at any time. Other examples of attacks that target the edge networks are scanning attacks, denial of service, and distributed denial of service.

Next, we give some examples of network-centric attacks and discuss the factors that enable such attacks on the Internet.

1.3.1 Incidents of network-centric attacks

- Over the years Renesys has reported many routing incidents. For example, already in 2016, it was reported that Con Edison, an Internet Service Provider (ISP) in New York, announced several prefixes owned by its customers and other networks.⁹ Networks such as NYFIX INC, Claren Road Asset Management, and Advanced Digital Internet, Inc. were affected by this incident. In 2013, Renesys reported that they had observed many interception attacks in which the traffic was re-routed through unwanted networks [10]. For example, during just the first part of 2013, traffic for more than 1,500 prefixes was reported to have been re-routed through potentially malicious networks in events lasting from minutes to several days and with hijackers appearing to be located in different countries. In general, the observed victims were diverse, with financial institutions, VoIP providers, and governments being the primary targets.
- The Spamhaus project maintains a list of real-time sources of spam that is used by various service providers and military and government organizations to block spam. In March 2013, the Netherlands-based internet provider CyberBunker was asked by Spamhaus to block spam and botnet communications originating from its network. However, CyberBunker refused to do so. In response, Spamhaus then asked the

---

⁹Todd Underwood, "Con-Ed Steals the Net" <http://www.renesys.com/2006/01/coned-steals-the-net/>, Jan. 2006

provider of CyberBunker, DataHouse, and its service provider, A2B Internet, to block CyberBunker's traffic. However, both providers refused to do so. In response, Spamhaus then started listing both of these providers in its list of spam sources. In retaliation, CyberBunker launched a distributed denial of service attack against Spamhaus. Later, CyberBunker also found itself at the receiving end of denial of service attacks.¹⁰ These attacks were among the largest of their kind on the Internet and resulted in significant collateral damage in terms of slowing down the Internet for millions of users world-wide.

- In February 2008, Pakistan Telecom started announcing to its provider, PCCW, a route for prefix 208.65.153.0/24, which is a subprefix of the prefix 208.65.152.0/22, which is assigned to YouTube [11].¹¹ Pakistan Telecom's objective was to block access to YouTube in Pakistan. However, this false announcement was leaked to the Internet and traffic intended for the subprefix, and thereby traffic intended for YouTube, was rerouted towards Pakistan Telecom. This resulted in users world-wide being blocked from accessing YouTube for several hours.
- Pilosov and Kapela [12] presented a live demo of an interception attack at the DEFCON conference in August 2008. They demonstrated the attack by intercepting the traffic from the conference network. The researchers diverted the traffic intended for DEFCON through servers under their control and rerouted the traffic back to the conference network. During the interception attack, routers were fooled into re-directing traffic to the attacker's network, where the presenters simultaneously used route path prepending to cause critical networks to reject their fake advertisements, allowing them to redirect the traffic to the victim network (original destination for the traffic).
- Internet attacks are also used to target rival organizations or enemy states. In August 2008, the country of Georgia was a victim of serious denial of service attacks. The attacks disabled important government websites. It was alleged that Russian intelligence agencies conducted these attacks just before Russia was to launch military action against Georgia. This incident illustrates the potential of cyber warfare to augment traditional military attacks.¹²

---

¹⁰M. J. Schwartz, "Spamhaus DDoS Suspect Arrested", <http://www.informationweek.com/security/attacks/spamhaus-ddos-suspect-arrested/> 240153788, Apr. 2013

¹¹M. Brown, "Renesys Blog: Pakistan Hijacks YouTube", <http://research.dyn.com/2008/02/pakistan-hijacks-youtube-1/> Feb. 2010

¹²J. Leyden, "Russian spy agencies linked to Georgian cyber-attacks", http://www.theregister.co.uk/2009/03/23/georgia_russia_cyberwar_analysis/, Mar. 2009

1.3.2 Factors contributing to network-centric attacks

In addition to the previously discussed general factors that contribute to cybercrime in Section 1.1.2, we next discuss factors that contribute to network-centric attacks in particular.

- **Growth and complexity of the Internet:** In the early days of the Internet, many network protocols were designed without security in mind. The number of networks and users connected to the Internet was small and the Internet was not considered a critical infrastructure. With only a few networks in operation, the network operators typically knew each other. Such personal trust was sufficient for network operators and protocol designers to expect that networks would behave as per specifications and without malicious intentions.

Since then, the Internet has grown in both user population and number of networks. The number of networks operating autonomously on the Internet has grown from 3,000 in 1998, to more than 45,000 in 2014.¹³ With such a large number of networks and different operators, the implicit trust that existed in the beginning when various protocols such as BGP were deployed, based on inter-personal relations among network operators, no longer holds.

Furthermore, new applications and services on the Internet have demanded the development of new protocols, and modifications to older protocols. Such dynamics have added more complexity to the working of the Internet and have given rise to unforeseen security vulnerabilities and misconfiguration in various protocols.

- **Sophisticated techniques by attackers:** Recently, attackers have developed attack techniques that are difficult to detect for a single network acting alone. For example, consider the sophistication that is involved in some current spamming campaigns. First, attackers often scan different networks for vulnerable hosts. These machines are then typically compromised and enrolled in larger networks referred to as botnets. There can be thousands or even millions of compromised hosts, referred to as bots, in a botnet. Together, these bots offer huge resources to the attackers. Using botnets, the attacker can then send a large number of spam mails without the individual bots being detected. Specifically, since each bot might only send a few spam mails, the individual spamming bots are typically not detected or blacklisted. For example, Ramchandran *et al.* [13] found that more than 65% of hosts (as identified by their IP addresses) known to be infected with the *Bobax* bot sent spam only once to their sinkhole.
- **Asymmetric nature of network-centric attacks:** In contrast to physical warfare, where years of research and millions of dollars have

¹³Tony Bates, Philip Smith, Geoff Huston, “CIDR report”, <http://www.cidr-report.org/as2.0/>, Nov. 2014

typically been needed to develop military weapons and train military personnel, large-scale network-based attacks are often inexpensive and can be executed by a single individual or small group of individuals. If these hackers have the skills and tenacity they can create havoc and damage by performing network-centric attacks.

Even scarier is the fact that a small group of individuals might not need significant computing resources of their own. As in the spam campaign case, these attackers often build strength by compromising thousands or, in some cases, millions of computers. Today, there is even a market where hackers offer compromised machines for rent. Attackers have the advantage that they only need to find one vulnerability to achieve their objective, while defending networks become vulnerable as soon as they make a single mistake.

- **Passive role played by network operators:** Often network operators are in the best position to police the miscreant activity within their own networks [14]. By monitoring their own networks, they can help ensure that compromised machines within their own networks do not cause prolonged harm to others, or they can use stronger filtering and security mechanisms to avoid deviation from protocols that could allow attacks that cause intentional harm to other networks.

For example, network operators could perform fine-grained filtering of routes advertised by their customers. If all network operators filtered customer routes accurately, the global routing system would be much more secure [11]. However, maintaining filter lists is challenging when the customer base is large, and thus such simple filtering techniques are not employed by network operators. In general, either intentionally or due to lack of resources, most network operators do not currently police their networks.

To make things worse, sometimes network operators have an economic incentive to turn a blind eye to the activities of their customers. For example, as discussed earlier, some large network operators do nothing to block spammers in their networks or worse, sell service knowingly to professional spammers for profit.

1.4 Problem formulation

This thesis focuses on network-centric attacks. More specifically, we focus on routing attacks and attacks involving edge networks through malicious activities such as spamming and scanning. In the following, we formulate the primary research questions addressed in this thesis.

Understand large-scale routing anomaly: With an increasing amount of sensitive information routed across the globe each day over the Internet, the importance of securing the route paths increases. While it is well known

that routing over the Internet is vulnerable to attacks and misconfigurations, the mechanisms and dynamics of real-world attacks are less well understood. One such large-scale routing anomaly occurred on April 8, 2010, when an Autonomous System (AS) owned by China Telecom announced approximately 50,000 prefixes (blocks of consecutive IP addresses) registered to other networks. Although this attack received widespread publicity, a systematic study of exactly what happened during the attack was not available in the public domain. To better understand what unfolded during this incident, part of this thesis provides the first large-scale characterization ever done of this incident. Characterization is intended to answer what allowed large amounts of traffic intended for other networks to be routed through China Telecom’s network as well as to assess the potential damage caused by this incident.

Collaboration among network entities to detect attacks: Besides routing attacks (prefix hijack, subprefix hijack, path hijack) [15], network operators have to confront malicious activities such as spamming and scanning that may emanate from their network. Unfortunately, miscreants are becoming increasingly sophisticated and security attacks are no longer isolated events. Instead, attacks often cover multiple domains and behaviors [13, 16, 17]. For example, Ramchandran *et al.* [13] found that routing anomalies and botnets are exploited by spammers to avoid detection while sending spam, making it difficult for a single network acting on its own to detect the attack. The examples they identify in their work go on to show how attackers combine different attacks and stealthy techniques to mount attacks. Similarly, Ying *et al.* [18] show that networks that are worm-infected, origins for spam, or the source of other edge-network-based anomalies also exhibit more anomalous control-plane routing behavior. These examples clearly show that there is a need for mechanisms that target attacks against routing, edge networks, or both.

Given the sophistication and distributed nature of many attacks, it has been suggested that collaboration among network domains and sharing of information can help in protecting against such attacks. Collaboration among network entities provides richer information, and can help detect and prevent such attacks [13, 16]. While collaboration between network entities has been proposed, and the value of such collaboration has been demonstrated, it remains an open problem to design distributed mechanisms that provide effective decentralized information sharing among different network entities.

In the past, various collaborative approaches have been proposed to detect routing attacks, spamming, and scanning. However, in general existing techniques require all information to be gathered and processed at a central location. Such centralization entails extra overhead in data transfer, and results in large processing overhead at a single location. Additionally, centralization raises the issue of a central point of failure for the system. Given the need to potentially process huge amounts of data, it is important that any collaborative effort to target distributed attacks should be scalable.

With regards to scalability, the system design must minimize the overhead involved in sharing information, while keeping the processing distributed.

To address the above challenges, as a part of this thesis we aim to design highly distributed, scalable, and effective collaborative mechanisms that target routing and edge-based-network attacks. Within the contexts of collaborating networks, collaborating servers, and collaborating end-users we design efficient, scalable solutions that aim to minimize communication and processing overhead when the processing is distributed. Each of these contexts results in unique design challenges.

Effect of scale, size, and locality: Many routing security mechanisms have been proposed to mitigate various attacks on inter-domain routing. At a higher level these proposals can be classified as hijack prevention [19–21] or hijack detection [22–24] mechanisms. Whereas past works have demonstrated the benefits of these proposals and considered deployment aspects [25–27], there are still many open questions related to the benefits of limited or regional deployment of the proposed mechanisms. For example, while past works suggest that carefully selecting ASes that deploy these mechanisms (mostly based on the size of the AS) may give tangible protection from the routing attacks, the locality aspects of the ASes is often not considered in these evaluations. This could be an important factor since large ASes spread across the globe may not agree to use a single solution due to political and financial considerations. This may also be a contributing factor to the lack of global deployment of any of these solutions.

Furthermore, given the lack of deployment of proposed mechanisms, regional government-issued legislation or regional agreements may be a future means to ensure that at least ASes within specific regions agree to use common routing security mechanisms. For example, the United States government or the European Union may push to have ASes and organizations under their respective jurisdictions share information to protect the common interests of the region. Similarly, mechanisms that are based on users’ participation may see more acceptance within the same region due to some users feeling a higher level of trust in collaborating with other users from within the same region.

Given the above observations, we aim to answer questions related to how the security gains of different previously proposed mechanisms are affected when they are deployed only in a specific geographic region, and how this compares with global scenarios. We also study the effect of the number and size of the collaborating ASes, answering questions related to the impact that the makeup of the collaborations have on their success in both detecting and preventing routing attacks.

1.5 Contributions

1.5.1 Study of large-scale routing anomaly

Chapter 3 in this thesis presents our case-based study of the large scale routing anomaly that occurred on April 8, 2010, in which an Autonomous System (AS) owned by China Telecom announced approximately 50,000 prefixes registered to other networks [9]. We label this incident the *China Telecom incident*. We use the China Telecom incident to understand (1) what can be learned about large-scale routing anomalies using public datasets, and (2) what types of data should be collected to diagnose routing anomalies in the future. We develop a methodology for inferring which prefixes may be impacted by traffic interception using only routing related updates and validate our technique using traces of paths that were taken by the data traffic. Our study clearly highlights that the decisions made in terms of routing by other ASes resulted in traffic being routed through the China Telecom network. Our analysis and results also highlight and support the need for collaboration among ASes to mitigate different attacks on the Internet. This chapter addresses the first set of questions outlined in Section 1.4.

1.5.2 Collaboration among network entities to detect attacks

The second set of problems outlined in Section 1.4 have been addressed in three different contexts. First, we propose a novel prefix-based collaborative framework, PrefiSec, which enables collaboration and information sharing among ASes. Second, we present a passive-measurement based approach, CrowdSec, to detect and raise alerts for possible routing anomalies. CrowdSec enables collaboration among users in contrast to collaboration among ASes as in PrefiSec. Such an approach is motivated by the need to detect stealthy routing attacks where attackers impersonate the victim or reroute the traffic towards the victim network. Such attacks can be detected only when the actual route taken by the affected traffic is taken into consideration. This system enables concerned citizens to collaboratively detect and report potential traffic hijacks to operators and other organizations that can help enforce route security. Finally, we also propose a specialized Distributed Hash Table (DHT) based solution, TRAP, which enables host-based evaluation of spam servers. Next, we highlight some of the contributions made in these three system designs.

Collaboration among ASes (Chapter 4): PrefiSec includes a distributed reporting system that allows participating members to effectively share observations, report suspicious activities, and retrieve information that others have reported. Our solution includes a distributed IP-prefix-based DHT extension of Chord. PrefiSec takes into account the hierarchical nature of the IP space, implements functionalities such as longest-prefix matching,

and allows us to effectively store and retrieve information about organizations and their IP prefixes, as well as evaluate them with regards to a range of different attacks and miscreant behavior. The key elements of our solution are: it is scalable, it does not require ASes to divulge strategic information, and it allows sharing of information about many different types of attacks.

Security using PrefiSec (Chapter 5): We show how PrefiSec can be used to build policies that help protect against various inter-domain routing related attacks by detecting anomalous behavior. While we focus on routing attacks such as prefix hijacks to show how PrefiSec can help, we also discuss how the framework can be used to report edge-network activities such as spamming, scanning, and botnet servers. Public wide-area BGP-announcements, traceroutes, and simulations are used to estimate the overhead, scalability, and alert rates of the system. Particular attention is given to the information that organizations should exchange with each other to detect different types of attacks and how this information can be effectively distributed among the participants in the alliance.

Collaboration among mail servers (Chapter 6): We present a specialized DHT-based solution, TRAP, which enables host-based evaluation of SPAM servers [28]. In contrast to the prefix-based evaluation of PrefiSec, TRAP keeps track of individual mail servers. This system can be seen as a specialized incubator for PrefiSec in that once the malicious sources are detected, these hosts can be mapped to appropriate prefixes using PrefiSec, and the malicious behavior of hosts can be attributed to the networks to which these hosts belong.

Collaboration among users (Chapter 7): We also present CrowdSec, our user-centric passive-measurement based approach to detecting and raising alerts for possible routing anomalies. In our evaluation of CrowdSec, we use longitudinal RTT measurements from a wide range of locations to evaluate the anomaly detection tradeoffs associated with our proposed mechanism. Considering two types of stealthy and hard-to-detect attacks (interception and imposture attacks) [29], we provide results for tradeoffs between attack detection rates and false alert rates. We present effects of system scale, participation, and relative distances between attackers, detectors, and victims. We also evaluate the set of detector nodes that provides the best detection rates for candidate victim nodes using a simple system model. Finally, we present a discussion and analysis of the overhead associated with different candidate architectures for CrowdSec, which include both central directory and fully distributed approaches.

1.5.3 Effect of scale, size, and locality

Chapter 8 addresses the problem of impact of the scale, size, and locality on hijack detection/prevention mechanisms. Here, we present a systematic evaluation of some promising and previously proposed hijack prevention and routing attack detection techniques, paying particular attention to the local-

ity aspects of their deployment [30]. In particular, we consider three example techniques that share (i) prefix origin information, (ii) route path updates, or (iii) passively collected round-trip time (RTT) information, and evaluate the impact of the number of participants, their size, and their locality on routing attack detection/prevention mechanisms. For our evaluation, we develop a data-driven methodology for each information sharing approach, which takes into account the geographical locality and the size of each of the potential participants. Using real-world topologies and routing information derived from measurement data, we then systematically evaluate the impact of each factor, either on its own, or accounting for the geographic locality of the participants, attackers, and victims.

1.6 Methodology

The work in this thesis uses a combination of different methodologies, including measurements, quantification, and simulation to evaluate different systems aspects discussed in the previous section. For example, measurements and data analysis are used to characterize real-world events, and simulations are used to evaluate newly proposed methods and system designs. Our measurement-based analysis and evaluation uses a combination of active and passive measurements to test hypotheses and combine alternate design choices. Throughout the thesis we identify and discuss the necessary assumptions and their limitations. While our datasets and evaluation have limitations, we do not expect them to lead to faulty conclusions.

1.6.1 Characterization and empirical observations

We perform a characterization study of the much debated routing incident that occurred on April 8, 2010, referred to as the China Telecom incident. A major challenge in any characterization study is getting access to the data around the phenomenon that is being characterized. For the China telecom incident we used data from RouteView [31] as a source of BGP updates around the time of the incident and data from iPlane [32] to extract the traceroutes around the time of the incident. These datasets have been used in numerous research works, providing insights into their usefulness in research. Questions of the form “how many”, “what is”, and “which” were formulated and answered with the help of this data, with the answers giving insights into the size and characteristics of the attacked networks as well as the underlying mechanisms that allowed the China Telecom incident to take place.

1.6.2 Evaluation of large-scale systems

PrefiSec and CrowdSec: We present PrefiSec and CrowdSec, systems that enable collaboration and sharing of information among ASes and end

users, respectively, to help raise alerts for various routing attacks. The main challenge in this part was to evaluate the proposed system. We based our evaluations of PrefiSec and CrowdSec on the state-of-the-art approach for performance evaluation described by Jain [33]. PrefiSec is based on sharing of information among large ASes, and it would have been difficult to bring together a large number of AS operators for live testing of such a system. Similarly, CrowdSec is based on sharing of information among end users spread across a large geographical region and, as with PrefiSec, it would have been difficult to evaluate proposed mechanisms without large-scale user participation. Instead, we use a combination of simulations, active and passive measurements, and data-driven analysis to evaluate PrefiSec and CrowdSec, and to evaluate different policies designed to detect different attacks.

For simulations, we use real world public data as a workload. This has several advantages. First, it gives us results that are as close as possible to real world scenarios, and second, since the data is public, others can repeat the evaluation using the same datasets. The major disadvantage with this approach is that we are not able to capture all aspects of a fully implemented system running on the Internet. Therefore, we acknowledge that it is possible that we may miss some idiosyncrasies and effects that would occur if implemented as a live system on the Internet.

Furthermore, for simulations, we leverage existing and well-tested open source tools and software as much as possible. For example, to simulate the structured Distributed Hash Table based peer-to-peer network called Chord [34], we use an open source tool called PlanetSim [35]. Leveraging existing tools helped us focus on research and avoid getting sidetracked in implementation issues.

We also performed active measurements in our design and evaluation of the proposed method to detect interception attacks, which is a more difficult type of routing attack to detect. Our active measurements give us a good approximation of what to expect in real world scenarios and help us evaluate what a real system would observe. Performing active measurements is time consuming and comes with many pitfalls [36]. The main challenge in performing active measurements was making sure that the system was up and running all the time. This task was especially challenging as we had to use services from public servers which were not under our administrative control. For example, our measurement methodology had to account for the fact that requests sent to such remote servers might be blocked if one requests the service quite often in a short time span. In general, however, we tried to leverage other active measurement platforms such as iPlane. For example, our evaluation of CrowdSec relies almost entirely on data collected using iPlane [32]. In addition to saving time, it also enhances our confidence in our evaluations, since these measurement platforms are well tested and extensively used by other researchers.

To present our performance evaluation results we use techniques such

as Cumulative Distributed Frequency (CDF) plots, box plots, and heatmap plots. Such techniques give a better view of the variability of the variable being measured than basic first order statistics such as averages. When required, we also use absolute numbers and percentages to comment upon the observed overhead, alert rates, and scalability of PrefiSec and CrowdSec.

TRAP: As with PrefiSec, we identify the most important aspects of TRAP and perform a simulation-based evaluation focused on these aspects. For the simulations we used a structured Distributed Hash based peer-to-peer network called Pastry and an open source implementation called FreePastry [37]. The simulation workload used for our TRAP evaluation was generated based on characteristics studies performed by other researchers.

Effect of scale, size, and locality on routing security mechanisms: Focusing on previously proposed routing attack prevention/detection mechanisms we used similar evaluation approaches as had been used for evaluation of these mechanisms in the past, but extended the evaluation methodologies to account for scale, size, and locality of the networks that participate in using those solutions. Our evaluation of hijack prevention mechanisms is based on simulations for which we used and modified open source simulation tool called BSIM [23], whereas the evaluation of hijack detection mechanisms is based on PrefiSec and CrowdSec.

1.6.3 Working with large datasets

For much of this thesis we work with large datasets. Our characterization of the China Telecom incident includes analyses of millions of route announcements and traceroutes. Similarly, our evaluation of PrefiSec, CrowdSec, and different security mechanisms based on scale, size, and locality perspectives includes using millions of BGP route announcements from several BGP servers and thousands of active traceroute measurements from several geographically distributed servers. When working with these large datasets we have followed some of the strategies identified by Paxson [36]. In the following, we describe some of the problems we encountered and how they were resolved.

Large datasets: Performing analyses on large datasets can lead to issues because of system limitations such as disk space and processing speed. Thus, instead of focusing on extracting the information that one needs, one can get sidetracked in dealing with issues that arise due to large datasets.

To resolve this, we often first worked on subsets of data, allowing us to quickly create and study different statistics and graphs, before creating statistics based on the full datasets. As the characterization work often involves working with many graphs, which can be time consuming to generate with large datasets, it often helps to break down the task into sub-steps, such that a smaller subset of data can be used before proceeding to the final steps. This helps identify and formulate initial hypotheses that can be tested on the larger dataset. If one chooses to perform all such tests with complete

datasets from the beginning, often much of the effort that has been spent will go to waste. When specific tests and graphs were deemed appropriate, only then did we choose to perform the analyses on the complete datasets.

Repeatability of analysis: Starting with meta data, we often had to perform several steps of data processing to arrive at different results and graphs. For example, for interception detection policy evaluation, we first had to map IP addresses to their originating networks. Next, we created route-paths based on these traceroutes (appended each IP address in the traceroute with the originating network). Several additional steps were used before final results were derived.

The use of multi-step processing can greatly simplify some tasks, but it can also complicate traceability unless proper bookkeeping is employed. Therefore, it is important to keep track of the various steps taken to arrive at results starting from the meta data. To keep track of such steps we created separate documents for each type of analysis that was performed, in which we used multi-step scripting and wrote down the steps that were taken for that analysis. For each step we kept a record of the script files, input data used, and the output files created. For the script files we also displayed a message before the script was run explaining in detail what the script file was expected to do, listing expected input and output files, and providing a short description of their contents.

Our systematic documentation allowed us to go back to the analysis (even after several months) and quickly get an overview of the various steps taken to arrive at the results. The documentation and scripts also helped us ensure that the same steps were always taken when repeating the analysis on new or larger datasets. Finally, using multi-step scripts allowed us to perform repeated steps of analysis for different evaluations quickly, without introducing errors.

Reproducibility and comparison: Reproducibility is a very important aspect of research, which is becoming increasingly challenging as data analysis is getting more complex and often involves large datasets (which may be more difficult to share). The analyses done in this thesis, be it for the characterization of a large-scale routing anomaly or for the evaluation of proposed mechanisms to detect routing attacks, involves large datasets. Most of our analysis is done using public datasets. This should allow other researchers to evaluate alternate system designs on the same data, to validate and compare with our results, or both. For the characterization of the large-scale routing anomaly that we performed, we have also made the data publicly available so that other researchers can reuse and build upon our dataset.

Ensuring correctness: After each step of the analysis we tried to ensure that the new results obtained were consistent with what was expected and any prior results we may have had. We continuously checked if the number of lines in the file before and after doing some processing was the same. Similarly, we often checked that the number of fields (comma, space,

or bar separated) in each line was the same after some simple processing was done, so as to ensure that the expected formatting remained consistent in each step of processing. Inconsistencies from the expected output often helped us detect errors in that particular step and helped us rectify them right away. Also, more domain specific sanity checks were applied to ensure integrity of the data. This could include leveraging relationships between the number of traceroutes performed and the number of observed AS-paths, for example.

1.7 List of publications

Parts of this thesis are presented in following publications listed in the descending order of their publication (submission) dates:

- [Rahul Hiran](#), Niklas Carlsson, and Nahid Shahmehri, **Collaborative Framework for Protection Against Attacks Targeting BGP and Edge Networks**, submitted to journal for possible publication, Jul. 2016.
- [Rahul Hiran](#), Niklas Carlsson, and Nahid Shahmehri, **Does Scale, Size, and Locality Matter? Evaluation of Collaborative BGP Security Mechanisms**, Proc. IFIP Networking, Vienna, Austria, May. 2016.
- [Rahul Hiran](#), Niklas Carlsson, and Nahid Shahmehri, **Crowd-based Detection of Routing Anomalies on the Internet**, Proc. IEEE Conference on Communications and Network Security (CNS), Florence, Italy, Sep. 2015.
- [Rahul Hiran](#), Niklas Carlsson, and Nahid Shahmehri, **PrefiSec: A Distributed Alliance Framework for Collaborative BGP Monitoring and Prefix-based Security**, Proc. ACM CCS Workshop on Information Sharing and Collaborative Security (ACM WISCS @CCS), Scottsdale, AZ, Nov. 2014.
- [Rahul Hiran](#), Niklas Carlsson, and Phillipa Gill, **Characterizing Large-scale Routing Anomalies: A Case Study of the China Telecom Incident**, Proc. Passive and Active Measurement Conference (PAM), Hong Kong, China, Mar. 2013.
- Nahid Shahmehri, David Byers, and [Rahul Hiran](#), **TRAP: Open decentralized distributed spam filtering**, Proc. International Conference on Trust, Privacy and Security in Digital Business (TrustBus), Toulouse, France, Aug./Sept. 2011.

Figure 1.1: Structure of the thesis

1.8 Thesis organization

Figure 1.1 shows the relationship between the problems considered (annotation at the bottom), different attacks targeted (annotation on the right), and the different chapters in the thesis. For example, TRAP (Chapter 6) and parts of PrefiSec (part of Chapter 5) concern attacks targeting the edge networks. The remaining chapters and contributions focus on detecting and mitigating the routing attacks.

Chapter 2 presents background information and related work. Our case study of the China Telecom incident and large-scale routing anomalies is presented in Chapter 3. The PrefiSec framework for collaboration among defending ASes and its evaluation is presented in Chapter 4. Chapter 5 shows how PrefiSec can be used to detect routing related attacks and evaluate the proposed methods. We also discuss how PrefiSec can be used to target edge-network based malicious activities such as spamming and scanning. Chapter 6 presents TRAP and a basic evaluation thereof. Chapter 7 presents our user-centric crowd-based approach to detect stealthy attacks such as interception and imposture attacks. Evaluation of routing security mechanisms with regard to scale, size, and locality is presented in Chapter 8. Chapter 9 concludes the thesis with an overview and discussion of future work.

Chapter 2

Background and Related Work

This chapter provides background on key concepts used in this thesis and discusses related works. Section 2.1 provides an overview of the basics of Internet routing, including an introduction to intra-domain routing. Section 2.2 gives an overview of the Border Gateway Protocol (BGP), which is the de facto protocol used for inter-domain routing. We follow that with a discussion of the types of attacks that are possible due to vulnerabilities in BGP in Section 2.3 and different techniques that have been proposed to mitigate threats to BGP in Section 2.4. Broadly, this research has focused on BGP attack prevention techniques (crypto-based) and detection techniques (anomaly-based). Section 2.5 presents a brief introduction to attacks involving edge networks, including spamming and scanning. Section 2.6 presents an introduction to peer-to-peer (P2P) networks. Section 2.7 gives a high level overview of intrusion detection systems. We conclude the chapter with a discussion of how the related work presented in this background section connects to other parts of the thesis.

2.1 Internet routing

With millions of hosts on the Internet spread throughout the world, the best possible routes should be taken to exchange traffic if any two nodes want to communicate with each other. This is where Internet routing plays an important role. Routing over the Internet is based on information about which networks or which Autonomous Systems (AS) control which part of the Internet Protocol (IP) address space. We will first give a brief introduction to ASes and then discuss the different routing protocols used on the Internet.

2.1.1 Autonomous systems and addressing

The network layer provides a host-to-host communication service. At the core of this service is the network layer's routing function, which is used to determine good routes between two communicating hosts. There are various classes of routing algorithms proposed and used in practice, including link-state, distance-vector, and path-vector.

For scalability reasons and administrative autonomy, the routers are organized into autonomous system. An Autonomous System (AS) is a group of networks operated by one or more network operators(s) that has a single and well defined external routing policy [38]. An AS is typically responsible for one or more IP prefixes (blocks of consecutive IP addresses). Routers within the same AS all run the same routing algorithm. An AS has a globally unique number associated with it, referred to as its Autonomous System Number (ASN). The ASN is used in the exchange of exterior routing information, and as an identifier of the AS itself.

Every host (any computing device such as mobile, laptop, tablet) on the Internet has an Internet Protocol (IP) address; either a 32-bit IP version 4 address or a 64-bit IP version 6 address. In the current number resource allocation structure, the Internet Assigned Number Authority (IANA) is the nodal authority that allocates IP prefixes to 5 Regional Internet Registries (RIR), which in turn allocate prefixes to Internet Service Providers (ISPs) and Local Internet Registries (LIR). Similar sub-allocations of addresses are used to allocate addresses to networks, subnetworks, and so forth down to the host level. A similar hierarchy is also observed in allocation of the ASNs.

2.1.2 Routing on the Internet

The Internet can be thought of as an interconnected network consisting of thousands of networks and ASes that allow exchange of information/data between any hosts that are connected to these networks. There are tens of thousands of ASes and millions of hosts on the Internet. The traffic that is intended for the hosts that are assigned IP addresses from within the range of prefixes allocated to an AS should be directed towards that AS. This is where Internet routing plays an important role.

Routing protocols help spread reachability and path information for prefixes, and specify how routers communicate with each other, as well as how they should share information that enables them to select appropriate routes between any two nodes in the computer network. Routing protocols help the routers to gain knowledge of the topology of the Internet.

Broadly, any routing protocol can be classified either as an Interior Gateway Protocol (IGP) or an Exterior Gateway Protocol (EGP). IGPs are used for exchanging routing information between routers within an AS. In contrast, EGPs are used to exchange routing information between ASes.

ASes use an IGP and some common metric, such as the shortest routes, to route packets within the AS, and an EGP to route packets to other ASes.

Figure 2.1: Routing on the Internet

Each AS might choose a different IGP, or one AS might even choose multiple IGPs. However, this is not the case with the choice of EGP, as it is necessary that an EGP is the same for all neighboring ASes. The Border Gateway Protocol (BGP) is the de facto protocol that is used globally as an EGP. BGP is a path vector protocol that gives the ability to implement policies in routing. A routing policy allows certain routes to be preferred over others. For example, a company may not want traffic routed through a particular AS due to local laws in a country where the company operates.

Consider an example setup of ASes and routers as shown in Figure 2.1. Suppose router *Y2* in AS *Y* wants to send traffic to router *X1* in AS *X*. To achieve this, *Y2* needs to forward the traffic to gateway router *Y1* using the shortest path. The least cost path in such cases is determined using an intra-AS routing algorithm or Interior Gateway Protocol (IGP).

Now, consider another scenario where router *Y3* needs to send traffic to AS *W*. AS *Y* has no direct connection to AS *W* and therefore needs to send the data through AS *Z*. To learn such reachability information and propagate it, an inter-AS routing algorithm, also called an Exterior Gateway Protocol (EGP), is used. Since inter-AS routing involves communication between different ASes, the same routing algorithm must be used by all the ASes on the Internet. The de facto inter-domain routing protocol that is currently in use is the Border Gateway Protocol (BGP).

2.1.3 Control-plane and data-plane

A *router* is a device that forwards data packets between computer networks. It is common to separate the forwarding of the actual data traffic from the communication associated with the control information used to precompute good forwarding paths [39].

- **Data plane:** In this plane, the router is concerned about per-packet processing and forwarding of data packets. When the packet arrives

at a router on the incoming link, the router forwards the packet to the appropriate output link. The router uses data from forwarding tables, often referred to as a Forward Information Base (FIB), which are constructed using control-plane information, to make forwarding decisions for the data traffic.

- **Control plane:** In this plane, routers exchange reachability information with other routers to determine the route taken by packets from sender to receiver. The control plane constructs the Routing Information Base (RIB) using both static pre-configured routes and dynamic routes, which are learned using dynamic protocols such as BGP. The RIB is used to build the FIB that will be used for forwarding in the data plane.

Systems designed to detect and mitigate security problems by analyzing packet headers and payloads are typically referred to as data-plane based systems. Similarly, systems based on analysis of data in signaling and control protocols are often referred to as control-plane based systems.

2.1.4 Intra-domain routing protocol

Before discussing inter-domain routing protocols such as BGP, we briefly describe some popular intra-domain routing protocols, or Interior Gateway Protocols (IGPs). As discussed earlier, intra-domain routing protocols are concerned with spreading reachability information within a network under the control of a single administrative domain. Different ASes can use different intra-domain routing protocols.

There are several classes of intra-domain routing algorithm, such as link-state algorithms and distance vector routing algorithms. Link-state algorithms use global information, where the network topology and all link costs are known by all the nodes. In contrast, distance-vector algorithms are distributed algorithms in the sense that each node receives some information from directly connected neighbors and uses this information to calculate and update its distance information and routes. Finally the nodes may redistribute the results from these calculations to their directly attached neighbors. Examples of link-state routing protocols are Open Shortest Path First (OSPF) and Intermediate System to Intermediate System (IS-IS). Routing Information Protocol (RIP) and Interior Gateway Routing Protocol (IGRP) are examples of distance vector routing protocols. Below, we briefly describe two of these protocols.

Open Shortest Path First Open Shortest Path First (OSPF) is a link-state routing protocol that floods the link-state information and uses Dijkstra's least-cost path algorithm. Each router constructs a topological map of the entire AS. The router then determines shortest-path using Dijkstra's algorithm. Link costs may be set up by the administrator. If all the link costs are set to 1, then minimum-hop routing is achieved. A router

broadcasts link state information whenever a link's state changes. Links' states are also broadcast periodically even if there are no link state changes. OSPF advertisements are carried by OSPF messages.

Routing Information Protocol Routing Information Protocol (RIP) is another example of an IGP that is a distance vector protocol. RIP calculates cost in terms of number of *hops*, which is the number of subnets traversed along the path from source router to destination subnet. In RIP, maximum cost of path is limited to 15. Routing updates are exchanged between neighbors approximately every 30 seconds using RIP advertisements.

2.2 BGP and inter-domain routing

Routing between different administrative domains in the Internet is referred to as inter-domain routing. The Border Gateway Protocol (BGP) is the de-facto inter-AS routing protocol currently in use. The primary function of a BGP enabled router is to exchange network reachability information with other BGP enabled routers. The network reachability information includes information on the list of ASes that the reachability information has traversed. This information is used to make decisions about routing. In addition to routing based on reachability information, BGP also enables policy-based routing. Here, economic and political factors can play an important role in choosing among different potential paths. We first discuss the different message types exchanged in BGP and then discuss how business relationships may affect routing decisions on the Internet.

2.2.1 Reachability information

A router running BGP establishes TCP connections with neighboring BGP routers over port 179 and then initiates the BGP session [40]. The use of the Transmission Control Protocol (TCP) ensures reliable communication between the routers. After a BGP session has been initiated, both routers exchange their tables of active routes with each other.

The basic message that helps BGP routers to spread information about changes in the network is the UPDATE message. The UPDATE message can either contain *announcements*, implying changes in existing or new routes, or *withdrawals*, implying unreachability of a previously reachable prefix. BGP UPDATE messages have the following format:

prefix : Attributes,

where for each prefix several attributes can be listed. Some of these attributes are described below [40].

- **NEXT-HOP:** Defines the IP address of the border router that should be used as the next hop to the destination prefix listed in the UPDATE message.

- **AS-PATH:** Specifies the list of all the ASes that will be traversed if this path is selected. It also implies the sequence of ASes traversed by the update message. The right-most AS in this path is often referred to as the origin AS for the prefix(es) that this route concerns.
- **ORIGIN:** Defines whether the origin of the path information is IGP, EGP, or INCOMPLETE. The ORIGIN is IGP if the Network Layer Reachability Information (NLRI) is interior to the originating AS, is EGP if the NLRI is learned via BGP, and is INCOMPLETE if the NLRI is learned by some other means.
- **MULTI-EXIT-DISC:** Multi Exit Discriminator (MED) is used by a BGP speaker's decision process to discriminate among multiple exit points to a neighboring autonomous system. If two ASes connect to each other in more than one place, a router in one AS sends the MED to tell a router in another AS which path the second router should use to reach particular destinations.
- **LOCAL-PREF:** Used by a BGP speaker to inform other BGP speakers in its own autonomous system of the originating speaker's degree of preference for an advertised route.

A naive example of how BGP propagates reachability information and how AS-PATHS are updated is shown in Figure 2.2. Here, Linköping University (AS 2843) is allocated the prefix *130.236.0.0/16*, and the local ISP SUNET (AS 1643) is the service provider to AS 2843. When AS 2843 announces prefix *130.236.0.0/16* to SUNET, using BGP, SUNET accepts the announcements, appends its own AS number (1643) to the AS-PATH, and forwards the announcement to its provider NORDUNET (AS 2283). NORDUNET, a regional ISP, appends its own AS number (2283) to the AS-PATH and forwards the announcement to the ABILENE network (AS 11537). The ABILENE network, a transit ISP, announces the path for AS 2843 to the WORLDBANK network (AS 10497), which is a customer of ABILENE. Thus, AS 10497 knows the path to AS 2843, which follows the AS-PATH “*AS 11537, AS 2283, AS 1643, AS 2843*”. In this example we have shown one potential path. However, in reality WORLDBANK may have multiple different paths to reach AS 2843. The path is chosen based on policy decisions by the AS.

2.2.2 Route selection

A BGP router may learn multiple routes to the same destination from its neighbors. To select the best route, the BGP route selection process is followed, which selects the best route based on rules that satisfy local priorities such as profit and performance. Gill *et al.* [41] describe a simplified BGP decision process, which is shown in Table 2.1. Each route a BGP router learns is tagged with a local preference attribute based on the network's

Figure 2.2: Spreading of reachability information through BGP

Table 2.1: Simplified BGP decision process [41].

#	Criteria
1	Highest local preference
2	Lowest AS Path Length
3	Lowest origin types
4	Lowest MED
5	eBGP-learned over i-BGP-learned
6	Hot-potato routing (lowest IGP cost to border router)
7	If both paths are external, prefer the older path
8	Lowest router ID

routing policies. The route with the highest local preference value is selected. The local preference can, for example, be used to give preference to the customer routes over the provider routes. If the router has multiple paths with the same local preference value, then the route selection is based on AS-path length and the route with the lowest length is selected. If there are multiple routes with the same AS-path length, then the route with the lowest origin type is selected (IGP is preferred over EGP; EGP is preferred over incomplete). Similarly, the path with the lowest MED attribute value is chosen as the best route, if all of the previous criteria are the same. The router applies the remaining steps in the BGP decision process to select the best route to reach the destination.

The structure of the Internet is hierarchical. At the top there are a few global transit ISPs, referred to as tier 1 ISPs. The second tier primarily

Figure 2.3: BGP routing policies example, with three different types of paths available for the *source* to send traffic towards the *destination*: a provider path, a peer path, and a customer path, each indicated in the figure.

consists of national ISP operators, and the third tier primarily consists of local ISPs. An example of such a hierarchy is shown in Figure 2.3. Here, the local ISPs at the lowest level have to pay the entity at the national level ISPs for transit services. Similarly, the national ISPs pay the tier 1 ISPs for their services. To reduce the cost of transit service, ISPs often interconnect with each other to form peer links, as shown using red dotted lines in Figure 2.3, for example. This allows ASes at lower tiers to avoid sending the traffic through entities in higher tiers, thus allowing these ASes to save money. Two types of business relationships emerge from such interconnections: customer-provider relationships and peering relationships. In a customer-provider relationship, one ISP pays another to forward its traffic, and in a peering relationship, two ISPs send traffic intended for each other by using a peering link without involving any provider network. Peering relationships often hinge on the balance of traffic flow between the two networks; however, it is the policy of some networks that they will peer with any other network that is willing to peer with them, regardless of the balance of traffic flowing between the two networks.

Consider the case where a national ISP (indicated as *source*) is to send traffic to a local ISP (indicated as *destination*) in Figure 2.3. Here, the *Source* has three different routes to reach the *destination*. One path goes through the transit ISP which is a provider to *source*, indicated as the *provider path*. The second path goes through the peer of the *source*, indicated as the *peer path*. Finally, the third path goes through the customer of the *source*, indicated as the *customer path*. In most cases, in this type of scenario, the *source* network prefers the *customer path*, as sending traf-

Figure 2.4: Route aggregation on the Internet

fic through the *provider path* would incur costs for the *source*, and the *peer path* could skew the balance of power in the peering relationship, potentially providing an incentive for the peer network to start charging the *source* network [42].

2.2.3 Route aggregation and longest prefix matching routing

It was recognized in the late-90s that with the severe growth in number of ASes on the Internet, the growth of routing table sizes on Internet routers was getting difficult to manage by software, hardware, and people. To minimize latencies, improve network stability, and reduce the amount of resources needed for route calculations, a framework for route aggregation/summarization was formalized by Chen and Stewart [43] and is currently employed on all routers. We next describe how route aggregation is used on the Internet.

Consider the scenario shown in Figure 2.4. Here, a hypothetical ISP called ISP *shark* is the provider for four organizations (*i.e.*, *edge network 1*, *edge network 2*, *edge network 3*, *edge network 4*), each allocated prefix *130.236.180.0/24*, *130.236.181.0/24*, *130.236.182.0/24*, *130.236.183.0/24*, respectively. ISP *shark* can aggregate these prefixes to *130.236.180.0/22* before propagating the announcement to the Internet.

Assume ISP *Shark* acquires another ISP called ISP *Tuna*, and connects *edge network 4* to ISP *Tuna*. ISP *Tuna* is originally announcing prefix *192.165.5.0/24*. In this scenario, ISP *Shark* may continue to announce the aggregated prefix *130.236.180.0/22*, but will typically configure ISP *Tuna* to announce two prefixes: *192.165.5.0/24* (originally announced) and *130.236.183.0/24* (*edge network 4*).

When other routers on the Internet see prefixes *130.236.180.0/22* and *130.236.183.0/24* (where the latter is covered by the former one) and need

to route to an IP address within the $130.236.183.0/24$ address space, they will choose to route via ISP *Tuna* and not ISP *Shark*, since this prefix is longer and hence more precise. This demonstrates the longest matching prefix rule commonly employed by routers on the Internet.

2.3 Security issues with BGP

Despite being one of the most important infrastructures in modern society, the Internet is highly susceptible to routing attacks that allow an attacker to attract traffic that was not intended to reach the attacker [11, 15]. Recently, there have been increasing occurrences of routing attacks. Some of these attacks have been performed to gain access to sensitive or valuable information, or to circumvent constitutional and statutory safeguards against surveillance of a country's citizens, while others have been accidental or have tried to deny users access to certain services (for purposes of censorship) [9, 15, 44, 45]. With many of these attacks affecting regular end users, routing path integrity is becoming increasingly important for all stakeholders, including network operators, users, and governments.

The most glaring vulnerability with BGP is that it is not possible to determine whether or not the routing information carried by BGP is correct and it is difficult to check true ownership of prefixes. It is also not possible to confirm whether or not the neighborhood links announced in the AS-PATHS really exist. We will first discuss three main classes of attacks that are used for attracting traffic intended for a victim network: prefix hijack, subprefix hijack, and path hijack attacks. We will then explain how these attacks can be used for blackholing, interception, or imposture attacks.

Prefix hijack attack: In a prefix hijack attack, the attacker AS announces one or more prefixes allocated to some other AS(es), which the attacker is not permitted to announce. To illustrate this type of attack, consider the simple network topology shown in Figure 2.5(a). Here, the attacker AS (*Attacker*) announces the prefix that is allocated to the victim AS (*Victim*). Assume that the ASes in this example use a simple policy that chooses the shortest route to send the traffic to the destination prefix. In this case, the AS named *Global2* receives updates with the following AS-PATH values: “*Attacker*” and “*Global1, National1, Victim*” from *Attacker* and *Global1*, respectively, for prefix $66.174.0.0/16$. In this case, given the shortest path policy, the *Global2* will choose to route the traffic towards the attacker network (*Attacker*).

Subprefix hijack attack: In subprefix hijack attacks, the attacker AS announces one or more sub-prefixes of a prefix allocated to the victim AS, which the attacker is not permitted to announce. To illustrate this type of attack, consider the simple network topology shown in Figure 2.5(b). Here, attacker *Attacker* announces the prefix $66.174.0.0/24$, which is a sub-prefix of the prefix $66.174.0.0/16$ allocated to *Victim*. Following the longest matching prefix routing rule, *Global2* will choose to route the traffic to

Figure 2.5: Example attacks targeting BGP, used to attract traffic intended for a victim network.

Attacker for the hosts covered by the $/24$ prefix. In fact, other ASes will also make the same decision. Due to routers always selecting the longest matching prefix when choosing between path alternatives, subprefix hijack attacks are particularly serious.

Path hijack attack: In path hijack attacks, the attacker AS announces non-existing links to ASes so that traffic intended for the victim network is routed through its network. An example attack is shown in Figure 2.5(c) where *Attacker* announces the path “*Attacker, Victim*” even when there is no link between the *Attacker* and *Victim* networks. In this case, *Global2* has a shorter path through the attacker network and ends up choosing the route through the attacker network.

Outcomes of successful hijack attacks: The above attacks can be further classified based on the actions taken by the attacker. In *black-holing attacks* the traffic is terminated at the attacker and the originator of the packet does not see a proper response. These attacks are generally easy to

Figure 2.6: Example actions taken by the attacker networks and their impact on the victim node.

detect, and since the users do not receive responses for their requests, they can quickly raise a flag about something being wrong. An example of such an attack is shown in Figure 2.6(a), where the attacker network (*Attacker*) attracts the traffic and simply drops it.

Alternatively, the attacker network can present itself as an imposter and send malicious and fake responses on behalf of the victim network. Such attacks are called *imposture attacks*. An example of this type of attack is shown in Figure 2.6(b), wherein the attacker network (*Attacker*) sends fake responses to *Global2*.

It is also possible that, after a successful hijack attack, the attacker network might then relay the attracted traffic to the intended destination after altering the traffic or making a copy of the data. Attacks of this type are called *interception attacks*. An example of such an attack is shown in Figure 2.6(c). Here, the attacker network reroutes the traffic through *Global1* towards the victim network. It is easy to see that imposture and interception attacks are quite difficult to detect, since end users keep getting responses to their requests.

2.4 Countering threats to BGP

There have been different proposals to counter the vulnerabilities in BGP. Broadly, most of these proposals can be classified as techniques using route filtering, crypto-based prevention techniques and, anomaly-based detection techniques.

2.4.1 Route filtering

Route filtering and cross-checking of AS origin-prefix mappings against routing registries are the first line of defense against BGP routing attacks and other routing incidents. With route filtering, suspicious announcements are filtered. These suspicious announcements can include announcements with loopback addresses, or announcements that contain private AS numbers, exceptionally long AS paths, or prefixes with a prefix length greater than 24.

The Internet Routing Registry (IRR) is a public distributed database of AS routing policies and origin-prefix mappings, which is made available for the purpose of validating the contents of BGP announcements. IRRs can be used for route filtering by querying them using *whois* commands or by downloading the complete database. Network operators can specify and publish their networks' routing policies using the Routing Policy Specification Language (RPSL) [46], allowing other network operators to filter routes that are not registered in the IRR [11]. The IRR also provides information that helps with troubleshooting failures.

Although IRRs have the potential to allow networks to perform accurate filtering and help avoid many accidental and intentional routing anomalies, existing routing registries have many shortcomings. Today's IRRs are often incomplete and inaccurate, as information is not updated with sufficient frequency and many of the address delegations may have changed since the original prefix block was issued [11]. Such changes are often due to acquisitions and mergers of companies, companies going bankrupt, or further fragmentation occurring within prefix blocks. Furthermore, corporations often consider their routing policies and topological information to be strategically important and therefore may choose not to publish such information. For example, it has been noted that if strict route filtering were to be employed, 54% of the prefixes on the Internet would be unreachable¹. IRRs also come with their own security issues, as it is difficult to effectively guarantee the authenticity of the data accessed through the IRR.

We note that, to a large extent, route filtering relies on the ability to map origin ASes to prefixes. An alternative to the IRR could be to build such a service based on reverse Domain Name system (DNS) lookups. Currently, there is a special domain called *in-addr.arpa*, used for reverse DNS, which

---

¹A. Toonk, How accurate are the Internet Routing Registries , <http://bgpmon.net/?p=140>, Mar. 2009

allows reverse DNS lookup of the domain name that is associated with a given IP address.

Suppose we want to resolve the IP address *10.20.300.25*. First, a Fully Qualified Domain Name (FQDN) is created by reversing the IP address and appending the domain *in-addr.arpa* to it. In our example, the new FQDN *25.300.20.10.in-addr.arpa* is then resolved using DNS, starting from the right. At the top of the resolution hierarchy are the *arpa* and *in-addr.arpa* servers. The main task of these servers is to direct the resolver to the correct class A server, in our case the *10.in-addr.arpa* servers. Assuming a well configured reverse DNS structure, these can then direct us to the correct class B server responsible for *20.10.in-addr.arpa*. This process continues until the resolver obtains the domain name of the relevant IP address.

Rather than returning the domain/hostname when given an IP address, one can imagine an alternative service similar to reverse DNS lookup, which returns (i) the AS or (ii) the prefix range for a given IP address. The query could be iterative, similar to reverse DNS lookup.

Unfortunately, setting up an accurate and efficient resolution hierarchy for such systems is non-trivial. In fact, setting up these DNS delegations is as difficult as setting up the IRR. Furthermore, it has been found that reverse DNS delegations are incomplete [47]. For example, Jung *et al.* [47] observed that for a router at the university level, between 10% and 42% of lookups did not resolve. The most frequent cause of these errors was absence of inverse mappings for IP addresses.

2.4.2 Crypto-based architectures for BGP security

Cryptographic methods have been suggested to secure BGP. In the following we discuss some of the most prominent proposals and their use of hierarchical public key infrastructures.

Secure-BGP

Secure BGP (S-BGP) [19] uses the Public Key Infrastructure (PKI) to provide origin and path validations by attesting to the IP prefix ownership and the AS numbers in order to ensure that the AS path in a route advertisement is not modified or forged. An organization that is allocated prefixes or an AS number has a certificate signed by the issuer of prefixes or AS numbers. Such certificates can be used to verify the origin AS-to-prefix mappings, providing origin validation. Using PKI, *route attestation* certificates are generated by intermediate nodes for route announcements that they propagate. Route update recipients verify the origin of the prefix and the certificates of the nodes on the route that the announcement has traversed, providing path validation.

The primary problem with S-BGP and similar solutions is that they require cryptographic changes to the BGP protocol. These approaches also

require significant processing overhead for route announcements and modifications to the BGP routers themselves. Gersch and Massey [48] argue that many small and medium ISPs do not expect much of a loss if they should experience a hijack. These ISPs have little motivation to deploy an expensive solution such as S-BGP. S-BGP-like solutions will require a large critical mass of deployment or infrastructure before their objectives are achieved. Without enough ASes participating, even organizations that are seriously impacted by hijacks have little motivation to deploy such solutions. Some works have therefore considered incentive-based strategies to improve the deployment of S-BGP [26].

Secure Origin BGP (SoBGP)

In contrast to S-BGP, Secure Origin BGP (SoBGP) [20] only provides origin validations (no path validation). SoBGP uses *EntityCert* to bind an AS number to a public key corresponding to a private key that the AS uses to sign other certificates. The content of this certificate needs to be signed by another third party such as a top-level backbone service provider, or an authentication service provider such as Verisign. Once the public keys are distributed for each AS, the certificates, called *AuthCert*, are created. These certificates bind prefixes that an AS can announce to the corresponding AS number. These certificates can then be used to validate the origin ownership of prefixes. SoBGP can also be used to verify whether the advertiser of a given route actually has a path to the destination. Using certified lists of AS pairings for each AS, a static map of network topology can be created and used to validate the ASPATHs announced in BGP updates. SoBGP deployment faces similar drawbacks to those associated with S-BGP, as discussed in the previous section.

Resource Public Key Infrastructure

Currently, the Internet Engineering Task Force's Secure Inter-domain Routing (SIDR) working group is developing a specification for the Resource Public Key Infrastructure (RPKI), a cryptography-based architecture to support improved security for BGP [21]. RPKI can be viewed as an implementation of S-BGP [19], which addresses only a subset of the attacks that S-BGP is designed to address.

Resource Public Key Infrastructure (RPKI): The RPKI structure is similar to the existing resource allocation structure for IP addresses. At the top is IANA, followed by the Regional Internet Registries (RIRs), which themselves can issue certificates to local ISPs and Local Internet Registries (LIRs). When the addresses are allocated to the entity below in the hierarchy, the upper level entity will also issue certificates that attest to the allocation of IP addresses or AS numbers to the lower entity. The main certificates used in RPKI are Certificate Authority certificates and End-entity certificates.

- *Certificate Authority (CA) certificates:* Any resource holder that can sub-allocate resources is able to issue Certificate Authority (CA) certificates that correspond to those sub-allocations. Thus, CA certificates can be issued by IANA, each RIR, and ISPs/LIRs. These certificates do not vouch for the identity of a subject but are concerned only with allocation of address blocks to the holder of the private key corresponding to the public key specified in the certificate. That is, a resource certificate attests to an allocation of resources to a resource holder.
- *End-entity (EE) certificates:* This is a certificate issued to a person or system and is used for encryption, authentication, and digital signatures. The term *End Entity (EE)* is used to distinguish it from a Certificate Authority certificate. The main objective of EE certificates in RPKI is the verification of signed objects related to the number resources such as IP prefix range and AS number.

Neither the RPKI CA nor EE certificates verify whether the AS that is originating a prefix is authorized to do so. Instead, Route Origin Authorizations (ROAs) are used to assert such authorizations. ROAs allow a holder of an IP address range to create an object that explicitly asserts that an AS is authorized to originate routes to a given set of prefixes. A ROA is simply an attestation that the holder of a set of prefixes has authorized an autonomous system to originate routes for those prefixes. A set of IP addresses specified in a ROA should be contained within the block of IP addresses specified by EE certificates. A ROA is signed using the private key corresponding to the public key present in the end-entity certificate.

With RPKI, a distributed repository consisting of multiple databases is used to store and make accessible all the signed objects to LIRs/ISPs. When certificates are created they are uploaded to this repository. Any downstream organization can have their upstream provider submit ROA requests on their behalf. A LIR/ISP can validate all ROAs by acquiring all the certificates from the repository. These databases are distributed among registries such as RIRs, NIRs, LIRs/ISPs. Registries are advised to maintain copies of repository data from their customers. Each RIR is expected to hold PKI data from all entities within its geographical scope.

Route Origin VERification

ROVER is a complementary approach to cryptographically secure BGP route origins [48]. ROVER is a method for publishing BGP route origins for IP prefixes in the pre-existing reverse-DNS infrastructure, which builds upon DNSSEC [49] and reverse-DNS services. Unlike RPKI, which requires separate infrastructure for certificates and distribution of signed information, ROVER does not entail such requirements as it is based on DNSSEC. ROVER uses two new DNS resource record types.

- *The Route Lock (RLOCK) resource record*: This resource record is used to indicate that a reverse DNS zone has enabled BGP route publishing. The RLOCK record is present at the apex of a reverse DNS zone. All route announcements that map to this zone will be invalid if they do not have an SRO record.
- *Secure Route Origin (SRO) record*: SRO record identifies origin ASN(s) for a prefix. The *Origin ASN*, a mandatory field in an SRO record, specifies an AS number that is authorized to originate a route announcement for the prefix corresponding to the SRO record's reverse DNS name [48].

It is required that the RLOCK and SRO records must be signed with DNSSEC. BGP routing announcements can be authenticated, once an organization publishes ROVER data in the reverse DNS. ROVER is being extended to address path hijacks and policy violations. ROVER has already been implemented in a public testbed.²

Discussion of RPKI and ROVER

A lot of standardization work has been done on RPKI, and the adoption of RPKI is gaining momentum. Although RPKI is incrementally deployable in parallel to the current operation of BGP, it does need a separate infrastructure to store and retrieve signed objects such as ROAs and certificates. Such infrastructure must be distributed with the capability to disseminate information quickly, and entails additional costs for the adopters of RPKI.

Cooper *et al.* [50] also suggest that a new ROA can cause many routes to become invalid if a new ROA for a large prefix is issued before ROA's for its subprefixes are issued. Thus, initial deployment of RPKI will require issuance of certificates to existing resource holders as an explicit event. With millions of prefixes and thousands of ASNs already allocated, this step can be seen as another hindrance in the deployment of RPKI.

RPKI provides the service of origin validation. However, it does not provide cryptographic assurance that the origin AS in a received BGP route was indeed the originating AS for this route [51]. To address this shortcoming BGPsec is currently being developed. BGPsec builds upon RPKI so as to also provide AS-PATH validation. However, AS-PATH validation in BGPsec relies on online cryptographic signatures that are deemed expensive [15]. Additionally, BGPsec cannot offer expected security unless every AS on a particular path enables BGPsec.

Another significant side effect of RPKI deployment is that it alters power and economic relations and raises strategic and policy issues [52]. In the current scenario, the responsibility for address allocation is detached from the

---

²BGP ROVER: Route Origin Verification, <https://rover.secure64.com/>, Mar. 2014.

operational responsibility for routing. RPKI negatively affects this decoupling, as it enables RIRs to impact routing on the Internet. Cooper [50] discusses the ways in which an authority can achieve stealthy revocations of a child's objects such as *CA certificates* and ROA. Political and business considerations might make it unacceptable that there is significantly more control vested in each parent node in the path up to the root of the RPKI hierarchy. Recent revelations about government sponsored online surveillance have raised concerns regarding systems that can be controlled by central authorities.

Finally, the deployment of RPKI could lead to circular dependencies resulting in long-term failures. In RPKI, objects are stored at directories that are controlled by their issuer. Parties that rely on these objects should have access to these directories to download and verify RPKI objects. Such RPKI objects are accessed over TCP/IP, which requires a valid or unknown BGP route to the issuer directory. However, RPKI objects can affect the availability of BGP routes, therefore also of TCP/IP, the very infrastructure over which they are delivered [50].

At first glance, ROVER may appear to have an advantage over RPKI. It proposes using the existing DNS infrastructure to authenticate origin. ROVER also builds upon DNSSEC, which has been in development since the mid-90's. However, the root of DNS was signed only recently, and except for a few countries, the adoption of DNSSEC has not picked up. Despite years in development, less than 0.5% of all domains had signed up for DNSSEC by Aug. 2016.³

Another issue with ROVER is that it requires manual configuration of the reverse DNS, resulting in many of the same problems as with creating and updating IRRs. For example, Jung *et al.* [47] observed that for a router at the university level, between 10% and 42% of reverse DNS lookup results were negative. The most frequent cause of these errors was absence of inverse mappings for IP addresses.

Due to the recent exposure of US-based NSA snooping incidents, the possibility of adoption of DNSSEC/RPKI could be further delayed. The root of the signed DNS is based in the US, and other countries may be critical of this fact when considering adopting DNSSEC. Similarly, RPKI also needs trust anchors, which typically would be under the control of the US-based ICANN/IANA. Based on the above discussion, it is likely that it will take many years before reaching a critical adoption level for effective RPKI and ROVER usage.

---

³Verisign Labs, Domains Secured with DNSSEC, <http://scoreboard.verisignlabs.com/>, Aug. 2016

2.4.3 Anomaly detection based techniques

Anomaly-based techniques are based on detecting anomalies in the routing and normally do not entail any changes to the existing BGP protocol. These techniques are typically either based on control-plane data (*e.g.*, routing related data such as route announcements in BGP) or data-plane data (*e.g.*, packet headers and payloads). A few hybrid techniques have also been proposed.

Control-plane based anomaly detection

There are several works that are based on control-plane data for the detection of anomalies in BGP routing, including PG-BGP [23] and PHAS [22]. The basic premise that these proposals build on is that an IP prefix should generally only be originated by a single AS. An IP prefix originated by more than one AS results in a multiple origin AS (MOAS) conflict. Sometimes, a MOAS could be legitimate and can then typically be observed over a long period of time [53]. However, a newly detected MOAS conflict can be an indication of a prefix hijack. The works by Kruegel *et al.* [54], Lad *et al.* [22] target detecting such MOAS cases.

PG-BGP [23] suggests treating unfamiliar routes cautiously, adopting potential bogus routes slowly, and allowing time for secondary processes to check their validity. While PG-BGP will help in slow decimation of bogus routes, it requires significant participation among the large providers. With PHAS, new prefix ownerships are always compared against historical route data from RouteViews, and prefix owners are informed any time a new origin AS is associated with their prefix.

Kruegel *et al.* [54] propose a technique that is based on utilizing geographical location data from a *whois* database and topological information in an AS connectivity graph to detect malicious BGP updates. They propose building an AS connectivity graph by passively monitoring BGP update messages. They observe that a valid AS-PATH contains at most a single sequence of core nodes (ASes of the Internet backbones such as large ISPs) and a path that has traversed core nodes and enters a periphery node (*e.g.*, a local provider, company, or university) never returns to the core nodes. They check the AS-PATH attribute of the update messages to determine if the sequence of ASes in the AS-PATH satisfies the above constraint to detect violations.

The major concern with these systems is that they require the updates collected at all participating BGP routers (or RouteViews servers) to be processed at a centrally administered location. Such proposals can therefore result in significant processing overhead at centrally administered nodes. There are also the inherent questions about who will own and operate such a system and any other problems that are associated with having a central point of failure. Additional concerns with such mechanisms is that they require privileged access to live BGP announcement data, making such ap-

proaches less appealing and difficult to deploy in practice.

Data-plane based anomaly detection

Subramanian *et al.* [55] use data-plane information collected using passive monitoring of the TCP flows for the IP addresses of the corresponding prefixes to check whether the underlying routes to different destinations work. A TCP flow is marked as complete if at least one data packet is observed after a SYN packet; otherwise the TCP flow is marked incomplete. When passive monitoring fails and they do not appear to observe a complete TCP handshake followed by at least one TCP packet, then they actively probe using ping, traceroute, and a port 80 scan on the destination IP address to validate the reachability of the prefix.

iSPY [56] is a prefix-owner-centric data-plane-based hijacking detection system. The system focuses on detecting normal prefix hijack events. Sub-prefix hijacks are not targeted and interception attacks cannot be detected by iSPY. ASes deploying iSPY need to monitor only transit ASes, which are few compared to the number of ASes at the edge. iSPY keeps track of the transit ASes networks' reachability information in the form of vPaths, where a vPath is the set of AS-level *forward paths* from a network employing iSPY to the core ASes on the Internet. Normally, traceroute is used to obtain these forward paths. When prefix hijack attacks are encountered on a victim network, the new vPath set returned is incomplete. If AS X is polluted by a prefix hijack attack, replies to traceroutes from a victim network to a network owned by X will not reach back to the victim network. This will result in cuts in the vPaths, where a cut is a failed link on the forward path. To distinguish prefix hijack attacks from normal network problems, the authors argue that the number of cuts resulting from prefix hijacks are larger than the number of cuts resulting due to other "normal" network problems. To improve probing coverage, they use a combination of a traceroute, ICMP ping and TCP ping messages.

Zheng *et al.* [29] present a data-plane-based approach to real-time detection of prefix hijack attempts. They also target imposture (attacks in which the attacker responds to senders, mimicking the hijacked destinations) and interception attacks, along with prefix hijack attacks. Their approach is based on network distances (measured in terms of router hops) from their monitors, which keep probing the network location of a target prefix. They observe that variations in network distances are not significant from these monitors on a normal day. However, if there is any form of hijack attack, then noticeable network distance changes occur. Therefore, if such network changes are observed, it is regarded as a sign of a possible prefix hijack event. The authors note that not all network distance changes are due to prefix hijack events. A mechanism to distinguish legitimate network changes and hijack incidents is incorporated into their approach.

Combined Control/data-plane anomaly detection

Other works have combined control-plane and data-plane data analysis for detection of routing anomalies [57, 58]. For example, Ballani *et al.* [58] use a simple signature to detect possible interception in the Internet. They consider a prefix p with origin AS O and next-hop ASes $N_1, \dots, N_n$. The next hop AS information is retrieved from the control-plane data. Next, they look at the traceroutes collected in the iPlane project [32]. A traceroute that reaches AS N_j , that belongs to the set of next-hop ASes for AS O , should be routed directly to AS O . If a data-plane trace is observed where this condition is not satisfied, that is, the next hop from N_j goes to AS N_i that does not belong to the next-hop set of AS O , then the anomaly is identified as a possible interception case. They tried to account for all such cases by improving IP-to-AS mappings, as well as validating cases based on data-plane and *whois* information.

Wong *et al.* [57] use data-plane verification of the AS-PATHs seen in the control plane data. Their proposed method sends probe packets that are indistinguishable from normal data packets except for the two entity ASes involved in the verification process. With this method the verifier node that wants to verify the route path shares a secret with the prover node, which assists the verifier in the verification process.

Shi *et al.* [59] propose the Argus system to correlate control-plane routing information and data-plane reachability data from a number of distributed diagnosis nodes. Argus mainly focuses on detecting black-holing attacks by exploiting the fact that affected nodes (due to black-holing) cannot communicate with the hosts in the victim prefix, while unaffected routers can.

Finally, Hu *et al.* [60] present a method of detecting hijacking attacks in real time, which combines passively collected BGP routing updates (control-plane data) with data plane fingerprints of edge networks. They target prefix hijack, subprefix hijack, and interception attacks in their work. The main insight on which the authors build their system is that *a successful hijacking will result in conflicting data plane fingerprints describing the edge networks* to which the prefix is allocated. Fingerprints of edge network consist of host-based information such as host uptime and network-based information such as firewall policies.

2.5 Attacks involving edge networks

As discussed in Section 1.3, besides the routing attacks that ultimately affect the edge networks, there are several other malicious activities that edge network have to frequently confront. Examples of such activities are spamming, scanning, denial of service, and distributed denial of service attacks. In this thesis, we focus on spamming and scanning attacks that target the edge networks and the users in the edge networks.

2.5.1 Spamming

Email has been one of the most widely used application over the internet. Billions of emails are sent and received everyday over the internet [61]. Unfortunately, most of the email that are sent are spam; *i.e.*, unsolicited bulk email that involves nearly identical messages sent to numerous recipients. According to Symantec report in 2016 [61], more than half of inbound business email traffic was spam in 2015.

Over the years spam detection techniques have been getting better. However, spammers are devising advanced techniques to avoid spam mails being detected. Spammers have been using various advanced techniques for their spamming campaigns, including using botnets to send large volumes of spam or to avoid blacklisting, content obfuscation of emails to mislead filters, exploit BGP to steal IP addresses to send spam mails, and using compromised web-mail accounts to send spam emails.

Total cost of spam is high. First, large amounts of processing power is consumed in processing the huge amount of spam mails that mail servers see today. Additionally, spammers are getting better at evading content-based filters and are more careful in choosing their targets. This has resulted in an increase in successful cases of phishing attacks and the spreading malware of through email spam. In addition to causing nuisance for users and network operators, spamming also require unnecessary spendings in terms of the cost incurred to develop and deploy anti-spamming techniques.

Given the fact that spammers are developing advanced strategies to send spam, it is necessary that users and organizations collaborate and share information with each other so as to get the upper hand in the fight against the spammers.

2.5.2 Scanning

Scanning is a common way for miscreants to look for vulnerable servers or hosts. With these techniques the attackers aim to find open ports and services running on a server or host. With vertical and horizontal scanning, an attacker scans multiple ports from the same server/host or the same port from multiple servers/hosts, respectively.

Several techniques such as TCP connect, TCP half-open scanning (SYN scan), FIN scan, fragmented packet port scan can be used to achieve scanning. Recently a mechanism where one can scan the entire internet (IPv4 address space) was presented by Durumeric *et al.* [62]. Although such technique enables one to find computers with security vulnerabilities rapidly, it also enables attackers to rapidly identify unpatched vulnerable computers and compromise them.

Port scanning attacks have been studied extensively. While Yegneswaran *et al.* [63] characterize scanning from a global perspective, Allman *et al.* [64] present characterization over a longer period of time (starting from 1994 through 2006). These studies show that the extent of the increase in scan-

ning activity on the Internet over the years has been manifold. Raftopoulos *et al.* [65] perform a measurement study after an Internet-wide scanning was mounted by Sality botnet in February 2011 to evaluate what happens to the scanned hosts. Their study shows that 2% of the scanned IP addresses responded to the scanning event and 8% of the repliers were eventually compromised. They also observed that 352,350 IP addresses were used to perform the scanning event. Another measurement study by Arlitt *et al.* [66] evaluate what information about targeted IT infrastructure can be gathered by external third-party organizations using scanning.

2.6 Peer-to-peer networks

A peer-to-peer network (P2P) is a type of decentralized and distributed network architecture in which individual nodes in the network act as both suppliers and consumers of resources. This is in contrast to the centralized client-server model where client nodes request access to resources and services provided by central servers. Broadly, there are two classes of P2P systems:

- **Unstructured P2P systems:** With unstructured P2P systems, peers organize themselves into a random overlay structure. The overlay can be either flat or hierarchical, but typically requires the use of ad-hoc flooding techniques to query content stored within the overlay network [67]. Freenet, Gnutella, and BitTorrent are some examples of unstructured networks. Unstructured P2P networks have issues with scalability, fault tolerance, and providing lookup guarantees.
- **Structured P2P systems:** Structured P2P systems use a Distributed Hash Table (DHT) to provide a lookup service similar to a hash table. DHTs are a class of decentralized distributed systems that store (key, value) pairs and allow any participating node to effectively retrieve the value associated with a given key. DHTs form an infrastructure that can be used to build more complex services. Each node in these overlay networks maintains a set of links to other nodes (in the form of a forwarding table). Together these links form the overlay network. DHTs have the following properties:
  - Decentralization: No central control required.
  - Scalability: The system functions efficiently even with thousands of nodes in the network.
  - Fault tolerance: The system is reliable even with nodes continuously joining, leaving, and failing.

DHT-based P2P systems assign uniform, random, and unique node identifiers (NodeIDs) to the peers. Each peer maintains a small routing table

consisting of its neighbor peers' NodeIDs and IP addresses. Lookup queries or messages can be forwarded across overlay paths to peers in a progressive manner. DHTs such as Chord [34], Pastry [68], Tapestry [69], and CAN [70] have been proposed. We explain the Chord protocol in detail in the following section.

Chord protocol for DHT based P2P: Chord uses a circular identifier space with 2^m unique identifiers, where m is a positive integer. Holder nodes and keys are mapped to the circular identifier space, with each node being responsible for all key-value pairs that map to identifiers with values smaller or equal to its own identifier, down to the identifier of the node with the next lower identifier, using modulo arithmetic to allow the identifier space to wrap around itself.

The nodes are arranged into an efficient overlay structure in which each node maintains pointers to its successor (the node following them in a clockwise direction), its predecessor (the node prior to it), as well as a finger table with shortcuts to nodes at other parts of the circular identifier space, with each shortcut being spaced exponentially further away from the node itself. Long shortcuts are used to quickly reach the correct region on the Chord circle, and increasingly shorter shortcuts are used to close in on the target node responsible for a particular key.

By extending Chord, we are able to leverage some of Chord's highly desirable properties.

- **Basic assignments:** Given a key, it is easy to identify a unique holder which is responsible for storing the corresponding key-value pairs.
- **Lookup guarantees:** In steady-state, the holder of a given key will be found if it exists. Furthermore, the holder will be found by forwarding only $O(\log N)$ messages to other nodes, while each node must store information about only $O(\log N)$ other nodes.
- **Node churn and basic recovery:** Chord can effectively handle nodes joining, leaving, or even failing. To account for such node churn, each node maintains updated information about its successor node on the circle. If a node fails/leaves, the lookup will eventually be resolved correctly using the information stored at the successor.

P2P applications: P2P-based networks have been used for many different applications. The most commonly known application of P2P is file sharing systems such as BitTorrent [71]. P2P is being used in digital currencies such as Bitcoin [72] and Peercoin [73]. A P2P network is also used in connection sharing applications such as OpenGarden⁴ to share Internet access with other devices. DHT-based P2P networks have been used to improve the scalability of a large range of distributed systems, including group

---

⁴"Open Garden", www.opengarden.com

communication [74], co-operative web caching [75], distributed reliable storage systems [76], secure and co-operative messaging systems [77], and for scalable anonymity protocols [78].

2.7 Intrusion detection

Intrusion detection is used to detect unauthorized computer system access or cases where legitimate users exceed their privileges. Intrusion detection systems (IDS) can be classified as either host-based or distributed intrusion detection systems. Host-based IDS monitors and analyses the internals of a computing system as well as network packets exchanged via its network interfaces [79]. However, attackers are using increasingly sophisticated techniques to launch attacks using a large number of compromised hosts which allows them to spread their attacks over a wide geographical area. Such co-ordinated attacks make it difficult for defending networks to detect attacks and sources of attacks. In order to detect such sophisticated large-scale attacks, there has been a growing need for collaborative effort to detect and spread information about intrusion sources and compromised hosts [80, 81]. This need has paved the way to the development of distributed IDS.

Distributed IDS consists of multiple Intrusion Detection Systems (IDS) over a large network, all of which communicate with each other or with a central server, which can assist in detecting network-wide attacks that would otherwise be undetectable by conventional host-based IDSes. Normally an IDS can be considered as consisting of two main parts: a *detection unit* and a *correlation unit* [82]. The detection unit focuses on detecting low-level intrusion attempts at host machines or subnetworks. The correlation unit gathers multiple low-level alerts from multiple detection units and combines them to create high-level intrusion reports. The placement of correlation units determines how information sharing is accomplished in a distributed IDS. Distributed IDSes can be classified as centralized, hierarchical, or decentralized IDSes depending on where correlation units are placed. An extensive survey of such systems is presented in [82].

2.7.1 Centralized IDS

Distributed Intrusion Detection System (DIDS) [83] is an example of an IDS where detection units are distributed and send monitoring information to a centrally located correlation unit, where the information is analyzed. DIDS consists of a DIDS director, one host monitor per host, and one LAN monitor for each LAN segment of the monitored network. Information is collected by different monitors, which send the information to the central director for analysis using a rule-based expert system.

DShield⁵ collect logs from agents, which may be *e.g.*, a personal firewall

---

⁵DShield, “Internet Storm Center”, <http://dshield.org/about.html>, Mar. 2014

on a home user computer, a commercial IDS on a large company's network perimeter, or a host-based IDS on a network of an educational institution. These agents constantly send information about unwanted traffic to the DShield database. These data are analyzed by volunteers or machines looking for attack trends and abnormal behavior. The resulting analysis results are posted on the Internet Storm Centre's (ISC) website, which can be seen by Internet users or downloaded by simple automated scripts. The primary focus of ISC is to provide a global Internet Security status so that Internet users and security personnel can be informed about current security attacks.

The main issue with centralized processing of alerts generated by agents is that it leaves a central point of failure. For example, if a correlation unit is attacked, the system will fail to detect and send alerts about distributed attacks. Even if fail-safe measures are implemented, the issues of scalability and efficiency still remain unresolved.

2.7.2 Hierarchical IDS

Hierarchical approaches have been proposed to overcome limitations posed by centralized IDS. With these approaches, collaborative IDSes are partitioned into multiple small communication groups based on features such as: geography, administrative control, and similar software platforms [82]. There is typically a correlation unit for each group, which is responsible for gathering and analyzing the data from the members of the group. This correlation unit sends the processed data to a node at a higher level in the hierarchy where the data is further analyzed.

The Distributed Security Operation Center (DSOC) [84] system comprises data collectors, remote data collectors, local analyzers, and a global analyzer. A data collector can be a host, a server, a firewall, an IDS, or any system that generates logs. All data collectors on the same site send the logs to a Local Intrusion Database (LIDB). Remote data collectors collect logs coming from critical data collectors and forward those logs in the LIDB of another site. A local analyzer analyzes formatted logs located in the LIDB and generates alerts and can also merge similar alerts. Alerts generated by the local analyzer are forwarded to the global intrusion detection database. The global analyzer is responsible for global intrusion detection in a network. It analyzes alerts from the global intrusion detection database and detects more sophisticated intrusions.

The Event Monitoring Enabling Responses to Anomalous Live Disturbances (EMERALD) [85] system focuses on detecting intrusion detection in a large enterprise network with thousands of users connected in a federation of independent administrative domains. Each administrative domain is composed of a collection of local and network services that provide an interface for requests from individuals, internal and external to the domain. In the EMERALD hierarchy, *service analysis* covers the surveillance of misuse of individual components and network services within the boundary of

a single domain. *Service analysis* is at the lowest tier in hierarchy. *Domain-wide analysis* sits in the middle tier and focuses on misuse across multiple services and components while *enterprise-wide analysis* sits at the highest tier and performs surveillance to detect coordinated misuse across multiple domains. EMERALD combines signature analysis with statistical profiling over the operation of network services and infrastructure.

Although hierarchical architectures scale better compared to centralized architectures, the processing capacity at the nodes in the higher levels still limits scalability and their failure can lead to failure (*e.g.*, to analyze alerts sent by the nodes below it in the hierarchy). Furthermore, nodes at the highest level might have a limited ability to detect sophisticated attacks as the data forwarded to them is at a higher level of abstraction.

2.7.3 Decentralized IDS

A number of approaches to decentralized architecture for correlation of distributed collected alerts have also been proposed. For example, DOMINO [86] uses Worminator, a tool to automate the process of creating signatures for SMTP based worms, at hosts that participate in the system. The Worminator parses alerts from a host IDS into the form of a watch list that is encoded using Bloom filters. The construction of Bloom filters helps to protect the confidentiality of the data being exchanged between different administrative domains. A scheduling algorithm is used to dynamically calculate subsets of correlation peers that communicate to exchange Bloom filters. Thus, the exchange of information that occurs among members within each subset makes the communication cost effective in terms of bandwidth and processing power and also avoids the need to collect data from detection units at a central location.

Dash *et al.* [80] propose a system composed of three primary subcomponents: local detectors, global detectors, and the information sharing system. Local detectors perform host-level intrusion detections, global detectors perform system-wide detections, and the information sharing system focuses on sharing state between the local and global detectors. Local detectors are placed at the end-hosts. The global detectors can be located in a central location or in a distributed manner at each end-host. It is assumed that a local detector communicates its state by beaconing to a random set of global detectors at regularly spaced epochs. A local detector is heuristic-based detector that counts the number of new outgoing connections to unique destination addresses and outgoing ports. Alerts are raised when a certain threshold is crossed. Alarms raised by local detectors spread in the network using the information sharing system and are aggregated by the global detectors to determine if the network is under attack. The authors propose a probabilistic model to detect anomalies at the global detectors.

2.8 Discussion

Despite an increasing number of observed routing attack occurrences [9, 15, 44, 45], it has proven difficult to incentivize operators to invest in existing solutions [11], and there is currently no universally deployed solution that prevents hijacking of Internet traffic by third parties [15]. For example, the deployment of crypto-based efforts [19–21] has been hampered by high deployment costs (e.g. cost associated with upgrading BGP servers to accommodate for the higher processing needs of cryptographic solutions) for network operators [11, 15]. Additionally, there is no visible incentive in using those proposals when the current form of BGP still offers adequate performance.

Alternatively, monitoring of path announcements and the data paths taken by data packets can be used to identify potential hijacks and other suspicious data paths [22, 23, 56, 87]. With routing paths being determined by the individual routing decisions of many involved operators and other organizations running their own Autonomous Systems (AS) [23, 45, 56], such techniques benefit greatly from information sharing between ASes. In spite of large number of anomaly detection based techniques having been proposed, none of them is actively used on the Internet. Many of these techniques are promising, but the main issue is that these techniques often need to be operated under the control of a single administrative domain. This raises questions regarding who should operate and pay for such systems. It also creates a central point of failure. High processing overhead and high false positives are other concerns with such mechanisms.

Monitoring the activities of organizations and their networks can open up possibilities for attack detection based on both control-plane and data-plane data. Such a combined monitoring technique has proven to be advantageous [18]. For example, Ying *et al.* [18] show that networks that are worm-infected, spam origins, and/or the source of other anomalies also exhibit more anomalous control plane (routing) behavior. This has also been demonstrated in work by Ramchandran and Feamster [88] who found that control-plane attacks such as prefix hijacking are exploited by spammers to perform attacks such as sending spam. Thus, an anomaly in the control-plane can help to effectively monitor the data-plane activities of concerned networks and help reduce cost and delay in response to various attacks. There is a need for a system that can achieve such an objective.

Although IDSeS have been designed to target a wide range of attacks, including spamming, scanning, denial of service, and worm infection, IDSeS typically only employ signature-based or anomaly-based detection in which packet headers and content are analyzed. Because IDSeS focus on data-plane data for their detection purpose, not control-plane data, IDS solutions typically cannot be used to detect and prevent routing level anomalies. Thus, they often cannot aid in correlating control-plane behavior with data-plane behavior, something which has been shown to be advantageous in detecting

more sophisticated attacks.

Various challenges arise from designing a scalable collaborative system that leverages both control-plane and data-plane information. To work with control-plane data such as BGP updates one has to focus on prefixes, their originating ASes, and AS-PATHS advertised in BGP updates. In contrast, to work with data-plane data one has to focus on individual IP addresses and host machines. To evaluate the behavior of networks, one has to map individual host machine's activities to the network to which that host machine belongs. In this thesis, we present PrefiSec, a framework that enables prefix-based (network) evaluations. We also show how the PrefiSec framework can be augmented by our TRAP framework, which allows reputation-based evaluation of hosts.

Attackers are increasingly devising sophisticated techniques to mount stealthy attacks that are difficult for a single individual network to detect. There is also a need for collaboration among different network entities to effectively mitigate challenges from attackers. However, such a collaboration could entail processing of huge amounts of data (control-plane and data-plane). Thus, any system that attempts to do this should be scalable and avoid any form of centralization so that there are no bottlenecks in the system. This requirement motivates the use of DHTs in PrefiSec.

DHTs have been used effectively to implement decentralized and scalable systems. There have been a number of applications developed that use DHT-based P2P networks to achieve decentralization and scalability in systems [74–77]. We take inspiration from such systems and propose an extension that enables prefix-aware DHTs to create the overlay network.

We have described a number of works in this chapter that perform collaborative control-plane data and data-plane data analysis to mitigate attacks that target wide-area routing and edge-networks (spamming and scanning). Collaboration among network entities has been proposed to protect against or mitigate a number of different types of attacks, including Distributed Denial of Service (DDoS) attacks [89, 90], scanning [91], spam [92], and BGP routing [93] attacks. Closely related to the systems presented in this thesis are systems that allow reporting of ratings [92] or the use of tamper-evident logs that can be exchanged with other participants to detect malicious behavior [93].

In contrast to these works, we focus on the general information exchange between organizations and present a unifying framework, PrefiSec, that allows information about different types of network attacks to be combined. We do, however, leverage some of the above solutions and use some of these types of attacks to demonstrate the generality and functionality of our framework. While PrefiSec enables collaboration among ASes, we also foresee a user-driven approach to detect and raise alerts for stealthy interception and imposture attacks. To this end, we propose, CrowdSec, which is low in processing overhead and runs at the end systems. The low overload is achieved by leveraging passive measurements data instead of performing active mea-

surements.

Evaluation of previously proposed routing attack prevention and detection have shown that if few large ASes spread across the globe use those mechanisms then the security gains are substantial and good protection against routing attacks and accidents can be achieved. However, none of the proposed routing mechanisms have been widely deployed. One major factor that might contribute to this lack of deployment is the inherent difficulty in achieving consensus among large ASes, which are spread across the globe, on using a common routing security mechanism. This is further complicated by diverging political, financial, or business considerations.

We consider the possibility that ASes within the same region such as North America or European Union are more likely to collaborate with each other due to government or regional legislation or agreements among ASes within the region to share information to mitigate the effects of routing attacks and incidents. Given this consideration, in this thesis, we evaluate how the attack detection/prevention rates of previously proposed routing mechanisms are affected when such locality aspects in the deployment of these mechanisms is considered. We also evaluate the effect of the number of ASes and the size of the ASes that deploy this mechanisms on the attack detection/prevention rates for these mechanisms.

Chapter 3

A Characterization Study of the China Telecom Incident

3.1 Introduction

On April 8, 2010, AS 23724, an autonomous system (AS) owned by China Telecom, announced approximately 50,000 prefixes registered to other ASes. These prefixes included IP prefixes registered to the US Department of Defense¹, which caught the attention of the US-China Economic and Security Review Commission [94]. Unlike previous routing misconfigurations, *e.g.*, the *Pakistan Telecom incident*² or the *AS7007 incident*³, China Telecom's network had the capacity to support the additional traffic that the announcements attracted. Furthermore, there is ample data-plane evidence suggesting that during the incident, Internet traffic was reaching its correct destination. This unique situation is what led some to suggest this was an attempt to intercept Internet traffic.

While the China Telecom incident has garnered attention in blogs, news outlets, and government reports [94], there has been no prior academic attempt to understand this incident. This dearth of understanding is especially apparent when considering the many questions that remain unanswered about this incident. These include (1) understanding properties of the hijacked prefixes, (2) quantifying the impact of the event in terms of subprefix hijacking, and (3) explaining how interception was possible. We tackle these questions using publicly available control- and data-plane measurements and highlight what types of data would be useful to better un-

---

¹J. Cowie, Renesys blog: China's 18-Minute Mystery, <http://www.renesys.com/blog/2010/11/chinas-18-minute-mystery.shtml>, Nov. 2010

²M. Brown, Renesys Blog: Pakistan Hijacks YouTube, http://www.renesys.com/blog/2008/02/pakistan_hijacks_youtube_1.shtml Feb. 2010

³S.A. Misel, Wow, AS7007!, <http://www.merit.edu/mail.archives/nanog/1997-04/msg00340.html>, Apr. 1997

derstand routing anomalies in the future. We emphasize that while we are able to characterize the incident and show evidence supporting the hypothesis that this incident appears to be an accident, there is currently no way to distinguish between “fat finger” incidents and those that have malicious intent based on empirical data alone.

Characterizing such large scale incidents has several benefits. Our study exposes the challenges and limitations of characterizing such large scale routing anomalies, which may help in characterizing future routing anomalies. Network administrators can effectively use this type of information to convince business decision makers to upgrade and develop new methods to counter such routing anomalies. Researchers may use this type of information to identify open problems in this domain. A concrete study of routing incidents can also help companies develop a better understanding of the risks involved to their networks, even when they have employed enough mitigation techniques within their own administrative domains.

Besides better understanding *why* new techniques need to be developed, the study also helps determine *how* new mitigation techniques can be developed to combat routing anomalies. In fact, the methodologies developed in this chapter are leveraged in later chapters, in which we develop techniques that can be employed to raise alerts about possible routing anomalies.

3.1.1 Insecurity of the Internet’s routing system

Attacks due to vulnerability in the Internet’s routing system are discussed in Section 2.3. For the sake of clarity, we discuss them again briefly here in the context of the China Telecom incident. Routing security incidents have happened repeatedly over the past 15 years [95]. These incidents typically involve an AS originating an IP prefix without the permission of the autonomous system (AS) to which the prefix is allocated: *hijacking*. Usually when hijacks happen, the misconfigured network either does not have sufficient capacity to handle the traffic, as observed during the *AS7007 incident*, or does not have an alternate path to the destination, as observed during the *Pakistan Telecom incident*. In these cases, the impact of the incident is immediately felt as a service outage or interruption of connectivity.

More troubling are cases of traffic *interception*, where traffic is able to flow through the hijacking AS and on to the intended destination. Without continuous monitoring of network delays or AS paths [58], incidents such as these are difficult to detect, thus creating an opportunity for the hijacker to monitor or alter intercepted traffic. Traffic interception was demonstrated in 2008 [12] and more recently occurred during the China Telecom incident.

Without an extensive monitoring infrastructure it is extremely difficult to measure the impact of the China Telecom incident. In this chapter, we define criteria that allows us to infer potential interceptions using only control-plane data. We use data-plane measurements [32] to validate our criteria and characterize the AS topologies that allowed for inadvertent interception.

3.2 Related work

While the China Telecom incident occurred in April 2010, it received little attention until November 2010 when the US-China Economic and Security Review Commission published their report to Congress [94], which included a description of the event. After the release of the report, the incident received attention in news articles and was investigated by some technically-oriented blogs such as BGPMon⁴ and Renesys⁵.

BGPMon, an organization that provides monitoring and analysis of BGP data, performed the first investigation of the China Telecom incident. Using control-plane measurements of BGP messages, they were able to identify anomalous updates for which the path terminated in “4134 23724 23724”. It was noted that ASes such as AS9002 (RETN-AS ReTN.net Autonomous System), AS7018 (ATT-INTERNET4-AT&T WorldNet Services), AS3356 (LEVEL3 Level 3 Communications), and AS3320 (DTAG Deutsche Telecom AG) picked up these false announcements and propagated them further to their customers and peers. They also studied the geographic distribution of the hijacked prefixes and found that the majority of hijacked prefixes belonged to organizations in the US and China.

Using both control- and data-plane data, Renesys confirmed the geographic distribution of hijacked prefixes observed by BGPMon. Using traceroute, Renesys was also able to show that network traffic was able to pass into China Telecom’s network and back out to the intended destination. Further analysis was performed by Arbor Networks⁶ which focused on understanding how much traffic was diverted to China Telecom using traffic flows observed through ASes participating in the ATLAS project⁷. They did not observe a significant increase in traffic entering AS 4134 on the day of the incident.

In contrast to the analyses presented in various blog entries, our focus is on analyzing the incident using only publicly available data to understand what can be learned using current public data and what types of data should be collected in the future.

3.3 Methodology

To characterize the events that took place on April 8, 2010, we use a combination of publicly available control- and data-plane measurements. In this

---

⁴A. Toonk, BGPMon blog: Chinese ISP hijacks the Internet, <http://bgpmon.net/blog/?p=282>, Apr. 2010

⁵J. Cowie, Renesys blog: China’s 18-Minute Mystery, <http://www.renesys.com/blog/2010/11/chinas-18-minute-mystery.shtml>, Nov. 2010

⁶C. Labovitz, Arbor blog: China Hijacks 15% of Internet Traffic?, <http://ddos.arbornetworks.com/2010/11/china-hijacks-15-of-internet-traffic/>, Nov. 2010

⁷Arbor Networks, <http://atlas.arbor.net>

Table 3.1: Summary of control-plane updates matching the attack signature.

Monitor (Location)	Number of Updates	Number of Unique Prefixes
LINX (London, England)	60,221	11,413
DIXIE (Tokyo, Japan)	80,175	15,773
ISC PAIX (Palo Alto, CA)	123,723	35,957
Route-views2 (Eugene, OR)	216,196	29,998
Route-views4 (Eugene, OR)	49,290	18,624
Equinix (Ashburn, VA)	44,793	13,250
BGPMon list	-	37,213
Total	574,398	43,357

section, we provide background on China Telecom’s network and the data sets used in our analysis.

3.3.1 China Telecom’s network

China Telecom is the 11th largest ISP on the Internet and maintains multiple ASes partitioning their resources into different geographic regions (*e.g.*, provinces) and types (*e.g.*, data centers vs. regional networks). Many of these ASes are found as customers of AS 4134 in the AS-graph [96] and can be further identified using `whois` data. Indeed, the erroneous BGP updates originate from AS 23724, which is actually an AS owned by China Telecom and is located in Beijing.

3.3.2 Control-plane measurements

We use a combination of BGP updates⁸ and topology data [96] to characterize the China Telecom incident.

BGP updates. We use Routeviews monitors as a source of BGP updates from around the time of the attack. We consider updates with the path attribute ending in “4134 23724 23724” to be associated with the incident⁹ Table 3.1 summarizes the updates and prefixes matching this signature from the Routeviews monitors.

Topology data. We use the Cyclops AS-graph from April 8, 2010 [96] to infer the set of neighbors of China Telecom and their associated business relationships. Knowing the neighbors of China Telecom is particularly important when identifying ASes that potentially forwarded traffic to (and out of) China Telecom during the incident.

Figure 3.1: Interception observed in the traceroute from planet2.pittsburgh.intel-research.net to 125.246.217.1 (DACOM-PUBNETPLUS, KR).

3.3.3 Data-plane measurements

We use data-plane measurements from the iPlane project [32] and extract traceroutes transiting China Telecom’s network on April 8, 2010. We first map each IP in the traceroute to the AS originating the closest covering prefix at the time of the traceroute. If we observe a traceroute AS-path that does not contain China Telecom (AS 4134 or AS 23724) on April 7 or 9, that *does* contain these networks on April 8, we conclude that this traceroute was impacted by the incident. Furthermore, if we observe a traceroute that was impacted, and the final AS in the path is not AS 4134 or 23724, we conclude that this traceroute was intercepted. Figure 3.1 shows a traceroute where interception was observed. This traceroute only transits AS 4134 (China Telecom) on April 8 and is able to reach the destination through AS 2914 (NTT) who provides transit for AS 9318 (Hanaro Telecom). In total, we observed 1,575 traceroutes transiting China Telecom on April 8. Of these, 1,124 were impacted by the routing incident and 479 were potentially intercepted, with 357 of these receiving a successful response from the target.

3.3.4 Limitations

We face limitations in existing data sets as we reuse them for the unintended task of analyzing a large-scale routing anomaly.

Inaccuracies in the AS-graph. AS-graphs suffer from inaccuracies inferring AS-relationships (*e.g.*, because of Internet eXchange Points (IXPs) [97]) and poor visibility into peering links [98]. These inaccuracies impact our analysis of interception, which uses the AS-graph to infer China Telecom’s existing path to a destination. We discuss this limitation in more detail in Section 3.5.3.

Inaccurate IP to AS mappings. We note that our mapping of IP

⁸University of Oregon, Route views project, <http://www.routeviews.org/>, Aug. 2012

⁹All but 36 prefixes originated by AS 23724 match this signature.

Figure 3.2: Top 10 countries impacted by the China Telecom incident.

addresses to ASes may be impacted by IXPs or sibling ASes managed by the same institution [99]. Since our primary concern is paths that enter China Telecom *only* on April 8 the impact of siblings (*e.g.*, per-province ASes managed by China Telecom) should be mitigated. This is because paths to China Telecom’s siblings would normally transit China Telecom’s backbone AS 4134.

3.4 Impact of the China Telecom hijack

We now consider the impact of the China Telecom incident in terms of the prefixes that were announced.

3.4.1 What is the geographic distribution of the announced prefixes?

Figure 3.2 shows a breakdown of the prefixes that were hijacked by country, with and without excluding prefixes owned by AS 4134. We see the bulk of prefixes are registered to organizations in the US and China, followed by Korea, Australia and Mexico, which is consistent with observations made by BGPMon.

Was it random? Figure 3.2 also plots the geographic distribution of all routable prefixes on the Internet. Here we can see a disproportionate number of Chinese prefixes (especially belonging to AS 4134) being hijacked. Additionally, when comparing hijacked prefixes to the global distribution of prefixes there appears to be little evidence for attack. Indeed, the US shows fewer prefixes being hijacked than would be expected based on the global distribution, while countries in the Asia-Pacific region (*e.g.*, China, Korea, Australia) have more hijacked prefixes.

Table 3.2: Organizations most impacted by the China Telecom incident.

Rank	Overall		Subprefix Hijacks	
	Prefixes	Organization	Prefixes	Organization
1	9,296	China Telecom (AS 4134)	8,614	China Telecom (AS 4134)
2	1,573	Time Warner (AS 4323)	371	China Educ/Research (AS 4538)
3	1,229	Korea Telecom (AS 4766)	11	China Telecom (AS 38283)
4	739	AT&T (AS 7018)	9	Telecom Holding (AS 34590)
5	569	Tata (AS 4755)	4	Cisco Systems (AS 109)

3.4.2 Which organizations were most impacted?

Table 3.2 considers the organizations most impacted by the China Telecom incident; both overall, and when only considering subprefix hijacks. Organizations with the most prefixes announced tend to be peers of AS 4134. Indeed, direct neighbors of China Telecom are most adversely impacted with an average of 85 prefixes hijacked vs. nine prefixes hijacked for all impacted ASes.

Critical networks were subject to hijacking. While they do not make the top five list, China Telecom announced some critical US prefixes. Government agencies such as the Department of Defense, the United States Patent and Trademark Office, and the Department of Transport were impacted. In addition, commercial entities such as Apple, Cisco, DE Shaw, HP, Symantec and Yahoo! were impacted. This fact was also noted in Blumenthal *et al.*'s report made to the US Congress [94].

3.4.3 Were any of the announcements subprefix hijacks?

We now consider the length of the prefixes announced by China Telecom relative to existing routes. If the event was simply a leak of routes contained in the routing table, it should be the case that China Telecom's prefixes would be the same length as existing routes. Additionally, prefix lengths can shed light on the impact of the incident since more specific prefixes are preferred. For each of the six Routeviews monitors (Section 3.3.2), we use the RIB tables as seen on April 7 to derive the existing prefix lengths. Route aggregation means that the prefix length observed varies between the vantage points.

Subprefix hijacking was extremely rare. In total, 21% (9,082) of the prefixes were longer than existing prefixes at all six monitors. However, 95% (8,614) of these prefixes belong to China Telecom (Table 3.2). Most of the observed subprefix hijacking is due to poor visibility of Chinese networks (AS 4134, 4538, and 38283) at the monitors. Excluding these networks, we observe < 1% (86) prefixes being subprefix hijacked. The lack of subprefix hijacks supports the conclusion that the incident was caused by a routing table leak.

Figure 3.3: Example topology that allows for interception of traffic following decisions made by the neighbors of China Telecom (AS 4134). First, AT&T (AS 7018) chooses to route to China Telecom. Second, Level 3 (AS 3356) chooses *not* to route to China Telecom.

Our analysis highlights the importance of using multiple vantage points to mitigate the impact of route aggregation on results. Indeed, an additional 236 prefixes were subprefix hijacked at one or more vantage points.

3.5 The mechanics of interception

The fact that traffic was able to flow through China Telecom’s network and onto the destination is highly unusual. We now discuss how interception may occur accidentally, based on routing policies employed by networks.

3.5.1 How was interception possible?

Two key decisions, when combined with an inconsistent state within the network responsible for originating the false announcements, allows for traffic to be intercepted. First, a neighboring network routes traffic for the hijacked prefix to the network making false announcements. Second, another neighbor does not choose to do so. These properties have also been discussed in related work [58].

We illustrate them with an example from the China Telecom incident (Figure 3.3). This figure was derived using a combination of BGP updates (from the Routeviews project) and a traceroute observed during the incident¹⁰.

Decision 1: AT&T (AS 7018) chooses to route to China Telecom. In Figure 3.3, AT&T (AS 7018) has two available paths to the prefix. However, since the path advertised by China Telecom (AS 4134) is shorter, AT&T (AS 7018) chooses to route to China Telecom.

Decision 2: Level 3 (AS 3356) chooses *not* to route to China Telecom. In order for traffic to leave China Telecom’s network and flow on to the intended destination, China Telecom requires a neighbor that *does not* choose the path it advertises. In the example above, this occurs when Level 3 (AS 3356) chooses to route through its customer Verizon (AS

¹⁰J. Cowie, Renesys blog: China’s 18-Minute Mystery, <http://www.renesys.com/blog/2010/11/chinas-18-minute-mystery.shtml>, Nov. 2010

Figure 3.4: Cumulative distribution function of prefixes each neighboring AS routed towards China Telecom.

Table 3.3: Neighbors that routed the most prefixes to China Telecom.

Rank	# of Prefixes	% of Prefixes	Organization
1	32,599	75%	Australian Acad./Res. Net. (AARNet) (AS 7575)
2	19,171	44%	Hurricane Electric (AS 6939)
3	14,101	33%	NTT (AS 2914)
4	14,025	32%	National LambdaRail (AS 11164)
5	13,970	32%	Deutsche Telekom (AS 3320)

6167) instead of through its peer China Telecom (AS 4134). Thus, China Telecom can send traffic towards Level 3 and have it arrive at the intended destination.

We next characterize what causes these decisions to be made using a combination of control- and data-plane data.

3.5.2 How many ISPs chose to route to China Telecom?

We first consider how many ISPs made **Decision 1**. We observe 44 ASes routing traffic towards China Telecom, with each AS selecting the path through China Telecom for an average of 4,342 prefixes. Figure 3.4 shows the cumulative density function of the number of prefixes each AS routes to China Telecom. The distribution of prefixes each AS routes to China Telecom is highly skewed, with some ASes being significantly more impacted than others. The top five ASes are summarized in Table 3.3, which highlights the role of geography in the hijack, with networks operating in Europe and Asia-Pacific regions being most impacted. Academic networks (AARNet and National LambdaRail) were also heavily impacted.

We consider the paths these five networks have to the victim prefixes and find that 98% have a path length of three or more using the standard routing policy model [100] (96% if we the assume shortest path routing).

Table 3.4: Organizations with the most potentially intercepted prefixes.

Rank	Prefixes	Organization
1	712	AT&T (AS 7018)
2	174	Eli Lilly (AS 4249)
3	136	China Telecom (AS 4134)
4	120	Sprint (AS 1239)
5	108	Level 3 (AS 3356)

This means that the three-hop path (“4134 23724 23724”) announced by China Telecom would have been shorter than their existing path.

3.5.3 Which prefixes were intercepted?

We develop a methodology to locate potentially intercepted prefixes using control-plane data. Control-plane data has the advantage that it may be passively collected in a scalable manner. We validate our technique and further analyze the interception that occurred using data-plane measurements [32] (Section 3.5.4).

We use the following methodology to locate potentially intercepted prefixes using only control-plane data. First, for each hijacked prefix, we use the Cyclops AS-graph (discussed in Section 3.3.2) and a standard model of routing policies [100], to compute China Telecom’s best path to the prefix. Since China Telecom does not normally transit traffic for the hijacked prefixes, we were unable to extract the paths normally used by China Telecom from Routeviews. Next, for each of these paths, we check whether the next-hop on China Telecom’s best path to the destination was observed routing to China Telecom for the given prefix.

We observe 68% of the hijacked prefixes potentially being intercepted; however, 85% of these prefixes are observed being intercepted via AS 9304, a customer of China Telecom, which may be an artifact of poor visibility of the Routeviews monitors. Excluding paths through AS 9304, we observe a total of 10% (4,430) of prefixes potentially being intercepted. Table 3.4 summarizes the organizations with the most intercepted prefixes. In the case of AT&T, Sprint and Level 3; these ASes are providers and peers to China Telecom that still provided China Telecom (a peer) with paths to their prefixes. Additionally, some Department of Defense prefixes may have been intercepted as China Telecom potentially still had a path through Verizon.

Limitations. Our method is limited in two key ways. First, we may not observe all announcements made by China Telecom’s neighbors (*i.e.*, we may incorrectly infer that they are *not* routing to China Telecom because their announcement is not seen by the Routeviews monitors). Second, we do not know which neighbor China Telecom would normally use to transit traffic for a given prefix and thus we have to infer it based on topology measurements and a routing policy model.

Table 3.5: Why networks chose not to route to China Telecom.

Reason	# of traceroutes	% of traceroutes
Had a customer path	139	39%
Had a shorter path	193	54%
Had an equally good path	18	5%
Other	7	2%

Validation. Without ground-truth data it is difficult to quantify the inaccuracies of our methodology as we may both over- or under-estimate the potential for interception. We use the data-plane measurements described in Section 3.3.3 to validate our methodology. Of the 479 traceroutes that were intercepted, 319 (66%) were observed in prefixes detected as intercepted using our criteria. This inaccuracy stems from a lack of control-plane data which leads to the limitations mentioned above. With more complete data, our method could better identify potential interceptions.

3.5.4 Why *didn't* neighboring ASes route to China Telecom?

We use data-plane measurements to understand why neighboring ASes chose not to route to China Telecom (**Decision 2**). In Figures 3.3 and 3.1, the reason that the neighboring AS does not route to China Telecom is because they have a path through a customer to the destination. However, this is only one reason an AS would choose not to route to China Telecom. We consider the cases of interception observed in the iPlane data and determine why the neighboring AS did not route to China Telecom using the Gao-Rexford routing policy model [100].

Table 3.5 summarizes the reasons neighbors of China Telecom did not route to China Telecom. Here we only consider the 357 traceroutes where interception was observed and a response was received from the target. The majority of neighbors handling intercepted traffic did not choose the China Telecom route because it was longer than their existing route for the prefix in question.

Providers inadvertently allowed interception of customer traffic. A significant fraction (39%) of neighbor ASes do not route to China Telecom because they have a path to the destination via a customer, such as AS 3356 in Figure 3.3. These providers inadvertently aided in the interception of their customer's traffic by forwarding China Telecom's traffic to the destination. While providers cannot control the traffic sent to them by neighboring ASes, it may be beneficial to monitor the neighbors sending traffic towards their customers for anomalies, so that customers may be alerted to potential interception events.

We observe seven traceroutes where it is unclear why the China Telecom path was not chosen. These traceroutes involved a provider to China Tele-

com that chose to route towards other customers, most likely as the result of traffic engineering or static routes being used for the customer ASes.

3.6 Conclusion

Using publicly available data sources, we have characterized the China Telecom incident that occurred in April 2010. Our study sheds light on properties of the prefixes announced, and supports the conclusion that the incident was a leak of random prefixes in the routing table, but does not rule out malicious intent.

3.6.1 Key observations

The geographic distribution of announced prefixes does not support targeted hijacking. The distribution of announced prefixes is similar to the geographic distribution of all globally routable prefixes with a tendency towards prefixes in the Asia-Pacific region.

The prefixes announced match existing routable prefixes. We observe that $> 99\%$ of the announced prefixes match those existing at the Routeviews monitors. This supports the conclusion that the announced prefixes were a subset of AS 23724's routing table.

Providers inadvertently aided in the interception of their customers' traffic. Many networks that routed traffic from China Telecom to the correct destination did so because the destination was reachable via a customer path that was preferred over the path through China Telecom (a peer).

3.6.2 Discussion

On diagnosing routing incidents. Our work highlights the challenge of understanding large-scale routing incidents from a purely technical perspective. While empirical analysis can provide evidence to support or refute hypotheses about root causes, it cannot prove the intent behind the incident. However, empirical analysis can provide a starting point for discussions about the incident.

On the available data. It is evident from study of the China Telecom incident how the media took up incorrect information and propagated it to the masses.. For example, it was originally reported¹¹ that 15% of Internet traffic was routed through China Telecom, rather than that 15% of the routable prefixes were hijacked, which may be closer to the truth [94]. When the results of analysis can lead to a real-world response it is important that the data used for analysis should be as complete as possible and that the

¹¹S. Magnuson, Cyber Experts Have Proof That China Has Hijacked U.S.-Based Internet Traffic, <http://www.nationaldefensemagazine.org/blog/Lists/Posts/Post.aspx?ID=249>, Nov. 2010

robustness/limitation of results are clearly stated. These two properties can be achieved by increasing the number of BGP monitors [101] and performing careful analysis of robustness and limitations [102].

On collaboration. Our study clearly shows the intricacies of routing, and the way that networks are dependent upon each other for the correct functioning of the Internet. Even if an individual network implements “perfect” filtering techniques that sanitize all route announcements from its neighbors, it does not guarantee that its prefixes will not be victim to prefix hijacks and interception attacks. This is exemplified in Figure 3.3. Here, AS 22394’s actions do not lead to interception of its traffic in any way. Instead, different decisions by ASes 7018 and 3356 are what allow the traffic interception. AS 22394 can not counter the outcomes of decisions made by these ASes with current mitigation techniques. This illustrates the importance of collaboration among networks.

On scalability. Finding the instances in which interception occurred was like finding a needle in a haystack. After analyzing millions of globally collected traceroutes, we were able to confirm 479 cases of interceptions, and only then because we already knew the source of the false prefix announcements and signature. It will be even more challenging to effectively detect ongoing interception attacks. Given the very large volume of route announcements and potential data paths, any future systems to aid in the detection of routing anomalies must be scalable.

Chapter 4

PrefiSec: A Distributed Framework for Collaborative Security Information Sharing

4.1 Introduction

Today, organizations and network owners must protect themselves against a wide range of Internet-based attacks. The Border Gateway Protocol (BGP) is susceptible to prefix hijacks, sub-prefix hijacks, and interception attacks [11]. Networks and the machines within these networks may be scanned, probed, or spammed with unwanted traffic/mail [64, 66, 88]. In addition, network owners must be aware that machines within their networks may be compromised, participate in botnet activities, DDoS attacks, act as spam servers, or cause harm in other ways.

Unfortunately, Internet miscreants are becoming increasingly sophisticated and security attacks are no longer isolated events. Instead, attacks often cover multiple domains and behaviors. While some of these attacks are difficult for a single network entity to detect, collaboration among network entities provides richer information, and can help detect and prevent such attacks [16, 88]. This was further illustrated by our case study of the China Telecom incident in Chapter 3.

With the number of cyber-attacks expected to rise and an urgent need for strengthened network security [103], it is important to design systems that help responsible network entities and organizations collaborate in the battle against miscreants. In this chapter, we present the design of a system that:

- provides scalable and effective sharing of network information;
- provides incentives for organizations to keep their security footprints

clean, keep their reports honest, and share actionable information; and

- helps detect and protect against a wide range of attacks such as routing anomalies, spamming, and scanning.

Sharing of information about prefix ownership and malicious traffic emanating from prefixes can also help target a wide range of attacks.

While collaboration among organizations has been proposed, and the value of such collaboration demonstrated (e.g., [14, 16, 104]), it remains an open problem to design distributed mechanisms that provide effective decentralized collaboration among disparate organizations and Autonomous Systems (AS) that often belong to different administrative domains. Our focus is how such a system can be made scalable and dynamic, and how it can best improve the security of the members and the network as a whole.

In this chapter, we present PrefiSec [105], a scalable distributed framework that allows organizations and network entities to form alliances in the fight against miscreants. PrefiSec enables collaborative evaluation of network activities and the prefixes associated with each network. PrefiSec includes a distributed reporting system that allows participating members to effectively share observations, report suspicious activities, and retrieve information that others have reported. PrefiSec is a novel IP-prefix-based solution based on the popular Distributed Hash Table (DHT) Chord [34], a popular distributed hash table (DHT). Our solution is distributed, takes into account the hierarchical nature of the IP space, implements functionalities such as longest-prefix matching, and allows us to effectively store and retrieve information about organizations and their IP prefixes, as well as evaluate them with regards to a range of different attacks and miscreant behavior. The structure also efficiently combines information from both control-plane and data-plane based monitoring techniques.

Focusing on organizations (rather than individual machines) has many benefits. First, the network owners are in a good position to police miscreant activity within their own networks; *e.g.*, they can ensure that compromised machines within their own networks do not cause prolonged harm to others [14]. Second, it allows the system to scale effectively as state information can be maintained on a per-IP-prefix basis, each spanning a block of IP addresses. When the networks as a whole are evaluated, it can act as an incentive for network operators to keep their security footprints clean, keep their reporting honest, and share actionable information.

It should also be noted that it is not uncommon for the same malicious hosts to alternate between various kinds of malicious behaviors (e.g., to execute DDoS, scanning and spamming attacks) [88, 104]. Having a combined repository of different types of miscreant behavior has the advantage that it can help flag suspicious activity across services, and can significantly help to improve early detection rates of future attacks of other types. The improved security that the alliances provide also acts as a participation incentive.

We show that our distributed solution for information sharing and re-

porting is scalable, comes at a low communication overhead, and that the alliance framework is effective in improving the overall security. In Chapter 5, we discuss how policies can be designed to monitor routing-related anomalies using PrefiSec. Many of these advantages come through sharing of information that the organizations can use to protect themselves, as well as information that they can use to clean up their own networks.

The remainder of the chapter is organized as follows. Section 4.2 provides an overview of our system design. Section 4.3 presents our novel overlay solution. We discuss the services offered by such a system in Section 4.4, and Section 4.5 concludes the chapter.

4.2 System Overview

Our design is centered on responsible organizations and network owners sharing actionable information that helps protect their network, as well as keeping their own security footprint clean and their reporting honest. PrefiSec relies on organizations reporting attacks, suspicious behavior, and other more general status information associated with different IP-prefixes, IP-addresses, and/or ASes.

The PrefiSec framework is an application layer service that leverages sharing of network activity observed by routers, network monitors, and other infrastructure. While our design allows both edge networks and ASes to join the alliance, for simplicity of presentation, we assume that a network is an AS with multiple prefixes. Like ASes, edge networks can have multiple prefixes. To map to our AS-focused presentation, edge networks are mapped under a single AS, making them responsible for a fraction of the AS's prefix space. Larger organizations that operate under multiple ASes can simply be considered as multiple members.

Figure 4.1 provides an overview of the PrefiSec architecture. Here, AS1-AS3 operate separate nodes in the PrefiSec overlay network. We assume that trusted personal relationships among network operators are used to create the overlay network. (A multi-tiered extension is also discussed.) PrefiSec is designed to effectively share and manage any information about ASes and their prefixes. As an example, we present mechanisms and policies designed to effectively detect and/or raise alerts about potential interdomain routing attacks. Relying primarily on reports about origin AS and AS-PATH announcements, we assume that each participating AS collects (e.g., [106, 107]) and shares selected BGP updates from its edge routers, for example.

4.2.1 Distributed overlay framework

At the center of our design is a distributed overlay framework that allows participating members to effectively share information about organizations, networks, and their IP prefixes (blocks of IP addresses). This includes information about individual hosts that have been performing malicious ac-

Figure 4.1: High-level PrefiSec architecture.

tivities such as spamming, scanning, and denial of service attacks. Here, the activities of individual machines are mapped to the organizations or ASes to which they belong according to the longest prefix matching, for example.

Scalable overlay structures: To keep track of the activity associated with each organization and its IP prefixes, we maintain two complementary distributed structures.

- **Prefix registry:** We design a novel Chord-based [34] DHT, which stores prefix origin information (e.g., prefix-to-AS mappings) and observations of edge-network miscreant activities (e.g., scanning, spamming, etc.). The registry keeps track of the prefix hierarchy, and uses a distributed longest prefix matching algorithm for efficient insertion/retrieval.
- **AS registry:** A second Chord-based DHT is used to store information about ASes, their relationships, and AS-to-prefix mappings.

Figure 4.2 provides an overview of our PrefiSec framework, and shows how the two registry structures are linked by the prefix-to-AS and AS-to-prefix mappings (pointers in figure). Here, a participating member operates a node in the distributed AS registry and one node in the distributed prefix registry (e.g., the large circle and large rectangle, respectively, in the bottom-half of the figure) according to a set of built-in policies and locally stored/retrieved information (as shown in the upper-half of the figure). Reports and queries with shared information and observations are used to populate the registries. Similar to many other DHT-based overlays, the overlays in PrefiSec are self-organizing with provable query look-up times. Incremental deployment is easily achieved by adding/removing nodes to/from these structures, as members join/leave the alliance.

Figure 4.2: Overview of the framework, its key components, and structure.

4.2.2 PrefiSec services

PrefiSec includes some basic services, which can be used to construct high level services that can help detect various attacks on the Internet. In this section we describe basic services enabled by PrefiSec. The basic services provided by PrefiSec are: IP-to-prefix mapping, AS-to-prefix mapping, prefix-to-AS mapping, and AS-to-AS relationship information. The complementary AS and prefix registry enables providing prefix-to-AS and AS-to-prefix mapping. IP-to-prefix mapping is enabled by our distributed longest-prefix discovery procedure. More details about these basic services are presented in Section 4.4. Higher level services to detect routing anomalies and sources of edge-based network attacks are presented in Chapter 5.

4.2.3 Distributed information sharing and aggregation

Members share information about prefixes and ASes using *reports* directed to dedicated *holder* nodes. These nodes are responsible for aggregating the information associated with each prefix and create smaller summary reports that can be shared with the alliance members. To ensure that there is no single point of failure and that no single node is responsible for the entire evaluation of a prefix, multiple holder nodes per prefix are used. CryptoPAN [108] is used to find additional holder nodes for each prefix, helping

us preserve the efficient longest-prefix matching properties of our prefix-aware DHT.

Finally, we note that the holder nodes can help the other alliance members (i) by answering direct queries, and (ii) by creating and forwarding aggregated summary reports about AS and prefix. Similar to publish-subscribe systems, members can subscribe to aggregated summary reports created by the holder nodes, which can aggregate the information from many reporters. We expect that responsible organizations subscribe to their own prefixes and AS-registry information.

4.2.4 Local repository and optimizations

Additional reductions of the total Chord lookup overhead are possible either by reducing the number of lookups executed or reducing the internal routing overhead associated with each individual lookup. First, each node maintains a local repository with information about the prefixes that it sees, records statistics for these, and then informs the appropriate holder nodes. The system operates according to a soft-state protocol, with a time-to-live-based cache of prefixes, and with entries being updated when changes are detected. A node that has out-of-date prefix information can easily find the new up-to-date prefix information using our Chord-based DHT. Nodes that do not know the longest prefix for a given IP address can easily and quickly update their local prefix tables using the global repository. Second, in the case where additional storage overhead is acceptable, the current prefix space allows existing 1-hop routing optimizations [109, 110] to reduce each lookup to a single-hop lookup. Section 4.3 provide additional details and an overhead analysis of such optimizations.

4.3 Overlay Structures

In the following we present PrefiSec’s Chord-based overlay structure. For our AS registry structure, we use Chord more or less “out of the box”. For simplicity, we pick a circular identifier space that is large enough to uniquely specify any AS (based on its AS number, for example). For the prefix registry on the other hand, basic Chord is not adequate and must be modified to allow for efficient IP to prefix lookup. Both structures are related by the AS registry’s AS-to-prefix mappings and the prefix registry’s prefix-to-AS mappings.

4.3.1 Distributed prefix registry

An important aspect of our framework is that we maintain information about IP prefixes. Unfortunately, Chord’s (flat) circular identifier space does not naturally capture the hierarchical relationships between prefixes of different lengths. Ideally, we would like prefixes of any length to be uniquely assigned

to holders, and, given an IP address, the structure should return the holder of the longest-matching prefix. For example, for address 123.123.123.23, prefix 123.123.123.0/24 should be given priority over prefix 123.123.0.0/16. In this section we describe how we modify Chord to achieve unique and consistent longest-prefix-based assignment and lookup, while preserving the basic properties of Chord (outlined in Section 2.6).

Holder assignment: When dealing with prefixes, natural choices for holder nodes may include the node that maps to (i) the center of the prefix p , (ii) the first address of the prefix, or (iii) the last address of the prefix. While all three choices may appear natural candidates at a first glance, the last choice provides much more efficient lookups than the others. This is the only one of the three choices that ensures that the next candidate holder for a prefix of length $k - 1$ is ahead of the holder of (or the same node as) the prefix of length k . Given a circular identifier space with clockwise direction, coupled with the fact that longest-prefix matching typically involves first searching for the candidate holder of a longer prefix, this property can hence significantly reduce the number of messages that must be forwarded. Based on this insight, our system defines the holder of a prefix as the node responsible for the last IP address in the prefix block.

Furthermore, with this selection of a holder node, in the majority of cases, the next candidate holder for a prefix of length $k - 1$ is the same as the holder of the candidate prefix of length k , and in the other cases, the next node is located in a region of the identity space for which the node has many shortcut pointers. For example, the holder node *always* remains the same for a prefix of length $k - 1$ as that of length k whenever the last significant bit in the prefix of length k is a 1. This occurs in 50% of the cases. This observation suggests that the total number of nodes forwarding the query is on average less than $O(\frac{m}{2} \log N)$ hops. While we leave a rigorous mathematical analysis of the hop counts for future work, the next subsection presents an empirical evaluation.

Figure 4.3 shows the assignment of holders for the prefixes. For prefix 0000/3, the last address is 0001. The first node present in the clockwise direction from this address on the identifier circle has id 0010. Thus, node 0010 is the holder node for prefix 0000/3. Applying similar arguments, prefixes 0000/1 and 1000/3 are assigned to nodes 0111 and 1100, respectively.

Longest-prefix discovery: Nodes can query the global registry to update their local repository about the global IP-to-prefix mappings. These queries are resolved using a two-level greedy routing approach. At a high level, we first identify the potential holder of the longest possible prefix (of length k , for example) that would be responsible for that IP address. If this candidate holder is not the holder of such a prefix, it relays the request to the candidate holder of the next longest potential prefix (or length $k - 1$), and so forth, until a holder is found. Since /24 prefixes are typically considered the most specific prefixes allowed by modern BGP routers, as announcements for longer prefix are ignored [11], we use $k = 24$ as our initial choice of k .

Figure 4.3: Holder assignment in prefix overlay.

The routing algorithm can be formalized as shown in Algorithm 1.

Algorithm 1 Chord-based longest-prefix matching.

```

 $k \leftarrow 24$  // Init prefix length  $k$ 
repeat
   $p \leftarrow \text{truncate}(IP, k)$  // Next candidate prefix  $p$ 
   $t \leftarrow \text{lastIP}(p)$  // Last IP  $t$  in range of prefix  $p$ 
   $h \leftarrow \text{Chord}(t)$  // Find holder node  $h$  using Chord
  if  $\text{isHolder}(h, p)$  then
    return  $\langle h, p \rangle$  // The prefix  $p$  and holder  $h$  are found
  end if
   $k \leftarrow k - 1$  // Reduce the prefix length
until  $k == 0$ 

```

Example: Continuing with our previous toy example with a total identifier space of $2^4 = 16$ and only four nodes: 0010, 0100, 0111, and 1100, Figure 4.4 shows the high-level messages when node 1100 queries for the longest-prefix match for address 0011. In this case, node 1100 first uses Chord routing (based on finger table entries) to route the query to the last address in the range 0011/4 (which is 0011). When node 0100 receives this query, it checks that it would have been responsible for both 0011/4 and 0010/3, but has no entries for either. It instead determines that the next biggest range is 0000/2 and uses Chord to route to the last address in this range, in this case address 0111. While node 0111 does not have an entry for 0000/2 it is in fact the holder of range 0000/1. This will thus be the longest prefix that matches the original query, and at this time the query is resolved.

Figure 4.4: Resolving query and longest-prefix query routing.

4.3.2 Replication and load balancing

Replication and load balancing is important for achieving reliability in distributed systems. To ensure seamless recovery from abrupt node departures, Chord typically replicates the information stored at a node at its successor. In addition, we also replicate the information across multiple holder nodes.

In PrefiSec, we use CryptoPAN [108] to map prefixes to independent holders. CryptoPAN is a prefix preserving IP address anonymization scheme, which is commonly used for anonymizing IP addresses in network traces. With CryptoPAN, IP addresses are mapped one-to-one in a manner such that two IP addresses that belong to the same $/k$ -subnet also are part of the same $/k$ -subnet in the new address space. In this paper, we leverage this key property of CryptoPAN for a different but novel purpose. More specifically, we apply CryptoPAN to the original IP prefix (or address) such as to obtain a H new keys (IP prefix).¹ The use of CryptoPAN helps us ensure that the hierarchical features leveraged by our prefix register are preserved. By using a hash-function and replication, load balancing is also provided for free.

In general, nodes should request evaluation reports from multiple holders. In the case of inconsistencies, the node detecting the inconsistency should inform the holders. While some inconsistencies may be temporary, others may need to be resolved. In this case, one of the holders (e.g., selected based on node IDs, and/or differences from the median consensus) summarizes the different holder reports and initiates a vote between the holders.

¹As cryptoPAN is a hash, we can either use H different global keys (iterative approach) or apply a global key multiple H times (recursive approach) to obtain H different keys.

4.3.3 Lookup optimizations

To handle membership churn, each node in chord typically maintains state information about more than the $O(\log N)$ nodes in their fingertables used for forwarding. In general, this overhead is small. In the case when some overhead in bookkeeping is acceptable, it is also possible to design a one-hop routing scheme [109,110], which effectively maintains information about membership changes (new nodes entering or old nodes leaving the network), and disperses such information to all the nodes in the overlay networks.

The proposal by Gupta *et al.* [109] achieves the goal of dispersing information in the overlay network in a timely manner by leveraging a well-defined system hierarchy that divides Chord's circular identifier space into k equal contiguous intervals called *slices*. Each slice has a *slice leader*, chosen dynamically as the node that is the successor of the mid-point of each slice. Furthermore, each slice is divided into equal-sized intervals called *units*. Each unit also has a *unit leader*, chosen as the successor of the mid-point of the unit identifier space. When a node detects a membership change, it generates an event notification message to its slice leader. The slice leader aggregates the notifications it receives from its own slices for a certain time period and then dispatches the information for other slice leaders, which then disperse the information to unit leaders and then onto to the other nodes.

While such schemes require each node to have a pointer to every alliance member, Gupta *et al.* [109] shows that the use of slice leaders allows timely, efficient, and scalable updating of the membership pointers and redistribution of responsibilities under node churn, even for membership sizes up to a few million members. With far fewer ASes than this, and typical core routers having entries for hundreds of thousands of prefixes, we foresee these optimizations to be feasible down to the granularity of ASes and the prefixes seen by core routers. While this work only considers one-hop optimizations, if more edge networks join the alliance, two-hop schemes are possible [109].

4.3.4 Overhead analysis

To evaluate the scalability of the routing overhead associated with longest prefix matching, we use a modified version of PlanetSim [35] to simulate the path the query message takes in alliances with varying numbers of alliance nodes. The global repository was populated with all public routable prefixes that were available from the Cyclops project² on Sept. 23, 2012, and node identifiers are assigned at random for each simulation.

Figure 4.5(a) shows the average number of end-to-end Chord lookups, for overlays with different numbers of alliance members. For each alliance size, we simulate the query path for one million random IP-address pairs and report the number of end-to-end Chord lookups and IP-level hops. From

²Cyclops project, <http://cyclops.cs.ucla.edu/>, Sept. 2012. (This list included roughly 0.55 million prefixes.)

Figure 4.5: Number of Chord lookups and IP-level messages to resolve a query.

Figure 4.5(a) we see that the average number of times Chord-level messages are sent in a 64 node overlay network is approximately 1.75. The average number of Chord-level messages increases to approximately 2.70 in an overlay network with 2,048 nodes. The average number of IP-level hops (without the use of 1-hop optimization) for different overlay network sizes is shown in Figure 4.5(b). We note that the average number of IP-level hops required in a 64 node network is approximately 6.20, but increases to approximately 14 for a network of 2,048 nodes. One can clearly see that our distributed routing mechanism is scalable even when increasing numbers of nodes are added to the alliance network.

We also include a best fit curve of the form $c \log^\alpha N$, where α is a scale parameter. As discussed above, we would expect the first metric to scale sub-logarithmically, and the second metric to scale somewhere between $O(\log N)$ and $O(\log^2 N)$. Our polynomial fittings suggest that the power α is roughly 0.67 and 1.29, respectively, for the two metrics, confirming our intuition. Note that the two metrics are identical for the case in which we use 1-hop optimization, and the IP-metrics for this case therefore are omitted.

Figure 4.6 shows the Cumulative Distribution Function (CDF) for the number of end-to-end Chord-level lookups and IP-level hops, for overlays with different numbers of alliance members. With 64 nodes in the overlay network, approximately 70% of the lookup queries were resolved using a single lookup. This number decreases slightly to 60% when the number of nodes in the overlay network increases to 2048 (Figure 4.6(a)). Similarly, slightly less than 80% of lookups resolve using less than ten IP level-hops with an overlay network of size 64. This number drops to approximately 65% when the number of nodes increases to 2048.

Finally, per-node storage overhead scales as $O(PH/N)$, and forwarding tables as $O(\log N)$ or $O(N)$, depending on whether 1-hop optimization is implemented. To put the per-node storage overhead into perspective, consider a scenario in which there are $P = 0.55\text{M}$ prefixes, $H = 5$ holder nodes, and $N = 100$ alliance members. In this case, each node must on average

Figure 4.6: Cumulative Distribution Function (CDF) of the number of Chord lookups and IP-level messages to resolve a query.

store 25K prefixes, substantially less than the number of prefixes stored on a typical core router.

4.4 Service implementation

4.4.1 Basic services

The following basic services are provided by PrefiSec:

- **IP mapping:** Given an IP address, this structure returns the prefix with longest length that covers that IP address. Our distributed structure also ensures that such mappings are consistent no matter which node in the overlay network initiates the query. We discuss details of this service in Section 4.3.
- **Prefix-to-AS mapping:** Given an IP prefix, an alliance member can retrieve the origin AS/ASes for this prefix using the prefix registry. To populate the registry, any alliance member that sees a new origin AS for a prefix reports the new origin to the prefix registry, which identifies the correct node that is responsible for that prefix.

To retrieve information about origin AS/ASes associated with a prefix, an alliance node queries the prefix overlay, which routes the message using the routing principle discussed in Section 4.3.1 to the holder node responsible for that prefix.

- **AS-to-prefix mapping:** Given an AS number, the set of prefixes allocated to that AS can be retrieved using the AS registry. The AS registry is populated and continuously updated by alliance members. When an alliance member detects a new origin AS for a prefix, the node responsible for information about that AS is informed. Queries regarding ASes (and their prefixes) are directed to the corresponding holder nodes.

- **AS-to-AS relationships:** Relationships such as provider-to-customer, peer-to-peer, sibling and neighboring ASes can be important sources of information while troubleshooting the anomalies in routing. The information about such relationships can be created and extracted from our AS registry. We will also show how other information, such as an AS being an (Internet eXchange Point) IXP, can also be created and extracted from our registry.

4.4.2 High level services

Our system relies on organizations monitoring network activities and reporting attacks, suspicious behavior, and other more general status information about organizations and their IP prefixes. In particular, our system provides services for three broad classes of attacks, each building upon some of the basic services provided by the framework. First, leveraging the bi-directional AS-to-prefix and prefix-to-AS mappings, as well as the longest-prefix mapping service provided by our overlay, we integrate distributed prefix-hijack and subprefix-hijack attack detection and reporting into our overlay. By distributing the processing across prefix and AS holder nodes, we can achieve the same security as centralized approaches (e.g., [22,23]) without the drawbacks of a single point of failure [11].

Second, building upon the services provided by the overlay (including mappings of AS-to-AS relationships of different types), we also design simple policies and mechanisms for collaborative interception detection techniques. While the (sub)prefix hijack detection is integrated into the structure, the interception detection mechanism is built on top of the overlay structure.

Third, we show how the prefix-based structure can effectively maintain information about a wide range of edge-network-based miscreant activities and attacks, including spamming, scanning, and participation in DDoS attacks. This approach shines light on the organizations and network owners themselves. Details about each of the services are presented in Chapter 5.

Placing the responsibility and focus on organizations and activities associated with their prefixes, rather than on individual machines, has many additional advantages. The network owners are in a good position to police miscreant activity within their own networks, to ensure that compromised machines within their own networks do not cause prolonged harm to others [14]. By helping each other identify miscreant behavior, organizations can help keep each other's networks secured and protected. As leveraged here, it allows the system to scale effectively since each organization is typically responsible for a few IP prefixes, each spanning a block of IP addresses. Keeping track of the activities associated with prefixes, rather than individual IP addresses, also has the advantage of having a better chance of capturing subnetwork-aware IP-spoofing [66] and the effects of dynamically allocated IP addresses from within a prefix. It has been found that correlated attacks are more easily detected based on aggregate behavior than by

using techniques that evaluate individual IP addresses [88].

Finally, it should be noted that the combined information collected from many organizations can significantly help detect correlated attacks across multiple domains and malicious behaviors [88]. These types of attacks are becoming increasingly common, and it is typically not possible to detect them with information that is only visible to a single domain, or with information pertaining to a single type of attack, illustrating the importance of large-scale collaboration between organizations. Our simple and scalable system design allows organizations to easily share such information within a single system.

4.4.3 Incentive-based hierarchy

Finally, it should be noted that it is possible to build additional services into the system. For example, to help ensure that nodes are honest in their reporting, it is possible to use a multi-tiered hierarchy together with the concept of community-based trust, in which nodes can earn additional service, when/if they are promoted to the top tier. For example, top-tier members could be provided with additional information; e.g., by allowing them to act as an *extra* holder node for their own prefixes. While generalizations to an arbitrary number of tiers is possible, for simplicity, we consider a system with two tiers: a more trusted tier and a less trusted tier. Nodes are promoted/demoted between these tiers based on voting, with more weight given to the set of nodes in the higher tier. While the mechanisms used to assign weights can affect the success of the system, a detailed evaluation of the dynamics are left for future work.

4.5 Conclusions

This chapter presents PrefiSec, a distributed framework that allows organizations and network entities to form distributed alliances in the fight against miscreants. The framework places the responsibility on organizations and network owners.

PrefiSec is a novel distributed IP-prefix-based solution, which maintains information about the activity associated with IP ranges and autonomous systems (AS), rather than treating each IP address individually. Our framework allows us to effectively combine control-plane and data-plane monitoring. Chapter 5 presents a case study of how such cross-monitoring can help identify attacks that are otherwise difficult to detect. Our case study concerns instances of interception attacks on the Internet.

Our solution extends Chord [34], leverages unique properties of CryptoPAN [108], and takes advantage of our new scalable protocols for efficient information sharing. The framework provides natural and scalable prefix-to-AS and AS-to-prefix mappings, and it effectively maintains AS relationship data and information about miscreant activities associated with networks

and their prefixes. More detailed policies leveraging these services to target different attacks are presented in Chapter 5.

Chapter 5

PrefiSec-based BGP Monitoring

5.1 Introduction

In Chapter 4 we presented PrefiSec [105], a scalable distributed framework to share information about prefixes and ASes. The framework allows members to effectively share route/prefix information, report suspicious activities, and retrieve information about organizations, networks, and their IP prefixes. Such functionalities are enabled by effective and scalable distributed basic services provided by the PrefiSec framework, such as IP-to-prefix mapping, prefix-to-AS mapping, AS-to-prefix mapping, and AS-to-AS mapping. The framework consists of complementary overlay structures: (i) a Prefix registry, and (ii) an AS registry. These two structures are linked by prefix-to-AS and AS-to-prefix mappings.

In Chapter 2 we discussed the need for combining control-plane (routing) data and data-plane (packet header and content) monitoring for detection of certain attacks. The PrefiSec framework presented in Chapter 4 enables this to be done effectively within a single framework. Overall, PrefiSec enables collaboration among network entities to target a wide range of attacks on the Internet. Within our solution framework, we present and evaluate simple policies that help participating entities improve their security. We consider policies that help protect against a number of BGP-related routing attacks, as well as efficient reporting of edge-network activities such as spamming, scanning, and botnet servers. In this chapter, we present the details of mechanisms and policies that are built upon the basic services offered by the framework. We present and discuss services for three broad classes of attacks.

1) *Alerts for prefix and subprefix hijacks:* BGP uses prefix announcements to determine the routing paths that will be taken by Internet Proto-

col (IP) packets. A prefix (or subprefix) hijack involves an AS announcing a prefix (subprefix) allocated to another AS without permission. Building on basic services provided by PrefiSec (Chapter 4), we design mechanisms for effective and distributed prefix-hijack and subprefix-hijack attack detection and reporting. In contrast to existing systems (e.g., PG-BGP [23] and PHAS [22]), which suffer from a single point of failure and for which processing at the centralized node can be extremely high [11], our solution distributes the processing across all participants and does not have a single point of failure, while achieving the same protection as prior systems.

(2) *Alerts for interception attacks:* Hijacked traffic is even more difficult to detect if the intercepted traffic is re-routed to the intended destination. As such interception attacks typically involve many ASes whose individual decisions can impact the success of the attacks [9], collaboration is important in defending against these BGP attacks. Leveraging our overlay and information that it maintains about different AS-to-AS, prefix-to-AS, and AS-to-prefix relationship information, we design simple policies and mechanisms for collaborative interception detection techniques. These techniques are low in overhead and can be executed independently by each alliance member.

(3) *Aggregated prefix-based monitoring:* Our system can also be used to effectively maintain information about a wide range of edge-network-based miscreant activities and attacks. Prefix-based monitoring and bookkeeping is a natural fit for many miscreant activities, including scanning, spamming, DDoS attacks, and botnet activity. For example, an organization can flag servers that do not appear to be legitimate mail servers (e.g., by comparing DNS records with the IP addresses of the servers from which it receives SMTP messages). Leveraging the scale of our prefix-based structure allows us to effectively aggregate (potentially sparse) information from many reporters, helping responsible organizations keep their network security footprint clean. With malicious hosts increasingly alternating between a wide range of malicious behaviors [88, 104], a combined repository of different types of miscreant behaviors also helps flag suspicious activity across services, and can significantly improve early detection rates [16].

While the improved security can act as a participation incentive in itself, differentiated service and other incentive services can be built on top of the framework to further incentivize participation and good behavior.

Evaluation and overhead analysis: We use public wide-area BGP-announcements, traceroutes, and simulations to estimate the overhead, scalability, and false alert rate of the system. Particular attention is given to the overhead associated with the information that organizations may exchange with each other to detect different types of attacks and how this information is most effectively distributed and processed.

Our analysis and experiments are encouraging. We show that our distributed solution for information sharing and reporting is scalable, incurs low communication overhead, and our proposed framework, PrefiSec, is ef-

fective in improving overall security. For example, our case-based study of the China Telecom incident (that occurred on April 8, 2010) shows that the system would have detected all hijacked prefixes, while maintaining relatively low communication overhead. In fact, during a regular day, the overhead and number of false alerts is even shown to decrease with the size of the alliance.

The remainder of the chapter is organized as follows. Section 5.2 describes our PrefiSec-based prefix and subprefix hijack detection policies. This section also includes a case-study of the China Telecom incident. Section 5.3 presents an interception attack detection policy that builds upon the services provided by PrefiSec. To measure overhead, we also present a case-study of using this policy on three globally distributed routers. Section 5.4 discusses aggregated prefix-based monitoring using PrefiSec. Finally, we complete the chapter with conclusions in Section 5.5.

5.2 Prefix and subprefix hijacks

A number of anomaly detection algorithms have been proposed to detect bogus route announcements, including PG-BGP [23] and PHAS [22]. With these approaches the processing of prefix ownership history for all prefixes is centralized. In contrast, with our system, responsibility for and processing of prefixes are distributed among holder nodes. These nodes act as information aggregators that maintain history for each prefix, improving the scale and accuracy of what is possible with the limited resources of a central approach. By distributing the responsibility across multiple holders, PrefiSec also avoids a single point of failure or trust.

5.2.1 Policy overview

In the following, we present how our framework can help detect prefix and subprefix hijack attacks.

Prefix hijack: We design a distributed prefix hijack detection policy based on PHAS [22]. As discussed in Chapter 4, participating organizations in PrefiSec operate one “node” in the AS registry and one node in the prefix registry. These structures keep track of AS-to-prefix and prefix-to-AS mappings. Furthermore, within the prefix registry, every routable prefix announced on the Internet is implicitly assigned a *holder* that performs information aggregation and evaluation for that prefix. The holder nodes are also responsible for detecting when there are changes in the origin AS for a prefix, as well as notifying the previous origin AS of the prefix when a new AS claims ownership of the prefix.

An overview of our policy is given in Figure 5.1. The prefix hijack detection mechanism is invoked at a node in the alliance network when it sees a new prefix p , a new origin for a prefix p , or the TTL for the prefix p expires (step 1). The node prepares a query with this information for the holder of

Figure 5.1: Prefix hijack detection mechanism

prefix p (step 2), and the query is routed to a prefix holder of the prefix p (step 3) over the overlay network. The message is routed using the routing process that we described in Chapter 4.

For each prefix p , the holder node tracks the ownership set $A_p(t)$ over some time window of duration t . If the holder sees a change in the origin set, the current owner(s) of the prefix are notified and the ownership set $A_p(t+\epsilon)$ is updated (step 4). The case when the prefix has not been observed before is treated as a case of subprefix hijack and the subprefix hijack detection routine is invoked at such times.

Subprefix hijack: When a prefix is observed for the first time, it could potentially be a subprefix hijack attack on an existing prefix. We call the affected prefix a superprefix, and note that it is important to determine which superprefix is being potentially targeted. At the time of such occurrence, our distributed policy finds the immediate superprefix of the announced subprefix and notifies the origin AS for the superprefix about the announcement. The origin AS for the superprefix is typically in the best position to determine if the announcement is part of a subprefix hijack attack, or whether the announcement is legitimate and authorized by the origin AS of the superprefix.

Figure 5.2 provides an overview of our subprefix hijack alert notification policy. The subprefix hijack policy is invoked by a holder node $h_{p'}$ when it receives a prefix query for prefix p' and does not have an entry for this prefix (step 1a). At this time, holder node $h_{p'}$ creates a superprefix query (step 3) and uses Chord (step 4) to send it to the next potential candidate, if needed. To find the next node to which to forward the query (step 3), the holder $h_{p'}$ reduces the prefix length, say k' , of prefix p' by 1. Say prefix p'' is the new prefix with prefix length k'' (step 3.1), the holder node then checks if it is the holder for the new prefix created (step 3.2).

When the query arrives at this holder node, it again invokes the subprefix hijack detection procedure (step 1b). The new holder node checks if it has

Figure 5.2: Sub-prefix hijack detection mechanism

records for the new prefix (step 2). If the queried holder node has a record for the new prefix, the holder node will send the response and quit the procedure (step 2a). However, if it is not the holder, a new query will be prepared (step 3) that will be routed over the overlay network, with the new prefix p'' as the key (step 4). The process continues recursively until the superprefix p'' for prefix p' is found. When such a superprefix is found, the holder node $h_{p''}$ reports owner set $A_{p''}(t - \epsilon)$ for prefix p'' about subprefix p' and the claimed origin set $A_{p'}(t + \epsilon)$.

5.2.2 Case-based overhead analysis

To evaluate the benefit of information sharing among alliance members, we present a short case study based on real announcements. Since we are interested in the overhead both when the networks are under attack and under normal circumstances, we analyze the announcements seen around the time of the China Telecom incident [9] that occurred on April 8, 2010, when China Telecom announced that it was the origin of approximately 50,000 prefixes originated by other ASes. More specifically, we use the routing tables and updates seen at all six servers participating in the Routeviews project during the first two weeks of April, 2010. We base our original ownership lists on the RIB data from April 1, 2010, and then use the updates observed during the period starting on April 1, 2010 as our warm-up period.

Tables 5.1 and 5.2 shows results for April 7 and 8, 2010, respectively. These days correspond to the day before and during the incident, respectively. Results are presented for increasingly large alliances, each obtained by incrementally adding servers to the alliance, in the following order: Routeviews 2, Linx, Paix, Dixie, Routeviews 4, and Equinix. Perhaps most interesting are the number of new prefixes seen in a day (“prefix not seen”) and

Table 5.1: Summary of route announcements statistics for a April 7, 2010.

Metrics	April 7, 2010 (day before incident)			
	1 node	2 nodes	4 nodes	6 nodes
Route announcements	9,179,307	16,000,217	18,912,643	23,206,562
Announced prefixes	323,899	327,400	328,630	328,826
Prefix not seen	1,949	1,367	1,349	1,354
Prefix not seen (unique AS)	290	241	242	247
Owner not seen	209	130	134	138
Owner not seen in last 72h	1,302	3,594	3,874	2,735
Owner not seen in last 24h	3,200	5,840	6,184	4,746

Table 5.2: Summary of route announcements statistics for a April 8, 2010.

Metrics	April 8, 2010 (day of incident)			
	1 node	2 nodes	4 nodes	6 nodes
Route announcements	10,177,068	19,371,838	23,388,253	31,265,557
Announced prefixes	331,348	332,922	333,825	336,526
Prefix not seen	10,554	10,346	10,332	10,330
Prefix not seen (unique AS)	273	250	261	263
Owner not seen	21,275	21,001	29,704	30,245
Owner not seen in last 72h	21,570	21,970	31,108	31,680
Owner not seen in last 24h	22,561	23,027	32,382	33,937

the number of prefixes with a new owners (“owner not seen”). In these cases, the subprefix and prefix procedures would need to be invoked, respectively.

Normal conditions: The traffic overhead is very small compared to that of PHAS and other techniques that would use central processing. For example, on April 7, 2010, PHAS would have required all 23 million announcements to be forwarded to and processed on a single node (totaling 867MB compressed or 3GB uncompressed data, if using all six monitors). The load scales proportionally with more members. In contrast, with PrefixSec, a particular node, say RouteViews 2, would make 209 prefix queries (due to prefixes with a new origin: “origin not seen”) and 1,949 subprefix queries (due to new prefixes seen: “prefix not seen”) to the overlay.

Furthermore, with six alliance members, out of all queries generated by individual nodes, 138 and 1,354 queries would eventually result in prefix and subprefix alerts, respectively. These results show that the number of alerts processed by holders scales very nicely with the alliance size. In fact, with the corresponding sub-prefix policy invocations being distributed across holder nodes, we note that the alerts generated per holder node decrease even more (faster than $1/N$). This additional reduction is achieved by distributed information aggregation at holders.

Figure 5.3 characterizes the overhead of such prefix insertions, as mea-

Figure 5.3: Cumulative Distribution Function (CDF) of the distance to the closest prefix in the global prefix registry, of newly observed prefixes.

sured by the distance (in prefix lengths) between the two holders for prefix p (to be inserted) and the longest-matching prefix p' for which prefix p is a subprefix. We use three reference baselines: the RIB of the server itself, the combined RIBs of four different servers, and the global Cyclops database. We note that prefix length differences can be substantial, but decrease with larger alliance sizes.

Referring to Table 5.1, we can also see that the number of updates to the registry if using a (small) 24-hour window is much greater than if also taking into account the RIB information one week earlier (as per the much smaller values for the “origin not seen” statistics). Of course, using an adaptive window approach may lead to additional improvements [58].

Day of incident: Our overlay allows effective collaborative detection of prefix and subprefix attacks. In fact, referring to Table 5.2, during the day of the incident the alliance would raise 40,675 alarms, including alarms for all 39,094 unique prefixes that had the specific signature associated with China Telecom [9].

As expected, Figure 5.4 shows that there is a significant increase in traffic overhead on the day of the incident (April 8, 2010), but that the reporting overhead quickly decreases after the incident. We note that our architecture would easily handle such an increase. First, only the holder nodes would need to communicate with the owners of the hijacked prefixes. Second, using a registry of ASes (and the prefix they claim) allows the holder nodes to easily sanity check the claims right away. For example, China Telecom would quickly have been flagged and additional care could have been taken until authenticity could be confirmed or a certain time had elapsed. Finally, the number of alerts to be sent can be reduced if we aggregate messages to the same AS. In fact, many of the claimed prefixes are associated with the same ASes. This can be seen by the smaller “Prefix not seen (unique ASes)” statistics, relative to the “Prefixes not seen” statistics. For these statistics, the super-prefix (found using Cyclops data) was mapped to an AS using the

Figure 5.4: Time-line of anomalous origin reporting.

Figure 5.5: Impact of alliance size.

RIPE whois database.

Present day (April 2014): Figure 5.5 presents per-day statistics from the eleven (11) monitors that remained active throughout April 2014, including the subset used for the 2010 data. While we observe large day-to-day variations during the month (logarithmic y-axis), it is encouraging that the average node generates on average less than 2K queries per day (not shown) and the number of subprefix hijack alerts (Figure 5.5(a)) and prefix hijack alerts (Figure 5.5(b)) scales very nicely with the number of members. For example, the average number of alerts changes from 1,153 to 3,246, and from 125 to 327 for the two types, respectively, as the number of members increases from one to eleven.

Keeping in mind that the storage overhead and number of prefixes (Section 4.3.4) that each holder node is responsible for decreases in inverse proportion to the number of alliance members, we note that the queries processed per alliance node remains roughly constant, as this directly cancels the linear increase in the number of original queries generated by the entire alliance. In fact, with a sub-linear increase in the number of alerts, it can be argued that the overhead per node decreases with growing alliance sizes.

Figure 5.6: Detecting route inconsistencies and potential interception attacks.

5.3 Interception attack

As discussed in Chapter 2, one of the harder problems with BGP security is the detection of interception attacks [11, 58, 111]. Figure 5.6 shows an example. Here, AS *B* announces that it is one hop away from *C*, although in reality, it might not be connected to *C*. This announcement will not result in any prefix origin triggers, as the prefix origin is not being announced by *B*, but may still allow AS *B* to intercept traffic on its way to *C*.

Interception attacks have been studied and characterized [9, 58]; however, as of today there is no straightforward way of automatically detecting these attacks. Instead, network owners must typically manually analyze and resolve suspicious inconsistencies between announced BGP AS-PATHS and the actual data paths. In this section, we show how our overlay can be used to reduce the number of suspicious path inconsistencies.

5.3.1 Policy overview

We envision that PrefiSec members should maintain history of the announced AS-PATHs they see for prefixes, and evaluate any new paths and prefixes for inconsistencies. The overview of steps involved in our interception attack detection mechanism are shown in Figure 5.7. The interception attack detection mechanism is invoked when new $\langle \text{AS-PATH}, \text{prefix} \rangle$ pairs are detected (step 1) at a node in the alliance network. At such times, the node initiates the steps to collect additional information and evidence that can be used to evaluate the potential interception attack. First, the node performs a traceroute to an IP address with the prefix (step 2). In the next step it processes the traceroute that the node has performed. It first uses the prefix registry to perform IP-to-AS mapping for every IP address observed in the traceroute (step 3.1). The node then creates a traceroute AS path from the information obtained from the mapping (step 3.2) and compares the traceroute AS with the announced AS-PATH (step 3.3). If the AS path (data-plane information) of the traceroute does not match the announced

Figure 5.7: Interception detection mechanism

AS-PATH (control-plane information), we use AS relationships information to find the legit reasons for the difference in the paths (step 3.4). If no legit reason is found, the node raises an alert and reports the appropriate AS holder and prefix holder nodes about it (step 4).

5.3.2 AS relationship information sharing

When monitoring the activities of different organizations, one of the most difficult tasks is to efficiently filter suspicious activities. To reduce the number of false alerts, it is important to keep track of legit reasons for suspicious path discrepancies between the announced AS paths and the actual paths taken by the data traffic. Figure 5.8 summarizes some common legit reasons for such differences [99]. We next describe how the AS registry in PrefiSec can be used to maintain information about such reasons.

IXP cases (Figure 5.8(a)): Extra hops that are not visible in the announced AS-PATHs but are in the traceroutes may be caused by Internet eXchange Points (IXPs) [99] (or by a prefix that has multiple origin ASes; a case discussed later in this section). Extending the approach by Mao et al. [99], nodes that detect an extra hop for AS X can report the AS before and after X to the holder of X . This node can then calculate the number of unique ASes appearing just before and after X , referred to as the fan-in and fan-out factor, respectively. If these factors are greater than some threshold, then the holder can classify X as an IXP.

MOAS cases (Figure 5.8(b)): In certain cases we may observe that AS X in the AS-PATH is replaced by AS Y in the traceroute AS path. Such a replacement is common when the prefix is originated by multiple ASes (MOAS). We note that such MOAS cases are an artifact of mapping from an IP address to an AS and not the result of a routing anomaly and can be identified by either the holders in the prefix registry or the holders of the

Figure 5.8: Different legit reasons for anomalies in traceroute AS path and routepath mismatching.

AS registry. Holders in the prefix registry could be informed about multiple co-owners, as described for IXPs, which could inform the AS holders about these relationships. Alternatively, the AS holders themselves can keep track of replacements reported by member nodes.

Loop cases (Figure 5.8(c)): Some traceroute paths exits and enters an AS more than once [99]. For example, an announced AS-PATH $\{A, B, C, D\}$, may have a corresponding traceroute path $\{A, B, C, B, C, B, C, D\}$. These cases do not require any additional information from the AS registry.

Missing-hop cases (Figure 5.8(d)): Occasionally an AS hop seen in the AS-PATH is not observed in the traceroute AS path. For example, in Figure 5.8(d), we see that AS Y is missing in the traceroute path. This could occur for various reasons, such as routers in Y not responding to

traceroute queries or using IP addresses from their neighbors. This case does not typically require any additional information from the AS registry, although it would be easy to add information about the AS for which this has been observed (at the holder of Y , for example).

Alias cases (Figure 5.8(e)): Cases when an AS X in the AS-PATH is replaced by an AS Y in the traceroute AS path may be due to a router having IP addresses from two different ASes on its interfaces. Such an IP address is called an alias IP address and may arise due to third-party address issues [112]. The alliance nodes can use existing methods [112] to detect whether the extra AS hop appearing in the traceroute is a third-party address, and report its findings to the holder node of the replacement AS hop Y in the AS registry, which can apply a threshold-based policy on the number of occurrences required to be classified as an alias pairing between X and Y .

Sibling cases (Figure 5.8(f)): Other potential causes for valid discrepancies are route aggregation and sibling ASes, owned by the same organization [99]. Figure 5.8(f) illustrates a case in which the AS-PATH is $\{A, B, Y, C, D\}$ and ASes X and Y are sibling ASes. In the traceroute path we may observe Y being replaced by any of the following: $\{X, Y\}$, $\{Y, X\}$, or $\{X\}$. Similar to the case of IXP hops, we can extend an existing threshold-based policy that keeps track of the fan-in and fan-out factor for two-hop segments $\{X, Y\}$ in the traceroute AS path that corresponds to a single AS hop in the AS-PATH [99]. When an alliance node encounters such extra hops, it will report the AS-hop before and after the two-hop segment to the holder nodes of both X and Y . As with the IXP case, if the fan-in and fan-out exceeds a threshold, the holder node detects a sibling relationship, and can inform the other holder node about its finding.

5.3.3 Case-based analysis

We next consider how a participatory AS can use the information provided by our overlay structure to identify suspicious and non-suspicious path inconsistencies. For this evaluation, we use measurements from three public Routeviews monitors and three closely public traceroute servers, each pair hosted by Global Crossing (AS 3549), Telstra (AS 1221), and Hurricane Electric (AS 6939). These servers are located in Palo Alto (CA), Sydney (Australia), and San Jose/Livermore (CA), respectively.

Evidence collection: As with the prefix hijack detection overhead, great reductions in the number of traceroutes that must be executed can be achieved using a short history of previously seen AS-PATHS. For example, Table 5.3 shows the reduction in the number of traceroutes to execute, when considering multiple RIB snapshots (each two hours apart), at each of the three servers. We observe diminishing returns, with most of the the advantages obtained using a relatively small history. Based on an average 49.6% reduction, we chose to use a history of 24 hours in the following analysis.

Table 5.3: Reduction in the number of traceroutes.

Extra history (hours)	Server		
	Telstra	Global	Hurricane
2h	25.8%	25.1%	24.3%
4h	31.4%	32.6%	33.5%
8h	34.9%	38.2%	38.9%
16h	47.4%	45.6%	47.1%
24h	49.9%	48.6%	50.4%
48h	53.6%	51.9%	53.4%

Figure 5.9: New routes observed during the first week of November, 2012.

Figure 5.9 shows the number of traceroutes that would have to be executed, as a function of time. In this case, we use data from the first week of November 2012. We observe that all three servers experience similar variations in the number of traceroutes they need to perform at each instance during this period, although the pairwise Pearson correlation factors (Telstra-GlobalCrossing: 0.403; Telstra-HurricaneElectric: 0.604; HurricaneElectric-GlobalCrossing: 0.661) suggest that there is only moderate correlation in the number of traceroutes they need to perform at each instance.

Path comparison: For our analysis, we first convert the IP-level traceroutes to AS-level traceroutes. While our prefix registry is designed to provide this mapping, for the purpose of our evaluation, we used the Cymru whois database. As not all routers respond with ICMP messages at TTL timeouts, some IP addresses (and hence also AS) will be unknown. Consequently, a path may have one or more “*” between two interfaces that belong to same AS. Similar to Mao *et al.* [99], we assume that such “*” belong to the same

AS as surrounding hops. For example, we convert traceroute path “ $A * A$ ” to a single hop of A . It is also possible to observe “ $*$ ” between interfaces belonging to different ASes A and B ; *e.g.*, as in traceroute path “ $A * B$ ”. As our objective is to determine the ASes traversed by traceroute path, in this case, it does not matter if the unknown hops belong to A or B . In the traceroute path, we simply include AS A followed by B (but ignore where the path switched from A to B). We applied this methodology to create traceroute paths for the traceroutes that we performed from all the monitoring points that we used in this case study.

As mentioned in the previous section, there are many valid reasons why a traceroute path will not be a *direct match* to the announced AS-PATH.

- First, the IP-to-AS mapping could be incorrect or out-of-date. While our prefix registry will help here, there is no 100% up-to-date information source for such mapping.
- Second, it may not be possible to map all routers along the paths as some routers may have disabled ICMP, or traceroute servers (in this case out of our administration) may terminate at some timeout value. This will result in incomplete traceroutes. When traceroute paths are created from incomplete traceroutes, the path may match the subset of the observed AS-PATH. We refer to the set of queries that match only the initial subset of observed ASes in the AS-PATH as *subset matches*. This scenario was common for the Telstra servers.
- Third, as described in the previous section, our AS registry is used to keep track of six legit reasons for route anomalies and the corresponding AS relationship information. We call paths that match after applying each such condition (sequentially) *IXP matches*, *MOAS matches*, *loop matches*, *missing hop matches*, *aliases matches*, and *sibling matches*, respectively.

To estimate the conditions a large-scale system would see, we populated the AS registry using public data. We used a list of known IXP prefixes [113] to identify prefixes assigned to IXP points. We identify whether the observed anomalies are the result of MOAS cases by carefully mapping IP addresses to ASes, using the whois Cymru database. For aliases and sibling relationships we use a list of alias IP addresses from the iPlane project [32] and sibling relationship data from CAIDA¹, respectively.

- Finally, the announced AS-PATHs and traceroute paths may change over time, and may therefore not always be in sync. To identify such paths we use a 2-hour window of path announcements seen at the server in the past (and in the future). Cases of these two types, that otherwise matched, are referred to as *past matches* and *future*

---

¹<http://as-rank.caida.org/data/>

Table 5.4: Analysis of traceroute paths and AS-PATHs during week two (Nov. 1-7, 2012).

	Telstra	Global	Hurricane
Announcements	$3.6 \cdot 10^7$	$3.5 \cdot 10^7$	$3.6 \cdot 10^7$
Traceroutes (new route)	102,689	63,434	60,628
No data (new route)	506	60,045	3,200
Succesful traceroutes	102,183	3,389	60,627
Direct matches	12,387	704	16,374
Subset matches	62,952	947	13,267
IXP matches	2,672	11	30,071
MOAS matches	309	NA	445
Loop matches	2,271	108	1,021
Missing hop matches	2,730	689	6,886
Aliases matches	11,650	276	7,020
Sibling matches	1,764	27	243
Past matches	4,209	363	1,916
Future matches	487	23	515
Unresolved traceroutes	3,333	244	9,464
Unresolved triples	539	82	1,422

matches, respectively. The future matches apply to cases in which the announced path changes have not yet propagated all the way to the monitor server. If employed, this policy would of course require a time delay (e.g., two hours) before classifying a path.

Table 5.4 provides a breakdown of the traceroutes that we performed for the three servers, during the first week of November, 2012. In this case, we applied each rule in order, such that only traceroutes that did not match the previous criteria were considered for each new row. The cases for which we have *no data* corresponds to cases in which the public tracroute server did not respond. While the number of such queries to the Global Crossing server is high, we still include these measurements in our analysis as there should be no reason to expect any bias in the performed traceroutes (as limitations were due to server limitations in the number of traceroutes that we could perform).

Referring to the second to last line in Table 5.4, we note that the number of traceroutes that we could not automatically confirm (3,333; 244; and 9,464) is much smaller than the original number of successful traceroute queries (102,183; 3,389; and 60,627). We also see that IXP, aliases, MOAS, and sibling matches each are responsible for a significant portion of this reduction of unique candidate announcements, validating the importance of the AS registry.

Further reductions: To further reduce the candidate cases to consider, we identify unique triples [58]. Referring to Figure 5.10, such a triple consists of

Figure 5.10: ASes involved in the anomaly

Table 5.5: Redundancy in unique triples using different approaches.

	Server	Traceroute	Triples	Redundancy in triples		
				History	Sharing	Hybrid
Week 1	Telstra	66,985	372	—	12%	—
	Global	55,144	292	—	15%	—
	Hurricane	—	—	—	—	—
Week 2	Telstra	102,689	539	32%	14%	41%
	Global	63,434	82	27%	26%	38%
	Hurricane	60,628	1,422	4%	6%	7%
Week 3	Telstra	77,012	492	45%	16%	54%
	Global	—	—	—	—	—
	Hurricane	56,518	1,244	35%	6%	39%

1. AS *B* that is responsible for the anomaly,
2. AS *C* that *B* claims it will use, and
3. AS *X* over which it actually forwards the packet.

The last line in Table 5.4 shows the number of cases to consider after grouping the remaining triples. A reduction of between 65-85% is achieved in the number of cases to consider. Such groupings help reduce the number of cases to consider, as the reasons for these anomalies are often the same.

For the remaining candidate anomalies, nodes can contact the holders of AS *B* in the triple. If the holder node has seen this anomaly before and has found no problem, then it can respond that the anomaly is indeed benign. Otherwise, the holder node may need to do additional analysis. The holder nodes are in a great position to take advantage of aggregate information. Table 5.5 shows the summary statistics for three basic aggregation and history-based approaches:

- **History-based approach:** The history column shows the percentage of triples that have also been observed by the same server in past weeks.

- **Sharing approach:** The sharing column shows the percentage of triples that also are seen by at least one of the other servers (one or two, depending on week).
- **Hybrid-approach:** Finally, the hybrid column shows the percentage of triples that apply to at least one of the rules.

We see that with just three members the number of triples that must be processed can be reduced by up to 43% (week 3). Taking a closer look at Telstra alone, we note that 45% of the unique triples observed in week 3 are also observed in previous weeks, suggesting that significant reductions are possible by maintaining history. Similarly, 16% of anomalous triples observed by Telstra in week 3 are also observed by Hurricane Electric. If Hurricane Electric were to share information with Telstra about those anomalies (and had already resolved the anomalies), Telstra would not have to perform extra analysis in these cases. Finally, if Telstra was to combine both the history-based and the sharing-based policy, it would already have access to information for 54% of the anomalies that it discovers in week 3. Clearly, there are significant advantages to maintaining history at the holder nodes, and sharing information across organizations.

We note that none of our policies or mechanisms thus far has relied on any knowledge of AS-to-AS relationships. An AS also wanting to investigate the *cause* for the remaining unresolved anomalies may want to make use of AS-to-AS relationships such as customer-to-provider (C2P), peer-to-peer (P2P), and provider-to-customer (P2C) [9]. Unfortunately, most ASes do not want to share their peering policies. To incorporate a classification of AS-to-AS relationships into the AS registry, future work could therefore include the design and evaluation of a distributed version of the valley-free [99] classification method by Shavit *et al.* [114].

Overlay overhead analysis: Referring back to Figure 5.7 and our distributed interception detection method, we note that the overlay is involved in steps 3.1, 3.3, 3.4, and 4. Out of these steps, steps 3.3, 3.4, and 4 are not expected to generate significant communication overhead. For example, steps 3.3, 3.4, and 4 are concerned with resolving anomalies in a traceroute AS path when compared to AS-PATH. Our case study shows that the number of anomalous traceroutes are significantly lower compared to the number of traceroutes that we had to perform. In fact, the cases to consider are further reduced when the triples of AS (as shown in 5.10) involved in the anomaly are considered. For example, for the Telstra router the number of anomalous triples to resolve using the AS registry was 539, compared to the 102,689 traceroutes that were performed over a complete week. Of course, as shown in Table 5.5, the number of anomalies that are unresolved after considering further reductions in anomalies based on history, sharing, and hybrid policies implies that number of anomalies and ASes to be reported at the end of interception detection procedure (step 4) is significantly lower.

We now consider step 3.1 of the interception detection algorithm. In this

Table 5.6: Reduction in IP-to-AS mappings (Nov. 1-7, 2012).

Day	Total IP addresses	Unique IP addresses	Unique /24 prefixes (reduction)	IP Addresses not resolved in last 24 hours (reduction)	Unique /24 prefixes not seen in last 24 hours (reduction)
1	295,890	9,247	4,328 (53.2%)	–	–
2	437,736	13,149	6,054 (54.0%)	9,672 (26.4%)	5,325 (60.0%)
3	107,101	5,219	2,396 (54.1%)	2,640 (49.4%)	1,746 (66.5%)
4	141,527	5,667	2,539 (55.2%)	3,644 (35.7%)	2,104 (62.9%)
5	440,938	11,816	5,292 (55.2%)	8,947 (24.3%)	4,756 (59.7%)
6	185,285	7,291	3,360 (53.9%)	3,785 (48.1%)	2,460 (66.3%)
7	252,159	8,333	3,800 (54.4%)	5,274 (36.7%)	3,121 (62.5%)
Total	1,860,636	60,722	27,769	33,962	19,512

step, the IP addresses in the traceroutes are mapped to ASes that announces those IP addresses. Using our prefix registry with our longest prefix matching aware routing, appropriate IP-to-AS mappings can be created. However, without careful implementation the number of IP-to-AS mappings can easily explode. For example, Table 5.6 shows the number of IP addresses that need to be mapped to ASes is slightly over 1.8 million over the duration of our case study for the Telstra server (to resolve over 0.1 million traceroutes).

We next evaluate three simple approaches that help reduce the number of IP-to-AS lookups.

- IP-to-prefix mapping approach:** First, we keep track of mapping of an IP addresses to /24 prefixes. Suppose an IP address i_1 is mapped to /24 prefix p_1 whose AS number is AS_1 . If another IP address i_2 , which maps to the same /24 prefix p_1 , needs to be mapped to AS, then such an additional lookup can be avoided as the corresponding AS for this prefix is already known from the previous lookup. Thus, further reductions can be achieved by keeping track of previous IP addresses and their /24 prefix mapping. We used /24 prefixes as BGP routers on the Internet normally filter prefix announcements with prefix length greater than 24.
- History based approach:** As with AS-PATHS, a history of previous IP-to-AS lookups can help reduce the number of IP-to-AS lookups. Suppose an IP address i_1 is mapped to a /24 prefix p_1 whose AS number is AS_1 . If such an IP address is required to be mapped to AS within time period t_m , then the result from the previous IP-to-AS mapping can be used to resolve such a mapping. We present the results for a 24 hour history, as IP-to-AS mappings are not expected to change very often.

- **Combined IP-to-prefix and history based approach:** With the combined approach, we keep track of mapping of IP addresses to /24 prefixes and ASes and also maintain a history of such mappings for the last 24 hours.

Table 5.6 shows the number of IP-to-AS mappings required for the Telstra router when the three different lookup reductions techniques are applied. While the total number of IP addresses to be resolved is over 1.8 million, we can see that combining the above approaches would reduce the number of lookups to 19,512 (sixth column). In fact, just avoiding repeated IP addresses during the same day reduces the number of lookups to 60,722 (third column).

The fourth column in Table 5.6 shows that the number of IP-to-AS lookups required with *IP-to-prefix mapping approach* is 27,769 (fourth column), and with the *history-based approach* the total number of IP-to-AS mappings comes down to 33,962. While the biggest reductions are possible with the hybrid approach, all these approaches required a significantly smaller number of lookups compared to the approximately 1.8 million IP-to-AS mapping that otherwise would have been needed.

5.4 Prefix-based information sharing

In addition to routing-aware information, our prefix registry is designed to help organizations share anomaly alerts and information about a wide range of attacks associated with edge networks and their different prefixes. We use four basic examples to illustrate the power and generality of the framework.

Scanning attacks: Both vertical (port) scanning and horizontal (IP) scanning are common on the Internet. To detect scanning, organizations typically monitor incoming (and outgoing) traffic, using intrusion detection systems. For example, using connection establishment information, tools such as Bro [115] can classify each connection as either good or bad. With some hosts responsible for a mix of good and bad connections, Allman et al. [64] propose a set of heuristics to classify hosts based on this mix. We note that our system is well suited to implement generalizations of such classification policies, in which prefixes are evaluated rather than hosts [66].

At this point, it should also be noted that the holder nodes (due to a sparsely populated identifier space and the use of CryptoPAN) are responsible for a range of closeby prefixes. This is particularly attractive as it has been shown that blacklists from nearby prefixes are correlated [91]. For this reason, the holders may in some cases want to alert their predecessor and successor nodes.

Spam and botnet server activity: As discussed in Section 4.2, organizations can monitor SMTP traffic from non-legit servers and report such activity to the holder nodes. This information can then be used to inform the owner of the prefix of suspicious activity, and/or provide organizations

with information about the general trust level associated with different prefixes. Prefix-based evaluation is particularly valuable for organizations that observe activity from a new prefix with which they may not have had much experience. Finally, we note that this approach naturally extends to many other activities, including the presence of botnet servers.

Cross class anomaly detection: It is common for malicious hosts to alternate between different malicious behaviors [16]. With access to information related to multiple types of malicious activities (e.g., scanning, spamming, botnet activity, and BGP routing attacks) associated with a particular prefix, the holder node is in a very good position to analyze the behavior and activity associated with this network.

Additional services and collaboration: Finally, it should be noted that there are several complimentary services that can be offered to participating organizations and network owners, which could be a good incentive for networks to join the alliance. Such services could either be provided as part of the framework or built on top of the framework.

One such service could be an attack correlation detection service. Correlated attacks are prevalent on the Internet, are often mounted by a single source, and target multiple selected networks. As targeted networks are not random, these attacks often involve a common set of networks. The holder nodes are in a good position to help other nodes identify common sets of correlated networks. For example, the holder can inform the victim networks of other networks that have experienced similar attacks. These nodes can contact commonly targeted networks and form smaller collaborative groups, in which additional actionable information can be shared.

5.5 Conclusions

This chapter presents attack detection mechanisms that can be built upon PrefiSec. In particular, we present (sub)prefix hijack detection mechanisms that leverage PrefiSec’s distributed IP-to-AS and AS-to-IP registries. We also present an interception attack detection mechanism that combines the control-plane and data-plane data in a scalable manner to raise alerts for possible interceptions on the Internet. We evaluated the proposed mechanisms using public wide-area BGP-announcements, traceroutes, and simulations. We show that the system is scalable with limited overhead, and discuss the general applicability of our framework, by explaining how PrefiSec can easily be used by network operators to detect and monitor scanning and spamming activity associated with different machines and prefixes within their networks.

Chapter 6

TRAP: Open Decentralized Distributed Spam Filtering

6.1 Introduction

When trying to identify miscreant networks and prefixes, it is important to be able to evaluate the behavior associated with individual machines. However, given the sophistication of techniques employed by attackers it is often difficult for an individual host or network to evaluate individual machines. It is therefore necessary for specialized servers to collaborate with each other and share information to detect sources of attacks. In this chapter we present TRAP, a distributed collaborative reputation-based system to detect malicious hosts based on information sharing among mail servers that are under different administrative domains [28]. The reputation values calculated with this system can, of course, be aggregated later over prefixes and networks using the PrefiSec framework, presented in Chapters 4 and 5.

E-mail is one of the most widely used Internet applications. However, today most e-mail is unwanted — spam. The Messaging Anti-abuse Working Group estimates that in the second quarter of 2010, nearly 90% of all e-mail was spam.¹ Spam is more than just a nuisance. It consumes resources and time, and is a vehicle for malicious software. Many solutions to the spam problem have been developed. Although none of them actually stop the spam, many of them reduce the number of unwanted messages that reach their intended recipients. However, this is something of an arms race: as spam protection gets better, spammers get smarter, so we need to keep developing new techniques to deal with spam. In this chapter we present TRAP, a distributed reputation-based system that can help e-mail recipients determine if messages are spam or not. We envision TRAP being used with

---

¹Messaging Anti-Abuse Working Group, <http://www.maawg.org>, Nov. 2010

conventional detection tools to form a complete spam detection and filtering solution. TRAP could also be used to throttle spam at the source: ISPs and other networks could use data in the TRAP system to identify (potential) spam sources within their networks, and e.g. limit their ability to send e-mail. The information could also be shared and aggregated over prefixes using PrefiSec, to help hold networks more accountable for the activities within their networks, instead of only blocking individual machines.

TRAP differs from currently deployed reputation-based filtering systems in that it is completely decentralized and completely open. Decentralized means that all participants in the system are equal: there is no central point of control or data collection point, and no hierarchy among participants. This can make the system very robust, since there is no single point to compromise or attack. Open means that the inner workings of the system are completely exposed: users can find out exactly how messages are processed and why TRAP yields the results it does. This allows participants to understand why the system behaves the way it does, which is vital when troubleshooting e-mail systems. However, adversaries have the same insight, which may help in attempting to subvert the system.

This chapter presents the design of the TRAP system and an evaluation of the behavior of the reputation system. The reputation system in TRAP is distributed with a very low overhead. Through simulations we show that despite the simplistic protocol, the reputation system is stable under normal (and many abnormal) Internet conditions.

6.2 The TRAP system

TRAP is intended to operate in tandem with mail servers that can use external software for spam detection (e.g. sendmail's *milter*² or Postfix's content filtering³). TRAP consists of mail sending nodes (senders), mail receiving nodes (receivers), trust holders (holders), trust requesting nodes (requesters) and experience reporting nodes (reporters). Figure 6.1 shows an overview of how these interact.

Senders are Internet hosts that send e-mail to TRAP participants (i.e. MTAs). Senders do not have to participate in TRAP; they are simply the subject of trust calculations. Receivers are hosts that receive mail and use TRAP to detect spam. Receivers participate in TRAP by being or using requesters and reporters.

A holder is a TRAP participant that stores the trust values of one or more senders. Any node in TRAP is eligible to be a holder, and each sender is handled by multiple holders. Requesters are hosts that request trust values from holders. Reporters are hosts that report experiences to the holders in the TRAP system. Requesters and reporters are either mail receivers, or

---

²Sendmail, "An interactive catalog of sendmail mail filters", <http://www.milter.org>

³The Postfix project, <http://www.postfix.org>

Figure 6.1: Overview of components of TRAP.

are used by mail receivers.

When a receiver receives an e-mail from a sender, it either uses or acts as a requester to get the trust value for the sender. The requester contacts the holder(s) for the sender to collect reputation values. These are processed by the requester to create a single trust value, which is then reported to the receiver. The receiver performs spam detection using the trust value, and finally reports its experience to the holders(s) for the sender by using or acting as a reporter.

The spam detection step is not specified by TRAP; any normal detection methods may be employed, but they can use the trust value from TRAP to increase their precision. For example, a threshold-based scheme (e.g. SpamAssassin⁴) could adjust the detection threshold based on the trust value of the sender, or a system could apply stricter filtering or more detailed checks to mail from untrusted senders.

6.2.1 Requirements

TRAP was designed to meet specific criteria: it must not rely on a central authority or central services; its operations must be open, so users know what it does and can diagnose failures; at the same time, it must be secure, so spammers cannot easily subvert the system on their own; and it must be scalable and interoperate with existing systems. For the purpose of an open and distributed design, TRAP uses a Distributed Hash Table (DHT) with an open API. For the purpose of scalability and to reduce cheating, we also outline the following requirements:

- TRAP should support large numbers of users, but be useful even when used by a small portion of the Internet.
- Communications overhead should be kept to a minimum.
- Holders must present consistent trust values for a given sender. Otherwise it becomes impossible to interpret the trust values.

⁴SpamAssassin, “The Apache SpamAssassin Project”, <http://spamassassin.apache.org/>

- Holders must be assigned so it is impractical for a sender to choose a specific holder and to set up a holder that handles a specific sender.
- A sender should never be its own holder, and no node should report its own score.

6.2.2 The TRAP overlay network

TRAP uses the Pastry [68] peer-to-peer system for message routing and data lookup. Pastry provides a self-organizing distributed hash table with provable look-up times for queries; important properties for scalability. Each node has a 128 bit identifier (assignment of identifiers is discussed below). Note that only requesters, reporters and holders must be part of the Pastry network; senders do not need to participate.

6.2.3 The TRAP protocol

The TRAP protocol messages contain a message type and zero or more attributes. Sender and receiver addresses and payload length are available from Pastry. Attributes consist of type, flags, length and value. One flag is currently used: *copy* (the attribute must be copied from request to response). There are four message types: reputation request, reputation response, experience report, and error.

Reputation request is sent from receiver to holder to get the reputation of a sender. The only required attribute is *target* (the hashed node ID for which the score is to be returned). Reputation response is sent in reply to a reputation request. The attribute *reputation* contains the reputation of the requested *target*. Experience report is sent from reporter to holder to update the reputation value. Required attributes are *timestamp* (the timestamp at which the experience occurred), *message-id* (a 128 bit integer that identifies the experience; message IDs must be monotonically increasing for each sender/reporter pair), and *target* (hashed node ID that the experience relates to). Holders do not respond to experience report messages. An error code is sent when it is not possible to generate a response. The attribute *error-code* indicates the type of error. Nodes can match responses to requests in any way they want. We recommend the use of a custom attribute with the *copy* flag set.

The use of a DHT ensures low communication overhead. To further reduce the communication overhead, additional constraints can be applied on the exchange of experience reports and reputation requests. For example, the experience reports can be sent during specific time intervals or only when unexpected experiences are encountered. Similarly, reputation requests can be sent only when the receiver cannot determine if the sender is malicious or not.

6.2.4 Node identifiers

Referring back to the original requirements in Section 6.2.1, requirements four and five are achieved by the way node IDs are assigned. Specifically, in the TRAP system we assign node IDs by computing a cryptographic hash of the node's IP address; TRAP currently uses the lowest 128 bits of the SHA-256 function over the node's IP address: $ID^0(addr) = H(addr)$. This ensures that a node can choose its own ID only by searching the entire IP address space and gaining control of the part that contains the sought-after IP address, which is infeasible.

6.2.5 Trust holders

Each sender is associated with several holders to ensure trust values are not lost and to make subverting the system more difficult. This, however, increases communications.

TRAP implicitly assigns holders to every IP address. To find the holder H of a given node, its node ID is hashed and the result treated as a DHT key; the holder is the Pastry node that handles that key. Additional holders are assigned by repeatedly applying the hash function:

$$\begin{aligned} H_0^{ID} &= \text{hash}(ID), \text{ if } i = 0, \\ H_i^{ID} &= \text{hash}(H_{i-1}^{ID}), \text{ if } i > 0, \end{aligned}$$

where $\text{hash}()$ is the global hash function, ID is the node identifier of the node, and H_i^{ID} is the i^{th} holder node for the node with identifier ID .

6.2.6 Calculation of reputation values

The full details of the reputation model that we have used is presented in Duma *et al.* [116]. Below we present some details of this model that we have used in TRAP. Table 6.1 gives the description of the notations used in discussing the reputation model. $E_{a,b}$ is the set of experiences that peer a has had with peer b and $E_{-,b}$ represents all the experiences about peer b . The trust that peer a has with peer b depends on $E_{-,b}$.

TRAP uses a dynamic trust metric that is resilient to oscillatory behavior. It consists of a short-term trust factor, a long-term trust factor and a penalty factor that can be applied to either the short-term or the long-term trust factor [116], and has values from 0 to 1.

Short-term trust factor: The short-term factor reflects new experiences; it is defined as a reinforcement learning update rule for non-stationary problems, and can be tuned using a positive and a negative sensitivity factor: the positive factor determines how fast trust increases in response to positive experience and the negative factor determines how fast trust decreases in response to negative experiences [116]. Short-term trust (T^s) is calculated as:

Table 6.1: Description of notations used for reputation model

Notation	Description
T^s	Short term trust value
$T_n^s(a, b)$	Short term trust value that peer a has for peer b after n^{th} transaction of peer b
T^l	Long term trust value
$T_n^l(a, b)$	Long term trust values that peer a has for peer b after n^{th} transaction of peer b
$T^r(a, b)$	Recommendation trust: measure of how much trust peer a places in the experiences reported by peer b
α	Positive sensitivity: weighs the effect of recent positive experiences on short-term trust value
β	Negative sensitivity: weighs the effect of recent negative experiences on short-term trust value
$e_i(a, b)$	Experience that peer a had about peer b in i^{th} transaction of peer b
$w_s(n)$	Significance weight
M_n^c	Misuse accumulator
f^p	Penalty factor

$$T_{n+1}^s(a, b) = \begin{cases} T_n^s(a, b) + \alpha T^r(a, x)(e_{n+1}(x, b) - T_n^s(a, b)), \\ \quad \text{if } e_{n+1}(x, b) - T_n^s(a, b) \geq -\epsilon, \\ T_n^s(a, b) + \beta T^r(a, x)(e_{n+1}(x, b) - T_n^s(a, b)), \\ \quad \text{otherwise.} \end{cases} \quad (6.1)$$

Here, $T_{n+1}^s(a, b)$ is the short term trust value that peer a has for peer b after the $(n+1)^{th}$ transaction of peer b , $T^r(a, x)$ is the amount of trust that a places on the experiences reported by x , $e_{n+1}(x, b)$ is the experience that peer x had with peer b in the $(n+1)^{th}$ transaction of peer b , $T_n^s(a, b)$ is the short term trust value that peer a had for peer b after n^{th} transaction of peer b , ϵ specifies the tolerated margin of error due to possible noise in experience evaluation, and finally α and β are positive and negative sensitivity factors.

Long-term trust factor: The long-term factor reflects the long-term behavior of the subject, and is an average over all previous experiences, weighted by the number of available experiences. The long term trust, T^l , is calculated as:

$$T_{n+1}^l(a, b) = \frac{w_s(n+1)}{n+1} \left[\frac{n}{w_s(n)} T_n^l(a, b) + T^r(a, x) e_{n+1}(x, b) \right]. \quad (6.2)$$

Here $T_{n+1}^l(a, b)$ is long term trust value based on $(n+1)$ experience about peer b , n is total number of experiences, $T^r(a, x)$ is amount of trust that peer a places in the experiences reported by peer x , $e_{n+1}(x, b)$ is experience

reported by peer x in its transaction with peer b , $T_n^l(a, b)$ is long term trust that peer a has based on n experiences about peer b . The *significance weight* $w_s(n)$ depends on the total number of experiences that are available for a trust calculation. The significance weight helps to account for the fact that when few experiences are available, the average is less likely to reflect the true long term peer behavior. Requiring a minimum number of experiences before giving full weight to the average weighted trust values, w_s can be calculated as:

$$w_s(n) = \frac{n}{\max(n, n_{min})}. \quad (6.3)$$

Penalty factor: When a subject misuses its trust, the difference between trust invested and actual experience accumulates and is converted to a penalty factor that is applied to the short- or long-term trust factor, either permanently reducing trust or making it increasingly difficult to regain trust. The penalty factor $f^p(a, b)$ at a peer a for peer b after the n^{th} transaction of peer b can be calculated as

$$f_n^p(a, b) = \frac{M_n^c(a, b)}{c + M_n^c(a, b)}, \quad (6.4)$$

where M_n^c is the misuse accumulator after n transactions of peer b . The M^c is based on the difference between the trust invested and the actual experience and is calculated as:

$$M_{n+1}^c(a, b) = \begin{cases} M_n^c(a, b) + T^r(a, x)(t_n(a, b) - e_{n+1}(x, b)), & \text{if } t_n(a, b) - e_{n+1}(x, b) > \epsilon, \\ M_n^c(a, b), & \text{otherwise.} \end{cases} \quad (6.5)$$

Combined trust metric: The three trust factors short-term, long-term, and penalty factor can be combined into a single trust metric. To permanently reduce the long-term trust factor, the penalty factor can be combined with a long-term trust factor as follows:

$$T_{n+1}^l(a, b) = T_n^l(a, b)(1 - f^p(a, b)). \quad (6.6)$$

In this way, the long term factor will continuously decrease if the peer exhibits oscillatory malicious behavior. Similarly the penalty factor can be combined with the positive sensitivity factor as:

$$\alpha \leftarrow \alpha(1 - f^p(a, b)). \quad (6.7)$$

In this way, the positive sensitivity will be decreased every time a bad behavior is encountered. The positive sensitivity factor is used in combination

with the short-term trust factor. A lower value of the positive sensitivity factor implies that it will become increasingly difficult for a node to regain higher short-term trust value.

Finally, the trust in a subject is calculated as the minimum of the short-term and long-term factors; *i.e.*,

$$t_n(a, b) = \min(T_n^s(a, b), T_n^l(a, b)). \quad (6.8)$$

We have previously used this trust metric in an experimental peer-to-peer intrusion detection system [117]. In TRAP unrated nodes start with a trust value of 0.4, which the evaluation shows is a value that only appears during the transition between extremes, and therefore can only be interpreted as unknown trust.

Using this method, the order in which experience reports are processed will affect the reputation value. Therefore, it is important that all holders process the experience report messages for a given sender in the same order. Conventional methods for ensuring that transactions are processed in the same order at all holders, such as distributed agreement, have unacceptably high communications overhead (several roundtrips per node). Therefore, TRAP uses a method that attempts to process experience reports in order at all holders, but makes no guarantees. This method requires no messages other than the experience report itself.

In TRAP, every experience report message carries a timestamp and message ID. The experience report messages resulting from any given experience all have the same message ID and timestamp. We can define a total order for experience reports relating to any given sender based on timestamp, message ID and sender node ID. In particular, we say that the order of two reports R_i and R_j is $R_i < R_j$ if one of the following three conditions hold:

1. $\text{timestamp}_i < \text{timestamp}_j$
2. $\text{timestamp}_i = \text{timestamp}_j$ and $\text{sender}_i < \text{sender}_j$
3. $\text{timestamp}_i = \text{timestamp}_j$ and $\text{sender}_i = \text{sender}_j$ and $\text{messageId}_i < \text{messageId}_j$.

When a holder receives an experience report, it delays processing it until it is old enough that it is likely to have reached all holders. For this to work all TRAP nodes must have synchronized clocks; we specify that all nodes must use NTP [118] against a reliable reference to ensure that the clocks are not *too* far apart.

6.3 Security of TRAP

Since there is considerable economic incentive behind spam, TRAP must be resilient against attacks intended to disrupt its operations or manipulate

the reputation of a given node. Here we briefly discuss how TRAP handles several typical attacks.

False addresses: If a sender were able to use false addresses, TRAP would be ineffective. However, in practice, senders are limited to the addresses they are assigned as it is infeasible to spoof the source address of an SMTP connection reliably. Since any given sender is limited to a small number of IPv4 addresses, attacks based on false addresses are impractical. With IPv6 TRAP would need to use network prefixes (*e.g.*, the first 48 or 64 bits of the address) to construct the node ID in order to achieve the same effect.

Malicious or malfunctioning holders: The simplest attack on TRAP is for a holder to falsify trust values. Nodes can also oscillate between good and bad behavior, to build up and then abuse trust [119]. TRAP mitigates these attacks in two ways: the trust metric is resilient to oscillation [116], and because trust values converge to the extremes, false values are easily distinguished from real ones.

Collusion attacks: In this attack, one or more malicious nodes send false experience reports to the holder of a target node (either to improve or reduce its reputation) [120]. TRAP must tolerate collusion since membership is completely open. Prevention of collusion attacks is an important topic of future work on TRAP.

TRAP also needs to contend with generic attacks on peer-to-peer systems such as the Sybil and eclipse attacks [121, 122]. For the most part these are mitigated in the Pastry layer, but the implicit node ID assignment in TRAP helps by ensuring that an entity only receives a quantity of node IDs equal to the number of network addresses it controls (or in the case of IPv6, network prefixes).

6.4 Evaluation of the reputation system

In TRAP holders must report reasonably consistent reputation values: otherwise requesters cannot interpret the values. Due to the use of non-guaranteed delivery for the experience reports, some inconsistency between holders is inevitable, but if the normal degree of consistency is known, requesters can process the values, and even identify potentially malicious or malfunctioning nodes. We have conducted simulations to measure the consistency between holders in TRAP. Specifically, we examine:

- To what degree do trust values diverge over time if the maximum and minimum trust values are not constrained? This gives an idea of the stability of the system, exposing problems that might be hidden if values are constrained to a fixed range.

- To what degree do trust values diverge over time if the maximum and minimum trust values are constrained (as they must be in a reputation system)?
- How does message loss and the TRAP processing delay affect convergence?
- How does the choice of parameters for the trust metric affect operation?

6.4.1 Method

Running a simulation of TRAP consists of creating an overlay network, generating mail messages and experience reports, randomly dropping messages, and calculating and observing trust values at the holders.

The simulation is created using the discrete event simulator in FreePastry [68], an open-source Pastry implementation.⁵ We place nodes randomly in Euclidian space and the delay between a pair of nodes is based on how far apart they are. This topology does not accurately simulate the Internet, but is adequate for our purposes.

Mail messages are generated by a single sender and sent to random recipients. E-mail sender behavior is modeled as a stochastic process with a Poisson distribution (i.e., messages are sent with a known average rate, independently of each other, in this case 20 messages per second) [123].

Node churn could be significant in P2P networks. For example, Bhagwan *et al.* [124] note that over 20% of the hosts in the Overnet peer-to-peer file sharing system arrive and depart every day. Such node failures along with link losses can result in message losses [125]. Motivated by these studies, we drop the experience reports randomly in our simulations. Unless otherwise specified, experience reports were dropped with a arbitrary probability of 8%.

Trust values were periodically written to file so we could see how they changed over time. We applied the penalty factor to the long term trust value; since the sender exhibits oscillatory behavior, all trust values eventually converge to zero. We used a positive sensitivity value of 0.0010 and a negative sensitivity value of 0.0020.

6.4.2 Results

Divergence with unconstrained trust value: We simulated six separate TRAP networks and measured the maximum difference in trust values between any two holders over the course of the entire simulation. We found

---

⁵FreePastry, Pastry: A substrate for peer-to-peer applications, <http://www.freepastry.org/>

(a) Change in trust value for mail sender with lower spam rate.

(b) Maximum difference of trust values at trust holder nodes.

Figure 6.2: Convergence and maximum difference of trust values for lower spam rates.

that divergence tends to increase over time, but remained very small in relation to the magnitude of the trust value. This indicates that we can expect holders to report very similar scores when the values are constrained.

Convergence with constrained values: Figures 6.2 and 6.3 show how trust values for a single sender develop over time for low and high rates of spam, respectively, when constrained between one and zero. Figures 6.2(a) and 6.3(a) show that the values rapidly move between 0 and 1, and the Figures 6.2(b) and 6.3(b) show that holders deviate significantly only during the transition between these extremes. We also found that the maximum deviation decreased with the increasing spam rate, from 0.35 for 10% spam to 0.1 above 40% spam. This means that the meaning of trust values is easy to determine: near the extremes, the values are reliable and indicate whether a sender is a spam source or not; whereas other values indicate a transitional phase during which trust values cannot be reliably used. This also means that inaccurate values from holders can be detected quite easily. Note that the trust value for any given sender may move between the extremes any number of times due to changes in sender behavior.

Effect of drop rate on convergence speed: Since TRAP does not use a reliable transport, dropped messages can result in deviation between holders. We ran simulations to determine what effect message drop has on convergence speed and deviation between holders. We found that the max-

(a) Change in trust value for mail sender with higher spam rate.

(b) Maximum difference of trust values at trust holder nodes.

Figure 6.3: Convergence and maximum difference of trust values for higher spam rates.

imum difference between holders increased only slightly with the increased drop rate for a given spam rate, but that drop rates affected convergence speeds considerably, as shown in figures 6.4(a) and 6.4(b).

Choice of parameters: As expected, when the processing delay d (for trust reports at the holders) is lower than the latency in the network, trust values diverge rapidly. With d higher than the average latency, trust values converged. With d set so 95% of packets had a latency of d or lower, convergence was rapid and maximum difference in trust values small. We assume that the clock skew between nodes is small compared to d .

6.5 Related Work

Typically, spam detection involves extracting features from an email message and use these features to determine if the message is likely to be spam or not. Different policies can be used to flag potential spam messages. For example, filters that look for particular phrases and words can be used to identify likely spam messages. In the case multiple filters are available, policies can assign weights to each filter. There exists many filters including threshold-based filters and probabilistic filters that use a Bayesian networks trained

Figure 6.4: Number of messages processed before the trust values converge for varying message drop rates.

using a corpus of e-mail [126, 127].

Spam detection methods are typically applied at a single mail receiver; however there exist a number of collaborative systems, where the users and/or mail servers collaborate to improve detection quality. An early effort was Vipul’s Razor⁶, which allow users to nominate messages as spam. Messages with enough nominations are recorded as spam in a catalog server that users can query. Variants of this type of filtering are now used in several solutions, such as Cloudmark [128], the Distributed Checksum Clearinghouses (DCC)⁷, Pyzor⁸, and appear in many other systems [129–134]. It should be noted that these systems are designed to detect spam messages; not to identify spam sources. Furthermore, many of the Razor-like systems rely on a central server for their operation.

Trust and reputation frameworks have also been used to aid spam filter-

⁶V. V. Prakash, <http://razor.sourceforge.net/>, May. 2007

⁷<http://www.rhyolite.com/dcc>

⁸<http://sourceforge.net/apps/trac/pyzor/>

ing. Two main variants exist: user reputations and sender trust. With user reputations, the (apparent) identity of the sender is used in spam filtering; with sender trust, trust is associated with the sending server. MailRank [135] use a central server to assign trust values to email addresses. McGibney and Botovich [92] proposed a system in which each mail server stores a trust value for each other mail server it is aware of. Mail servers exchange trust values with each other, and the trust in a sender is a combination of direct experience and reputation. This system has no central control, but experiences are only reported to the local neighborhood. The Internet is estimated to have 2-4 million mail servers⁹. It is not clear whether trust values in McGibney and Botovich's system would percolate through a network of that scale fast enough for the system to react to changes in sender behavior.

Foukia *et al.* propose a collaborative system for defense against spam [136], in which federations of systems share trust and information about spam mail traffic volumes, so that all participants can calculate a global trust value and traffic rate for mail senders. If a high volume of spam mails is detected by a server, that server will notify all the collaborative servers. The collaborating servers will cooperatively restrict the quota on the number of emails they can accept from spam sending servers. The major issue with this approach is the timely response to newly compromised mail servers since by the time that the collaborators learn about spamming server, the server would have already sent large number of spam mails. Another issue is that nodes oscillating between good and bad behavior will still end up achieving their spam objective.

6.6 Discussion

In this chapter we have presented TRAP, a distributed reputation-based system for evaluating whether mail senders are sources of spam or not. We use an existing reputation system and evaluate it for consistency so that reputation values can be interpreted in a meaningful way. Unlike previous reputation systems for spam filtering, TRAP is completely decentralized and completely open: there is no reliance on a central service of any kind, and the internals of the system are open for anyone to examine. This creates new challenges as it makes it easier for spammers to subvert the system, but in return, the system can be made very robust and easy to join.

TRAP uses the Pastry protocol to implement the underlying peer-to-peer network, which gives proven performance for communication, and a dynamic trust metric that is resilient to oscillatory behavior. This makes it harder for spam sources to subvert the system. The results presented here show that the fundamental component of TRAP, the trust management system, is efficient and robust. We have presented initial findings concerning

⁹M. Still, Measuring the popularity of SMTP server implementations on the Internet, http://www.stillhq.com/research/smtpsurveys_feb2010.html, Feb. 2010

tuning the parameters of TRAP. However, a considerable amount of work still remains: the precise mechanics of joining and leaving the system are not defined; several security challenges remain; and the precise tuning of the various parameters in TRAP requires additional experimentation with the network.

TRAP is intended to be used as part of a spam solution; in itself it will be insufficient to determine what is and what is not spam, but we think that it can improve the accuracy of a complete system. TRAP can also be used as a complement to PrefiSec. In particular, the trust values of individual sources (IP addresses) calculated using TRAP can be mapped to networks and their prefixes using PrefiSec. Networks with many spam sources can then easily be identified. This may not only be useful for keeping network owners accountable and encouraging them to help address spam sources directly, it will also help identify networks that are more likely to harbour other types of miscreant behaviors.

Chapter 7

Crowd-based Detection of Routing Anomalies on the Internet

7.1 Introduction

Despite being one of the most important infrastructures in today's society, the Internet is highly susceptible to routing attacks that allow an attacker to attract traffic that was not intended to reach the attacker [11, 15]. In this chapter we present a crowd-based approach that lets users on the Internet collaborate with each other and raise alerts for potential routing anomalies.

Section 2.3 described how successful hijack attacks can result in black-holing, interception, and imposture attacks. While *black-holing attacks* (e.g., as used in some censorship attacks) are relatively easy to detect, as the traffic is terminated at the attacker and the traffic source may not obtain expected end-to-end responses, attacks where the attacker also impersonates the destination (*imposture attacks*) or re-routes the traffic to the destination (*interception attacks*) are much more difficult to detect. This chapter focuses on the detection of such stealthy interception and imposture attacks.

Although this is a globally important problem and several crypto-based efforts (described in Section 2.4) to secure Internet routing has been proposed [19–21], deployment have been hampered by high deployment costs and failure to incentivize network operators to deploy these solutions [11, 15]. Instead, operators and other organizations running their own Autonomous Systems (AS) are typically limited to monitoring path announcements made by other ASes and/or the data paths taken by the data packets themselves [23, 56]. However, these detection-based approaches are typically limited by the number of BGP vantage points contributing BGP updates, the limited view these provide, or the high overhead associated with continuous

monitoring using active measurement techniques such as traceroutes.

Recently, there have been increasing occurrences of routing attacks. Some of these attacks have been performed to gain access to sensitive or valuable information, or to circumvent constitutional and statutory safeguards against surveillance of a country's citizens, while others have been accidental or have tried to deny users access to certain services (for censorship purposes) [9, 15, 44, 45]. With many of these attacks affecting regular end users, routing path integrity is becoming increasingly important not only for operators, but also for end users.

As an effort to help detect routing anomalies and to push operators to implement secure mechanisms, we foresee a user-driven approach in which concerned citizens collaboratively detect and report potential traffic hijacks to operators and other organizations that can help enforce route security. To reduce the number of active measurements needed we argue that clients could use passive monitoring of their day-to-day network traffic to identify prefixes (range of IP addresses) that may be under attack.

In this chapter we present and analyze a user-centric crowd-based approach in which clients (i) passively monitor the round-trip times (RTT) associated with their users' day-to-day Internet usage, (ii) share information about potential anomalies in RTT measurements, and (iii) collaboratively help identify routing anomalies that may require further investigation. The use of passive measurements ensures zero measurement bandwidth overhead, helps distribute the monitoring responsibility of prefixes based on the users' interaction with services provided within each prefix, and ensures timely initial detection of potential routing anomalies as seen by the users themselves. The use of crowd-based sharing of potential anomalies allows the number of alerts that need further investigation to be reduced during normal circumstances and the attack detection rates to be increased during attacks.

Our general system design does not make any restrictions on how the passive measurements are performed. This could be done at the clients themselves or at proxies, for example. The RTT measurements could be calculated separately or by leveraging the average RTT estimates calculated as part of the congestion control of any TCP connection. Furthermore, the RTT estimates could be collected for all connections, a random subset of connections, or a targeted subset based on domains and prefixes on which the client places additional value, for example. Determining the best design choices for these aspects is beyond the scope of this thesis. Instead, we focus on how these measurements can be combined with collaborative information sharing to provide alerts about traffic anomalies that may require further investigation.

This work makes three main contributions. First, we use longitudinal RTT measurements from a wide range of locations to evaluate the anomaly detection tradeoffs associated with crowd-based detection approaches that use RTT information (Section 7.3). Considering two different types of

stealthy and hard-to-detect attacks (interception attacks and imposture attacks), we provide insights into the tradeoff between attack detection rates (when there is an attack) and false alert rates needing further investigation under normal circumstances (when there is no attack), and how this tradeoff depends on different system parameters such as system scale, participation, and the relative distances between detectors, attackers, and victims. Second, using a simple system model and trace-driven simulations, we evaluate the set of detector nodes that provides the best detection rates for a candidate victim (Section 7.4). Of particular interest is the relative weight that should be given to candidate detectors at different distances from the victim node. Finally, we provide a discussion and analysis of the overhead associated with different candidate architectures, including both central directory approaches and fully distributed solutions (Section 7.5). Regardless of the implementation approach, the system is shown to operate with relatively low overhead and simple heuristics can be used to identify sets of nodes that are well-suited for collaborative detection of individual prefixes.

The chapter is organized as follows. Section 7.2 provides background on routing attacks that are targeted in this chapter. Sections 7.3-7.5 present each of our three main contributions (outlined above). Finally, Section 7.6 presents related work, before Section 7.7 concludes the paper.

7.2 Targeted routing attacks

As discussed in Section 2.3, one of the biggest vulnerabilities with BGP is its inability to confirm the allocation of prefixes to corresponding ASes. This allows malicious entities to perform a *prefix hijack attack* in which the attacker announces one or more prefixes allocated to other networks. The effectiveness of this type of attack is typically determined by the route announcements made by the different ASes announcing the same prefix and the routing policies of the ASes that must select which of the alternative paths to use. Since forwarding routers always select more specific subprefixes, a more effective attack is a *subprefix hijack attack* in which the attacker announces one or more subprefixes of the prefix allocated to the victim network.

These attacks can be further classified by the actions taken by the attacker. In *black-holing attacks* the traffic is terminated at the attacker and the originator of the packet will not see a proper response. Although some investigation may be needed to determine the cause, in general, these attacks are relatively easy to detect. In contrast, it is more difficult to detect attacks in which the attacker relays the attracted traffic to the intended destination (allowing proper end-to-end communication), or where the attacker impersonates the intended destination. These two types of attacks are typically referred to as *interception attacks* and *imposture attacks*, respectively, and will be the focus of this chapter. For a more detailed discussion of these types of attacks we refer the interested reader to Section 2.3.

Many techniques have been proposed to protect against routing incidents

and attacks such as those described above. These techniques typically either use prefix filtering or cryptography-based techniques and rely for implementation and deployment by responsible organizations. However, deployment of these solutions has been slow and the solutions do not work well unless a large number of networks deploy them [11, 15].

Rather than relying on organizations or dedicated infrastructure, this paper considers an alternative approach in which we leverage passive measurements at the end hosts to detect suspicious RTT deviations caused by underlying attacks. This allows any concerned citizen to help detect anomalous routes, which then can be analyzed more closely using active measurements and complimentary control-plane information.

7.3 System model and detection tradeoffs

We consider a general system framework, which we call CrowdSec, in which stationary end-user clients passively monitor the RTTs to the prefixes (range of IP addresses) with which the client applications (e.g. a web browser) are interacting. Leveraging such measurements has many advantages. First, passive measurements does not add any measurement traffic to the network and hence does not affect the bandwidth share given to the user applications. Second, the high skew in service accesses [2] ensures repeated measurements (over time) to many prefixes.

Each CrowdSec client passively monitors the RTTs associated with their users' day-to-day Internet usage. Using Grubbs' test [137], each client individually identifies potential anomalies in the RTTs and shares information about these anomalies through the CrowdSec system, including statistics reflecting how unlikely such an RTT deviation (or more extreme) are, given past observations. Using binomial testing (or alternative tests) these shared statistics are then combined into refined statistical measures that reflect how unlikely the combination of observations would be under normal conditions, allowing for collaborative detection of routing anomalies.

For the Grubbs' test, similar to TCP's timeout threshold calculations, we do not transform the measurement values, but instead use the raw RTT values. We note that the non-symmetric tails in the RTT measurements will typically cause the RTT measurement not to be normally distributed, especially for short paths, but closer to log-normally distributed [138]. While log-normal distribution characteristics may suggest that the RTT measurements could potentially benefit from the transformations before applying the Grubbs' test, we note that even after taking the logarithm of the RTTs, the statistics from the Grubbs' test should only be considered as a relative measure of an anomaly, not as a measure of their absolute probabilities. Similarly, there may be hidden correlations that affect the absolute values calculated with the binomial tests. Again, we use these methods as tools to detect anomalies, rather than assign probabilities.

As RTT-based measures include less information than full traceroutes

Figure 7.1: Evaluation scenarios for example interception and imposture attacks.

and other active measurement techniques might provide, CrowdSec and other purely RTT-based approaches should not be used as primary evidence for routing attacks. Instead, further investigation is typically needed to distinguish temporary increases in RTTs due to bufferbloat and other temporal congestion events, for example, from routing anomalies. Yet, as active traceroute-driven approaches that capture the data path come at much higher overhead and can be forged by the attacker, the use of passively collected longitudinal RTT measurements can be used to identify opportune times to perform such active measurements and to sanity check the claimed data paths. In the following we describe and evaluate the CrowdSec approach and how well such systems are able to detect attacks (when attacks occur), while maintaining low alert rates under normal conditions (when there is no attack).

7.3.1 System model and evaluation framework

We consider a simple model with a set of detectors ($\mathcal{D}$), attackers ($\mathcal{A}$), and victims ($\mathcal{V}$). In the case of a successful *interception attack*, we assume that an attacker $a \in \mathcal{A}$ successfully re-routes the traffic on its way from a detector $d \in \mathcal{D}$ to a victim $v \in \mathcal{V}$ (that would normally take the route $d \rightarrow v \rightarrow d$) such that it instead takes the route $d \rightarrow a \rightarrow v \rightarrow d$. The hijacked path in this example is illustrated in Figure 7.1(a). Similarly, in the case of a successful *imposture attack* (Figure 7.1(b)), we assume that the attacker a intercepts the traffic between the detector d and victim v , and responds directly to the detector d , such that the end-to-end traffic (including the replies) takes the route $d \rightarrow a \rightarrow d$ instead of the intended route $d \rightarrow v \rightarrow d$.

Under the above assumptions, the RTT effects of any successful interception and imposture attack can be estimated based on observed individual RTTs. For example, in the case of a successful interception attack by a on d and v , we compare the new RTT (estimated as $\frac{1}{2}(RTT_{d,a} + RTT_{a,v} + RTT_{d,v})$)

Table 7.1: Summary of datasets analyzed in chapter.

Year	Nodes (ave)	Traceroutes	Simulated attack scenarios	
			Interception	Imposture
2014	113	278,690	15,279	62,576
2015	169	368,887	18,233	—

during the attack) with the previously measured $RTT_{d,v}$ of the original path (over some prior time period). Here, we use $RTT_{x,y}$ to denote the RTT between a source-destination pair (x, y) in our measurements. Similarly, for the imposture attack, we compare the new RTT (estimated as $RTT_{d,a}$ during the attack) with the previously measured $RTT_{d,v}$ of the original path (over prior time period). Finally, we also evaluate the system during an example day without attack, when comparing with the previously measured $RTT_{d,v}$ of the original path (over prior time period).

For our evaluation, we leverage (actively collected) traceroute measurements recorded by PlanetLab nodes as part of the iPlane [32] project. In particular, we use daily RTT measurements from more than 100 globally distributed PlanetLab nodes to other planet lab nodes. We use a month’s worth of training data (e.g., 278,690 successful traceroutes during July 2014) and evaluate the performance of different detection techniques for the following week, during which we simulate different attack combinations. Table 7.1 summarizes the datasets and simulated attacks analyzed in this paper. The lower number of simulated interception attack scenarios (compared to imposture scenarios) are due to both $RTT_{d,a}$ and $RTT_{a,v}$ needing to be present for the day of the attack, in addition to sufficient history of $RTT_{d,v}$ values during the earlier (training) period.

At this point we note that, although the system design proposed in this chapter would use passive measurements, the use of active measurements for the evaluation should not impact our conclusions regarding the effectiveness of the approach. In particular, accurate RTT measurements can be obtained with both measurement approaches. Additionally, given the same source-destination pairs at the same time, the RTT measurements with the two types of measurements should be very similar as long as the active measurements do not significantly impact the load along the path.

7.3.2 Single node detection

We envision that clients monitor their RTTs with different candidate victim IP addresses (or prefixes) and raise alarms when the measured RTTs deviate significantly from previously observed RTTs. For this analysis we apply Grubbs’ test for anomaly detection [137]. We use an initial warmup period of at least N measurements, during which we assume that there are no anomalies, and then apply the Grubbs’ test one measurement at a time. To minimize the effect of RTT variations, each “daily” RTT measurement considered here in fact is generated by taking the minimum RTT over three tracerouts performed each day.

Figure 7.2: Tradeoff between the alert rates of individual clients during attack and normal circumstances.

We have applied both one-sided and two-sided hypothesis tests in each step. One-sided tests are natural for interception attacks, as the re-routing typically will not reduce the RTT. Although imposture attacks can result in both increases and decreases of the RTTs, it is important to note that the impersonator can easily hide such RTT reduction by adding additional delays. Unless otherwise stated, in the following we will show results for one-sided tests.

For our evaluation we use a 31 day warmup period (July 2014 or Jan. 2015) and evaluate the alert rates when there is an attack and when there is no attack, respectively, during the following week (first week of Aug. 2014 and Feb. 2015). Figure 7.2 shows the tradeoff for a single client. Here, we show the fraction of measurements for which the client raises alerts during an attack and the fraction of measurements for which the client raises (false) alerts during normal circumstances, for different thresholds N of the minimum measurements needed before raising alerts. For each threshold N , we varied a threshold-based decision variable p_{ind}^* and counted the fraction of simulated attack (and non-attack) cases in which the observed RTT (with approximate probability p , as calculated with Grubbs' test, for example) was considered an outlier (i.e., cases for which $p \leq p_{ind}^*$) and plotted the measured fractions during attack and non-attack conditions on the y-axis and x-axis respectively, while keeping p_{ind}^* as a hidden variable. We note that in practice the threshold parameter p_{ind}^* can be tuned for a desirable tradeoff. Since the approximate probability values p are only approximations under the assumptions of the test used to calculate them, p_{ind}^* should generally not be interpreted as a strict probability threshold. However, in all cases, the use of p_{ind}^* and p values provides a simple method for deciding where to draw the line for anomalies when the events have been ordered relatively based on a common likelihood measure p .

We note that the detection accuracy significantly depends on the number of measurements prior to detection, and that there are substantial improvements with increasing N . This is seen by the tradeoff curves for larger N being shifted more and more towards the top-left corner. Furthermore, we

see that with a single node, the detection rate is consistently at least an order of magnitude higher than the alert rate during normal circumstances and can be more than 50 times higher for some thresholds. For example, with just $N = 25$ measurements a single node can achieve an alert rate of approximately 50% during attacks at the cost of less than 1% alerts (that may need further investigation) under normal conditions.

Given the high number of packets and high skew in the services that modern clients communicate with [2], most clients will quickly build up a significant sample history for many prefixes. For these services, N may therefore be much greater than 25, allowing for an even more advantageous tradeoff than illustrated here. Next, we look more closely at how these alert tradeoffs can be improved through collaborative information sharing, regardless of the individual clients' alert rate tradeoff.

7.3.3 Collaborative detection

As routing anomalies typically affect many users, CrowdSec can leverage observations from different clients to refine the alert rate tradeoffs, and identify the extent of the anomaly (e.g., by measuring how many clients are affected). With CrowdSec, the users that detect an anomaly share the information with each other, allowing alerts about a particular prefix to be based on the combined anomaly information from multiple observers. In the following, we first describe a general collaboration analysis framework, before briefly discussing different statistical techniques.

We assume that clients always report their significant p-values (smaller than p_{ind}^*) and only report non-significant p-values with a small tunable probability. Here, a p-value represents an estimate of the probability that a client sees the observed RTT value given a history of RTT values, as estimated using Grubbs' test, for example. Given these two sets of p-values and knowledge of the tunable reporting probability, we can estimate both the number of non-reporting clients and the p-value distribution of the clients, allowing us to place the significant p-values in context, while keeping the communication overhead at a (tunable) minimum.

We use hypothesis testing to determine which prefixes and events to flag as potential anomalies. As for the single node case, our null hypothesis is that there is no significant RTT deviations from normal conditions. However, in these collaborative tests we can use all reported p-values. For our evaluation, we have tried three alternatives: the Binomial test, Fisher's test, and Stouffer's test. With the binomial test, a combined probability is calculated based on the binomial distributed probability of observing k or more "significant" p-values out of a total of n , given a probability threshold p_{bin}^* . With Fisher's and Stouffer's test, respectively, corresponding Chi-square and Z-score based metrics are calculated.

While Fisher's and Stouffer's tests theoretically should be able to make better use of the information in the individual p-values, we ran into numer-

Figure 7.3: Alert rate during normal circumstances, during interception attacks, and during imposture attacks.

ical problems with both these tests for the type of values reported in our context. In particular, limited numerical accuracy typically causes rounding errors for these methods, resulting in combined p -scores of either 0 or 1. Of these three methods, only the Binomial test therefore seems applicable to our context. In the following we therefore apply the Binomial test, with a normal distribution approximation when applicable.

7.3.4 Evaluation results

We next take a closer look at the attack detection rate during example attacks and (false) alert rate during normal circumstances when applying the Binomial test. Figure 7.3 shows the alert rates as a function of the probability threshold p_{bin}^* used in the binomial test for the cases when (i) there is no attack (normal circumstances), (ii) interception attacks occur, and (iii) imposture attacks occur. For the attacks, we include results for both single-sided and two-sided tests. As expected, for the interception attacks there is almost no difference between the one-sided and two-sided tests and the imposture attacks generally have lower detection rates (especially in the case where only the one-sided test can be applied).

While the alert rates are closely coupled with the selected probability threshold, it is important to note that there are very large differences in the alert rates during an attack (regardless of the type of attack) and during normal circumstances. This is very encouraging, as it shows that a high alert detection rate (during attacks) can be achieved, while maintaining a small (false) alert rate during normal circumstances. For example, in this case, a threshold $p_{bin}^* = 10^{-20}$ can detect the majority of the interception attacks, without resulting in a single false alert during normal circumstances.

As expected, the absolute alert tradeoffs are highly dependent on the Binomial threshold p_{bin}^* values. For example, both the attack detection (Figures 7.4(a)) and alerts during normal circumstances (Figures 7.4(b)) decrease substantially with decreasing p_{bin}^* values. For these and the re-

Figure 7.4: Alert rates as a function of the binomial threshold p_{bin}^* under different circumstances.

maintaining experiments we have extended our evaluation model to take into account that not all nodes will be affected by the routing anomaly. Here, we have used individual threshold $p_{ind}^* = 10^{-6}$ and assumed that 50% of the nodes are effected. This choice is based on the work of Ballani et al. [58], which suggests that the fraction of ASes whose traffic can be hijacked by any AS varies between 38% and 63% for imposture attacks, between 29% and 48% for interception attacks, and the fraction increases to 52-79% for tier-1 ASes. Figure 7.4(c) shows the impact of the percent of affected nodes (with 40 detector nodes). Note that the alert rates for normal conditions here are the same as for the case when none of the detectors are affected; i.e., the right-most curve in Figure 7.4(c).

In all of the above cases, there is a substantial region of p_{bin}^* values for which we observe significant differences between the attack detection rates and the alert rates under normal conditions (for that same p_{bin}^* value). Figure 7.5 better illustrates these differences for the case of interception attacks. (The results for imposture attacks are similar, and have been omitted due to lack of space.) Here, we plot the fraction of successfully detected attacks during interception attacks as a function of the alert rate during normal circumstances, with the threshold value p_{bin}^* as a hidden variable. For example, Figure 7.5(a) simply combines the information in Figures 7.4(a) and Figures 7.4(b) into a more effective representation of the tradeoffs when there are different numbers of detectors.

We note that there are substantial improvements when using additional detectors (Figure 7.5(a)). For example, with 60 detectors (50% of which are affected) we can achieve a detection rate of 50% while maintaining an alert rate (under normal conditions) below 10^{-4} . This is more than two orders of magnitude better than with a single node (Figure 7.2), illustrating the power of the information sharing and distributed detection achieved with the CrowdSec approach.

Figure 7.5(b) shows the tradeoffs for different fractions of affected nodes. Here, 0% affected nodes corresponds to the case when no detector nodes are affected and the 100% case is when all nodes are affected. As expected, the

Figure 7.5: Tradeoff between the fraction of detected attacks during interception attacks and the false alert rate during normal circumstances.

best tradeoffs are achieved when many nodes are affected. However, with as little as 30% affected nodes, substantial improvements in the detection tradeoff curve are possible (difference between 0% and 30% curves) and an attack detection rate of 50% can be achieved with an alert rate of less than 10^{-2} .

Overall, the low false positive rates (note x-axis on log scale) combined with high detection rates (y-axis on linear scale) suggests that the CrowdSec approach provides an attractive tradeoff as long as there is a reasonable number of affected detectors. Of course, careful data path measurements would always be needed to validate and diagnose the underlying causes behind any detected route anomaly.

Finally, Figure 7.5(c) shows the impact of the individual threshold value p_{ind}^* used to decide when a candidate anomaly should be reported (and considered in the binomial test). While there are some (smaller) benefits from using larger individual threshold p_{ind}^* values when trying to keep the communication overhead at a minimum (left-most region), we note that there are no major differences in the tradeoffs achieved with different threshold values. Section 7.5 takes a closer look at the communication overhead.

7.4 Detector selection

This section looks closer at which nodes are the best detectors and which detectors' alerts are best combined.

Figure 7.6(a) and 7.6(b) show the attack detection rate (during attacks) and alert rate during normal circumstances, as reported by nodes at different distances, respectively, when running all possible interception victim-attacker-detector attack combinations within our dataset. Here, we group triples based on their detector-victim distance $RTT_{d,v}$ and detector-attacker distance $RTT_{d,a}$. Although these distances are most applicable to imposture attacks (where the RTT on the y-axis replaces the RTT on the x-axis), we note that the detection rates during an interception attack (Figure 7.6(a)) also are high for all values above the diagonal, while none of the buckets re-

Figure 7.6: Alert rates as function of the relative RTT distances between detector, attacker, and victim.

sult in substantial alert rates under normal conditions (Figure 7.6(b)). This is encouraging, as it suggests that it may be possible to use the same set of nodes as detectors for both imposture attacks and interception attacks.

While the distance $\frac{1}{2}(RTT_{d,a} + RTT_{a,v})$ in most cases (except when the triangle inequality is violated) is no less than the original distance $RTT_{d,v}$, we also include a heatmap for the detection rate for these candidate distances in Figure 7.6(c). As expected only points close to the diagonal (with similar RTTs for the two cases) have intermediate detection rates.

Thus far, we have ignored the relative distances of the affected nodes. We next take a closer look at how the set of affected nodes impacts the best set of detector nodes to combine. For this analysis, we extend our system model to include a parameter β that determines the set of detector nodes most likely to be affected by an attack. In particular, we assume that a detector node $d \in \mathcal{D}$ is affected by an attacker a of victim v with a probability proportional to $\frac{1}{RTT_{d,a}^\beta}$. When $\beta = 0$ there are no biases and when $\beta = 1$ the probability is inversely proportional to the RTT distance to the attacker. This bias model is motivated by ASes close to the attacker being most likely to be affected by the attack [58].

To assess the impact of node selection to protect candidate victims v , we similarly consider candidate policies in which sets of candidate detectors are selected based on their distance to the victim v . We have considered both policies that skew the probability according to a Zipf distribution (with probability proportional to $\frac{1}{r^\alpha}$, where r is the rank of the node, after sorting them based on distance to the victim, with rank 1 being closest to the victim) and based on the relative distance to the victim (with probabilities being assigned proportional to $\frac{1}{RTT_{d,v}^\alpha}$). As the conclusions are the same for both policy types, we only show results for the rank-based Zipf approach.

Figures 7.7(a) and 7.7(b) show the tradeoff in detection rates as a function of the alert rates for different skew (β) in the affected detectors and different Zipf parameters (α) for the selection probabilities, respectively. In these figures we have used $\beta = 1$ and $\alpha = 1$ as our default values. Although the tradeoff curves are better (with higher detection rates, given the

Figure 7.7: Detection tradeoff for different skew in the rates at which nodes are affected and detector nodes selected, respectively. Default parameters: $\alpha = 1$, $\beta = 1$, 50% affected nodes, and 40 detector nodes.

same alert rates) when the affected nodes are less biased (small β), the bias (Figure 7.7(a)) has a relatively modest impact on the tradeoff. This result suggests that our results thus far are robust to which nodes are affected by the attacks.

However, referring to the impact of detector selection (Figure 7.7(b)), it is interesting to see that there are some additional tradeoffs to take into consideration when selecting detectors for each candidate victim. For example, when low false positives are desired, there are benefits to biasing the selection to intermediate skews. However, the results are relatively insensitive to smaller variations in the chosen bias. In fact, a fairly simple policy that picks detectors uniformly at random ($\alpha = 0$) achieves a relatively nice tradeoff even when there is an underlying bias of which nodes are affected.

Whereas most of the results presented here have been for interception attacks using the Jul/Aug. 2014 iPlane data, we note that the results for imposture attacks are similar and that the results are not limited only to that time period. Figure 7.8 summarizes our results for both imposture and interception attacks, and for the Jan/Feb. 2015 time period. Here, we show the attack detection rates for example scenarios where we vary one factor at a time (both to a smaller and to a larger value) relative to a default scenario when we fix the alert rates during normal circumstances at 0.01. While the imposture attacks are somewhat more difficult to detect, the general tradeoffs otherwise remain the same for the different parameters.

Finally, having discussed the alert rate tradeoffs and how they are impacted by different factors, we note that the best tradeoffs and parameter values will be system dependent. For example, parameters that are suitable for a small system may result in excessive communication overhead and reporting in larger systems. As systems grow in size, the individual parameters (e.g., p_{ind}^* , p_{bin}^*) and alert rates may therefore need to be tuned, and alerts may need to be aggregated (hierarchically, for example) in order to reduce the overhead and improve the accuracy.

Figure 7.8: Attack detection rates under different attacks, while keeping a fixed alert rate of 0.01. Default parameters: $\alpha = 1$, $\beta = 1$, 50% affected nodes, and 40 detector nodes.

7.5 Candidate architectures and overhead

This paper describes a general crowd-based approach for passive RTT monitoring and collaborative anomaly detection to help identify routing anomalies. Although we envision that the system would primarily be implemented at stationary end hosts run by concerned citizens, the system could also easily be operated at any node that can perform passive end-to-end RTT measurements, including at operator operated proxies and middle boxes. In this section we discuss the different ways a general CrowdSec system could be implemented.

Monitoring: Host-based packet capture (pcap) tools for network traffic monitoring and analysis is available for Unix-based, Linux-based, and Windows systems. Many of these tools use the libpcap/WinPcap libraries and allow RTT measurements to be passively recorded. For example, generic tools can be created based on the basic per-packet timing information available in standard tools such as Wireshark¹ or tcpdump². Intrusion detection systems such as Bro³, also provide built-in functionality for a wide range of per-packet, per-object, per-connection, or per-host analysis.

Alternatively, browser-based extensions such as Firebug⁴ and Fathom [139] could also be used to collect RTT measurements and other connection information. While socket-based methods typically incur lower delay overhead than HTTP-based methods [140], it has been shown that Fathom is

¹Wireshark, <https://www.wireshark.org/>, May. 2015.

²tcpdump, <http://www.tcpdump.org/>, May. 2015.

³Bro, <https://www.bro.org/>, May. 2015.

⁴Firebug, <http://getfirebug.com/>, May. 2015.

able to eliminate most upper-layer RTT measurement overhead [139]. Such browser-based measurement techniques may therefore be an attractive monitoring option, as they allow greater operating system flexibility and simplify ubiquitous deployment.

Architecture: The general CrowdSec approach is not bound to a particular architecture and could be implemented using both centralized and decentralized architectures. At one end, a centralized directory service would allow clients to report their outliers (and corresponding p-values) to a central server that would then perform the “collaborative” outlier detection and apply the Binomial test. This solution is both simple and effective, but places a lot of responsibility at one or more central nodes. At the other end of the spectrum, Distributed Hash Tables (DHTs) such as Chord [34] and Pastry [68] can be used to distribute the workload across participating nodes. In this case, the reports are sent to a “holder” node of a particular prefix, where holders can easily be determined using a prefix-aware DHT [105].

For the collaborative detection calculations, the holder (or server) responsible for each prefix must keep track of outlier reports and maintain an estimate of the total number of current detectors for the prefix. Such estimates can easily be done through periodic or probabilistic detector reporting, for example. Although it is possible to further offload the holder nodes (or centralized directory servers) by moving the collaborative detection calculations to a subset of the detector nodes, these calculations can easily be done at very low overhead on the holder nodes.

Detector selection: In the case it is not feasible to maintain reports from all detectors, a subset of detectors may be selected. As we have seen here, there are some benefit tradeoffs based on which detectors are used. Similar to in our simulations, such detector selection could be performed by assigning detector weights based on their relative distance to the monitored prefix (candidate victim). Assuming that detectors report their RTTs (as measured to the monitored prefix), such weighting and/or selection could easily be done at the holders or central servers.

Privacy: Only a subset of RTT measurements will need to be shared by participants. Detectors would not be expected to reveal how often services/prefixes are accessed and should have the option to opt in/out of monitoring selected services/prefixes. This can easily be achieved by ensuring that only the minimum necessary information is shared in the outlier reports. For example, in particularly privacy concerned systems the reports could report only whether a measurement is an outlier or not (not the p-values themselves). Note that as long as all detectors use the same individual threshold p_{ind}^* the Binomial tests applied here would still be directly applicable.

Overhead: Without loss of generality, assume that each detector node $d \in \mathcal{D}$ keeps track of a set $\mathcal{V}_d$ of prefixes (each associated with a candidate victim). Furthermore, let $D = |\mathcal{D}|$ denote the total number of detectors, $\overline{V}_d = \frac{1}{D} \sum_{d \in \mathcal{D}} |\mathcal{V}_d|$ the average number of prefixes monitored per detector

node, and $V = |\cup_{d \in \mathcal{D}} \mathcal{V}_d|$ the total prefixes monitored.

Using this notation, a centralized directory would keep track of V prefixes and in a decentralized setting each holder node (assuming all detectors also act as holders) would be responsible for on average $\frac{V}{D}$ prefixes. It is also easy to show that the total number of individual outlier reports inserted into the systems (either at the centralized directory or a holder node, for example) is proportional to $\sum_{d \in \mathcal{D}} |\mathcal{V}_d| = D\bar{V}_d$. For example, in the case that the D detector nodes also act as holder nodes, each holder node would see a reporting rate equal to the average outlier rate observed by a single detector node (such as itself), equal to $\bar{V}_d$ times the individual nodes' per-prefix outlier alert rate.

Furthermore, the total number of active measurements (e.g., traceroutes) needed to further investigate if the data path had in fact been compromised is equal to the product of the the number of detectors per prefix ($\frac{DV_d}{V}$) times the total collaborative alert rate (equal to V times the collaborative per-prefix alert rate). After cancelation of the V terms, we note that the number of active measurements triggered by the system would be proportional to $D\bar{V}_d$, but this time with a smaller proportionality constant (equal to the collaborative per-prefix alert rate). Spread over D detector nodes, such measurement effort is very low, especially given the relatively low (compared to a non-collaborative system, for example) collaborative per-prefix alert rates observed with the help of the collaborative techniques discussed in this paper.

7.6 Related work

Passive and active end-to-end measurements: Network level end-to-end measurements have been used for a wide range of purposes, including identifying and troubleshooting disruptions on the Internet. For example, NetDiagnoser [141] leverage the end-to-end measurements between sensors and knowledge of the network topography to provide a troubleshooting tool that allows ISPs to detect the location of network failures. Others have used active probing techniques to detect Internet outages [142] and other network reachability issues [143]. Quan *et al.* [142] propose to perform active probes using ICMP pings to addresses within /24 address blocks in the IPv4 address space. With their approach, potential outages are detected when a sharp drop followed by a later increase in the overall responsiveness for the block is observed. Hubble [143] is a hybrid approach that combines passively collected BGP data and active measurements to detect network reachability issues. Passive network measurement data have been used to characterize censorship by nation-states or the effects of natural disasters such as earthquakes on communication [144]. In contrast, we focus on detecting interception and imposture attacks, which are generally difficult to detect since these attacks do not lead to service outages or interruptions.

Crowd-based techniques: Client-side measurement tools and browser

extensions have been designed to measure a wide range performance and security issues, including website performance, port filtering, IPv6 support, and DNS manipulations [139, 145]. Passive monitoring from a large number of users has also been used to characterize geographic differences in the web infrastructure [146], to build location-based services [147], to build AS-level graph [148], and to detect outages and other network events [149]. None of these works consider crowd-based detection of routing anomalies caused by stealthy routing attacks such as interception and imposture attacks.

BGP attack detection: Techniques such as prefix filtering and crypto-based techniques such as Resource Public Key Infrastructure (RPKI) [21], BGPsec [51], and DNSSEC-based [49] approaches such as ROVER [48] have been proposed to cryptographically securing and authorizing BGP route origins. Unfortunately, deployment of these solutions has been slow and the solutions do not work well unless a large number of networks deploy them [11, 15]. The main reasons for slow deployment have been limited incentive for independent organizations, and the absence of a single centralized authority that can mandate the deployment of a (common) security solution. Deployment may also have been hampered by the political and business implications of hierarchical RPKI management giving some entities (e.g., RIRs) significant control over global Internet routing [50].

Without large-scale deployment of crypto-based solutions [11], organizations often instead rely on centralized and decentralized monitoring solutions for anomaly detection in BGP. For example, BGPMon⁵ and Team Cymru⁶ centrally collect routing information from distributed monitors, and create alerts/summary reports about routing anomalies to which organizations can subscribe. PrefiSec [105] and NetReview [93] provide distributed and/or collaborative alternatives.

Section 2.4 described several data-plane based, control-plane based, and hybrid techniques for detecting various routing attacks on the Internet. Typically data-plane based techniques use active traceroute measurements by different organizations and control-plane techniques rely on the AS-PATH in different BGP route announcements. For example, Zheng et al. [29] use measurements to detect imposture and interception attacks. However, in contrast to our passive RTT measurements, their framework requires significant active (traceroute) measurements. Shi et al. [59] shows the value of using distributed diagnosis nodes when detecting black-holing attacks. The authors propose to correlate control-plane routing information and data-plane reachability data to aid in the detection of attacks.

Rather than relying on active measurements or sharing of BGP announcements between organizations, this chapter considers an alternative approach in which we leverage passive measurements at the clients to detect suspicious RTT deviations caused by underlying attacks. This allows any concerned citizen to help detect anomalous routes, which then can be ana-

---

⁵BGPMon, <http://www.bgpmon.net/>, May 2014.

⁶Team Cymru, <http://www.team-cymru.org>, May 2014.

lyzed more closely using active measurements and complimentary control-plane information. We show that seemingly simple measurement data such as RTT collected passively can be effectively used to raise alerts for possible interception and imposture attacks.

7.7 Conclusions

In this paper we have presented and evaluated a user-centric crowd-based approach in which users passively monitor their RTTs, share information about potential anomalies, and apply combined collaborative statistics such as the Binomial test to identify potential routing anomalies. We have shown that the approach incurs low overhead, and provides an attractive tradeoff between attack detection rates (when there is an attack) and alert rates (needing further investigation) under normal conditions. Detection-alert tradeoffs and weighted detector selection, based on candidate detectors' distance from the potential victim nodes, have been evaluated using longitudinal RTT measurements from a wide range of locations. We have shown that there are significant advantages to collaboration (e.g., as shown by the high attack detection rates that can be achieved without adding many additional alerts during normal circumstances), that there are benefits from introducing a slight bias towards selecting nearby detectors, and that these systems can be effectively implemented using relatively simple monitoring (e.g., on client machines) and data sharing over both central and distributed architectures at a low overhead.

We believe that this type of system can allow concerned citizens to take a greater role in protecting the integrity of their own and others' data paths. Given the slow deployment of operator-driven solutions (typically cryptographic and requiring significant adaptations to work properly), this type of passive crowd-based monitoring provides an effective way to bring traffic anomalies (due to attacks, router misconfigurations, or other valid reasons) to the forefront. As such, this type of crowd-based system can push more operators to take actions to provide better protection for the data traffic paths of their customers and others. Future work includes analyzing the interaction between active and passive measurements to diagnose traffic anomalies. Interesting research questions include determining the best way to combine longitudinal passively collected information (e.g., as collected and analyzed here) with targeted traceroutes and other complementary information (such as topology information) to determine the chain of events that allowed the anomaly to take place.

Chapter 8

Does Scale, Size, and Locality Matter? Evaluation of Collaborative BGP Security Mechanisms

8.1 Introduction

A number of important questions arise when considering cooperative information sharing across ASes and other network entities/organizations for the purpose of detecting or preventing routing attacks. For example, how do the detection/prevention rates of the different techniques scale with the number of participants? What is the impact of the size of each participant, or the information available to the participant? And what is the impact of the location of the participants sharing the information? The latter question may be particularly important as it may help provide insights into the effectiveness of regional government-issued legislation or regional agreements. For example, the United States (US) government or the European Union (EU) may push to have ASes and organizations under their respective jurisdictions share information in order to protect the common interests of each region.

Prior work on routing security mechanisms has used data-driven analysis to illustrate the power of large-scale information sharing between large ASes, but little attention has been paid to the effect of the geographic locality of each participant. Although many ASes have points-of-presence in many geographic regions, ASes operated by organizations from the same country or geographic region may be more likely to share information with each other openly or through legislation, for example. Ongoing geographical

and political polarization may further contribute to potential location-based participation and sharing restrictions. Motivated by these observations, in addition to analyzing each of the above three questions on their own, this chapter places particular focus on the impact of the locality of the participants. Locality is considered both on its own and with regard to size-based inclusion within and across regions, as well as with regard to the scale of the (local or global) information sharing alliance.

The main contribution of this chapter is a systematic data-driven evaluation of some promising, previously proposed hijack prevention and routing attack detection techniques. In particular, we consider the questions outlined above in the context of three broad classes of techniques that share (i) prefix origin information, (ii) route path updates, or (iii) passively collected round-trip time (RTT) information.

For our evaluation, we develop a data-driven methodology for each information sharing approach that takes into account the geographic locality (e.g., the region in which the AS is registered) and the relative size (e.g., measured by the number of neighboring ASes) of each of the potential participants. Using real-world topologies and routing information derived from measurement data, we then systematically evaluate the impact of each factor, either on its own, or accounting for the geographic locality of the participants, attackers, and victims.

Our results provide insights into the tradeoffs between global and local deployment. While the results highlight the value of detection and prevention close to the source of an attack, we also find cases where regional collaboration may achieve many of the benefits achievable through global deployment. Other interesting findings include the observation that the largest ASes are not always the best at hijack detection when the attacks are from other regions. Instead, collaboration with mid-sized ASes may be beneficial. This is in contrast to the deployment of hijack prevention mechanisms, which benefit significantly from large ASes participating, regardless of whether the deployment is global or regional. Our scale- and size-based evaluation also provides insights into other deployment related issues, including the relative deployment benefits during different phases of an incremental rollout.

Chapter outline: Section 8.2 provides background and sets the context. The following three sections present our evaluation results for three general classes of collaborative prevention and detection systems. In Section 8.3 we evaluate (sub)prefix hijack prevention techniques that use prefix origin information, in Section 8.4 we evaluate hijack detection mechanisms that use path announcements, and in Section 8.5 we evaluate interception and imposture detection techniques that use passively collected RTT measurements. Finally, related work and conclusions are presented in Sections 8.6 and 8.7, respectively.

Table 8.1: Examples of systems, the information they share/use, and the attacks they can help detect/prevent.

Informa- tion shared	Prefix hijack	Subprefix hijack	Inter- ception	Impos- ture	Example solutions
Prefix origin	✓	✓	✗	✗	Route filter- ing [11, 150], RPKI [21], ROVER [48]
Route path updates	✓	✓	✗	✗	PHAS [22], Pre- fiSec [105], PG- BGP [23]
Passive mea- sure- ments	✗	✗	✓	✓	CrowdSec [87]
Active mea- sure- ments	✗	✗	✓	✓	Zheng et al. [29], PrefiSec [105]

8.2 Collaborative information sharing

Various systems have been proposed to detect, mitigate, or prevent routing attacks and other unwanted routing incidents. These systems typically rely on collaborating ASes sharing different information. For our analysis we focus on three broad classes of techniques that share and/or use the first three types of information in Table 8.1. They correspond to prefix origin information, route path updates, and passively collected RTTs. Route path updates can easily be collected at individual routers or at the AS level, and then shared with other ASes. RTTs can easily be passively collected and shared by almost any network entity [87]. For more general or detailed descriptions, we refer the interested reader to Section 2.4, or in the case of PrefiSec (Chapters 4 and 5) and CrowdSec (Chapter 7), to their respective chapters. In the following, we describe how the different systems that we evaluate here use the shared information.

8.2.1 Hijack prevention using prefix origin

Ideally, a hijack prevention mechanism should prevent an AS from accepting and propagating bogus route announcements. If implemented widely, such mechanisms could then prune away bogus route announcements close to the source and prevent an attacker from reaching more ASes and users.

Previously proposed mechanisms that can provide hijack prevention in-

clude prefix filtering [11], PG-BGP [23], RPKI [48], and ROVER [151]. Many of these techniques build a trusted and formally verifiable database of prefix-to-AS pairings between the IP prefixes and the ASes that are allowed to originate them. If used correctly, routers implementing RPKI [48] and ROVER [151] can be assumed to pick the right origin and avoid propagating the wrong origin for prefixes. To achieve this, two conditions must be satisfied. First, the AS that wants to defend itself against route hijacks on its prefixes, or its provider AS, must register a prefix-to-AS mapping with RPKI or ROVER. Second, as part of their validation of prefix-to-AS mappings, relying ASes must successfully retrieve and check these AS-to-prefix mappings from the RPKI or ROVER records.

With PG-BGP [23], the acceptance of suspicious routes is delayed, and routes are accepted and propagated only after a certain threshold time duration has passed. Since suspicious routes are typically short-lived [152], the performance of PG-BGP is usually similar to that of RPKI and ROVER. Therefore, for the purpose of our evaluation, we only simulate the performance of RPKI and ROVER.

8.2.2 Control-plane based anomaly detection

There are several works that are based on control-plane data for the detection of anomalies in BGP routing, including PHAS [22], PrefiSec [105], and PG-BGP [23]. While PHAS and PG-BGP aggregate all information centrally, PrefiSec distributes computing and detection across participants. Otherwise, the approaches are relatively similar. For each prefix, these protocols track the origin ASes observed by its participants and raises alerts when there are changes. The common idea leveraged by all these protocols is that an IP prefix should be originated by a single AS. An IP prefix originated by more than one AS results in a Multiple Origin AS (MOAS) conflict. While some MOAS are legitimate and can be observed over long time periods [53], a newly-detected MOAS conflict can be an indication of a potential prefix hijack. By keeping track of the AS-to-prefix mappings observed in AS-PATH announcements, these protocols can flag new potential MOAS cases. Naturally, as more ASes participate and share their observed path announcements, the system will have more complete AS-to-prefix mappings.

8.2.3 Route anomaly detection using passive and active measurements

Section 2.3 described several outcomes that are possible if hijack attacks are successful. In particular, Figures 2.6(b) and 2.6(c) were used to illustrate two difficult-to-detect outcomes, called imposture and interception attacks, respectively. In an *imposture attack*, the attacker impersonates the intended destination for the traffic and in an *interception attack* the attacker redirects

the traffic to its intended destination, possibly after making a copy or modifying the data, for example. These attacks are particularly stealthy when the users originating the traffic receive uninterrupted service.

Both active traceroute-based anomaly detection techniques [29, 105] and passive RTT-based anomaly detection techniques [87] have been proposed to detect these attacks. While we will focus on the use of passive measurements, we note that the approaches are in general fairly similar. For example, Zheng et al. [29] use changes in the number of hops in the traceroute paths to identify potential hijacks, while Hiran et al. [87] use changes in the RTTs to identify potential anomalies.

In both types of systems, measurement information from multiple sources is shared to provide stronger evidence and more accurate flagging of suspicious events. For example, in CrowdSec [87] clients or other network entities (e.g., middleboxes) share RTT outlier information and collaboratively identify prefixes with many affected clients, so as to identify potential routing anomalies. For collaborative detection, the system uses statistical tests based on binomial hypothesis testing. One of the main advantages of using passive measurements is that, in contrast to active measurements such as traceroutes, they do not add additional traffic overhead.

8.3 Evaluating hijack prevention techniques

Several studies have suggested that there are significant benefits to deploying hijack detection and prevention mechanisms on several large ASes across the world. However, global deployment that spans multiple geographic regions and jurisdictions is non-trivial and may not be practical due to political and economic reasons. It may be more practical to push or incentivize the deployment within a geographic region such as the US or EU. For example, governmental legislations or other regional mechanisms may be used to push or incentivize agreements between ASes within a region.

In this chapter, we evaluate and compare the benefits and drawbacks of deploying three different general classes of prevention and detection techniques regionally versus globally. For each class of techniques we simulate the effectiveness of the general technique when different subsets of potential candidate participants employ the technique and share information between each other. Within this context, we then answer questions related to the impact of locality and size of the participants, as well as the number of participants. For example, what is the impact of the number of ASes that deploy the hijack prevention mechanisms, either from a specific region (e.g., North America or Europe) or globally that deploy the hijack prevention mechanisms? And, what is the impact of size of the ASes that deploy the hijack prevention mechanisms from a specific region or globally?

In this section we answer the above questions in the context of hijack prevention mechanisms such as route filtering [11, 150], RPKI [21], and ROVER [48].

8.3.1 Simulation-based evaluation methodology

For our simulation-based evaluation, we modified and extended the existing BSIM [23] simulator. BSIM simulates route propagation using the standard Gao-Rexford model [153], which captures the behavior of the economy-driven policies used in practice [41]. The model distinguishes between *customer-provider* relationships (where the customer AS pays its provider) and *peer-peer* relationships (where two ASes often agree to transit each other's traffic for free). In particular, the model assumes that ASes use a routing policy in which customer routes may be exported to all neighboring ASes, but routes learned from peers or providers are exported only to the customers. In addition, the policy prefers customer routes over peer routes (since they bring revenue) and peer routes over provider routes (since provider routes cost money). In cases of multiple tied routes, the routes with the shortest AS paths are chosen. Finally, for the purpose of the simulations, if there are still ties, these ties are broken (arbitrarily) by picking the route over the AS with the lowest AS number.

We extracted the Internet AS-level topology and AS relationship information for every pair of neighboring ASes from public data [154]. We use a snapshot from August 2015, which contains 51,507 ASes and 199,540 AS relationships.

For evaluating hijack prevention mechanisms, we simulate how the routes would propagate in the presence of hijack prevention mechanisms compared to the case when these mechanisms are not present. We measure the fraction of ASes that end up forwarding packets along the correct path in both scenarios and report the percentage increase in the number of ASes that choose the correct origin.

To calculate the percentage increase we first simulate each example attack when no ASes deploy the prevention mechanism and when a random subset of ASes deploy the mechanism, respectively, and then report the average increase in the number of ASes that route to the correct destinations when the prevention mechanism was deployed. In each example scenario, we perform simulations by randomly choosing victim and attacker ASes from selected example regions. Across all scenarios, we randomly picked N ASes to deploy the mechanism from the set of ASes with at least X neighboring ASes, and reported the average over 500 simulations per scenario (with 95% confidence intervals for the average). The use of threshold is in part motivated by larger ASes (with many neighbors) being more likely to have the resources to deploy hijack prevention mechanisms [26]. Here, the degree threshold X is used to bias the size of the individual participants and the parameter N captures the size (scale) of the alliance as a whole.

To compare different deployments, we use locality-, size-, and scale-based criteria to randomly pick subsets of the nodes on which to implement the mechanism. In all simulations, victim nodes are selected at random and the reported metrics are calculated over all nodes in the network.

As is common practice, for our evaluation we varied one parameter at a

Figure 8.1: Average percentage improvement in the number of ASes that choose the correct origin when different subsets of the global set of ASes participate.

time, while keeping all the other parameters constant. Our default degree threshold $X = 20$ was selected to map to an intermediate value in the range of interest (0-50), and the default alliance size N was selected to be equal to the number of ASes with at least 50 neighbors.

Before presenting our results, it should be noted that the simulations have limitations. First, the AS relationship data used for the simulations is not perfect and does not take into account more complex AS-to-AS relationships. For example, two ISPs may interconnect at multiple peering points and have different types of relationships at each point [154]. Second, not all network operators follow the standard rules for route export. However, it is believed that there are few exceptions [41].

8.3.2 Global baseline: scale and size

For reference, we first present results when the participating ASes are selected from the global set of ASes. Figure 8.1 summarizes these results. Figure 8.1(a) shows the percentage improvement in the number of ASes that chose the correct origin, as a function of the number of participating ASes. With our default threshold $X = 20$, the right-most points correspond to the case when all 2,626 ASes with at least 20 neighbors participate. In comparison, Figure 8.2 shows the same plot for ASes in North America (NA), the European Union (EU), and the rest of the world (all ASes excluding those in NA and EU).

Referring to the global deployment results (Figure 8.1(a)), all regions observe significant advantages from higher participation. For example, with 500 random participants we observe an average improvement of more than 15% across all victim-attacker pair scenarios. In comparison, when all ASes with degree of at least 20 participate the improvements are consistently above 45%. While overall numbers are lower when only local ASes participate (Figure 8.2), we note that local deployment is important when pro-

Figure 8.2: Impact of the number of participating ASes, when ASes are selected from a particular geographical region or the “rest of the world”.

tecting against attacks from within the region. This is demonstrated by the higher percentage of improvements when the attacker is in the region deploying the security mechanism.

Figure 8.1(b) shows the percentage improvement as a function of the threshold degree X . With our default alliance size $N = 1,093$, the right-most points correspond to the case when all ASes with a degree of at least 50 participate.

From these results it is clear that the high-degree ASes are the ones that offer the most protection. For example, if all the 1,093 top-ASes with more than 50 neighbors participate we observe improvements of more than 40% for all victim-attacker scenarios. This shows the importance of getting the large ASes onboard in these deployment efforts. The general observation that collaboration by a few large ASes can provide much of the protection is not new. Similar observations have been made by Gersch et al. [25] and Karlin et al. [23], for example. In this work, we take this analysis one step further and consider the impact of regional deployment.

Figure 8.3 presents location-based results for when only ASes in a certain region deploy the prevention mechanism, and where ASes deploying the mechanism are selected based on their degree. We note that regional deployment can provide similar improvements as in a global deployment when the attacker is local. The improvements are noticeably lower when the attackers are located in other regions. For example, the percentage gain in ASes choosing the correct origin for attackers in NA is greater when ASes in NA deploy the prevention mechanisms compared to the gains when ASes in other regions deploy the mechanisms. These results illustrate that enforced deployment of these mechanisms may be a good way for regions to clean up their own networks.

The locations of victim networks play a smaller role. Although all networks would benefit from such a deployment, the local networks would not gain much more protection than external networks. These mechanisms should perhaps best be seen as mechanisms for the greater good, with the

Figure 8.3: Impact of the degree threshold of the participating ASes, when all are selected from a geographic region or the “rest of the world”. For these figures, we choose $N = 207$, $N = 571$, and $N = 315$, respectively.

results showing that there is great incentive for governments and network operators to come together to help ensure that prevention mechanisms are deployed on a large scale.

8.3.3 Location-based discussion

It is often stated that you should *keep your friends close and your enemies closer*. Our results highlight that this is also an important lesson in today’s networks. First, starting with our friends, our results show that there are substantial gains from local deployment, regardless of where the attacks come from. For example, if all ASes with a degree of at least 20 deploy these mechanisms we observe a 30% gain for the NA-based victims regardless of attacker region (e.g., Figure 8.2(a)). In EU (Figure 8.2(b)) the corresponding gain is 20%.

Second, considering the attackers, our regional results clearly show that detectors close to the attackers help the most. For example, when ASes in NA deploy hijack prevention schemes, the damage from hijack attacks originating from ASes in this region can be significantly controlled in all regions. These results show that the largest benefits come with global deployment, and that there may be benefits to subsidizing or otherwise incentivizing international partners to implement these mechanisms too.

Overall, the percentage gain when hijack prevention mechanisms are deployed by all (roughly 2,500) ASes around the world with a degree of at least 20 varies between 40% and 50% for different combinations of victim-attacker regions (Figure 8.1(a)). If only 500 random ASes around the world deploy the mechanism, the gain is roughly 15%. By comparison, when the 431 ASes with a degree of at least 20 in NA deploy the hijack prevention mechanisms, the gain varies between 23% to 43%, depending on which victim-attacker pair combination is considered (Figure 8.2(a)). The higher numbers partially reflect the big impact of the NA-based ASes, many of which are high-degree ASes with peering points around the globe, but also demonstrate the value of regional deployment to help protect against hijack attacks.

8.4 Evaluating hijack detection mechanisms

8.4.1 Methodology and datasets

To evaluate hijack detection mechanisms based on AS-PATH updates we have extended and modified a framework that was previously used to evaluate alert rates for PrefiSec [105] to account for ASes and their locality. While the evaluation framework is designed for PrefiSec, the results presented here also apply to PHAS [22] and PG-BGP [23]. Given the same information, these systems' detection rates are the same. The main differences between these systems are their communication overhead and where the processing is performed.

For our analysis, we collected the RIB files and AS-PATH announcements observed at all six routeviews servers active during the time of the China Telecom incident [9], on April 8, 2010, when China Telecom announced origin for 50,000 prefixes originated by other ASes. Using announcements from around the time, we compare differences and similarities of the detection rates during an actual attack.

Focusing on a two-week window around the time of the incident, we first used the RIB files from April 1 and a warm-up period to initialize the AS-to-prefix mappings seen by different selected subsets of ASes. Of particular interest here is the degree (size) and locality of the collaborating ASes. In our evaluation, we consider sets of collaborating ASes selected from NA, EU, the “rest of the world” (reference point, rather than a region), and from the global set. In contrast to the original evaluation frameworks, which treated the routeviews servers as the participants [105], we use the AS information of the ASes contributing announcements to the routeviews servers and AS-to-region mappings to identify subsets of information seen by different subsets of collaborating ASes.

In total, the six routeviews servers have 100 vantage points that belong to 73 unique ASes. Of these, 38 are NA-based, 21 EU-based, and 14 map to other geographic regions.

For each of the subsets of collaborating ASes that we chose, we then look at each day in the time window and simulate and report the number of prefixes and origins, respectively, that the ASes in the subset would not have seen prior to that day. These two metrics directly measure the number of cases that must be flagged (and further investigated) as potential prefix and sub-prefix attacks, respectively.

8.4.2 Global baseline

As a baseline, we first present results for when the collaborating ASes are selected globally. Figure 8.4(a) shows the number of alerts raised for both “new prefixes” (possible subprefix hijacks) and “new prefix origins” (possible prefix hijacks) announced during the incident (on April 8) as a function of

Figure 8.4: Average number of alerts raised when global ASes collaborate the day of the China Telecom incident.

number of collaborating ASes. We also include the number of alerts of these two types raised due to announcements made by China Telecom.

We see that the number of alerts for possible prefix hijacks increases with the number of collaborating ASes, and that 40,575 alerts (for both prefix and subprefix hijacks) are raised during the day of the attack if all the nodes collaborate. With the exception of a few “new prefixes” and “new prefix origins”, almost all alerts are due to the China Telecom announcements associated with the incident, which caused traffic for these prefixes and subprefixes to be hijacked.

Only a few ASes are needed to detect the majority of the subprefix hijacks (“new prefixes”). This result can be explained by subprefixes being propagated to almost all ASes due to more specific prefixes being preferred. For prefix attacks (“new origin”) additional ASes are much more beneficial, with some diminishing returns after reaching 40 ASes. This happens because ASes during these instances become divided into two groups: ASes that continue routing to the victim network and ASes that choose to route to the attacker network. Thus, additional collaborating ASes increases the chance that conflicting origins are detected and hijack alerts are raised.

Figure 8.5 puts the above numbers in perspective, showing the number of alerts for the days before and after the attack. In addition to being orders of magnitude lower than during the day of the incident, the flatter “new origin” curves suggest that the “new origin” announcements during these days propagated somewhat further than the China Telecom announcements.

Figure 8.4(b) shows the number of alerts as a function of the degree threshold to be included in the alliance. For every threshold, 10 ASes with a degree of at least X are selected at random. Here, the right-most displayed threshold is picked so that the selection set include exactly 10 ASes, and the following points (moving to the left) are picked so as to roughly double the selection set for each point. The degree threshold of 1 is included as a reference point.

Figure 8.5: Average number of alerts raised when global ASes collaborate the day before (April 7) and after (April 9) the incident.

The figure shows that the number of alerts for the China Telecom incident is higher when the degree threshold is small, and the number of alerts is quite low when large ASes collaborate. This is a very interesting observation as much prior work has suggested collaboration between the largest ASes, but it can be partially¹ explained by most of the high degree ASes being NA-based. For example, of the ASes with a degree greater than 1,174, all but one (i.e., 9 of 10) are NA-based, and when the threshold is 646, there are 18 NA-based and 2 EU-based. However, these NA-based ASes do not have as good a vantage point of the China-based incident, with only a subset of the paths propagating to these ASes. With a lower degree threshold more ASes from outside NA and EU will be included, improving the results. This illustrates that the vantage points offered by global collaboration can be more valuable to the prefix hijack detection than having only the large ASes collaborate. Similarly, multi-hop BGP peering can also help. The detection numbers for subprefix attacks (“new prefixes”) are less dependent of the AS degree (size) and locality; again, indicating their wider propagation.

8.4.3 Location-based analysis

We now discuss the benefits of regional collaboration for hijack detection. Figure 8.6 shows the number of alerts as a function of number of ASes for different regions. For all of the three regions (NA, EU, and “rest of the world”), the number of alerts increases as more ASes share information. If all NA-based ASes collaborate there are 22,178 alerts (13,214 “new origin” and 8,964 “new prefix”). Sharing among all EU-based ASes raises 10,829 (3,620+7,209) alerts and sharing among all the ASes in the “rest of the world” category would raise 36,328 (27,280+9,048) alerts. Whereas the sub-prefix detection (“new prefix”) is similar for the different regions, the differences in total alerts are substantial. For example, despite there being far fewer ASes in the “rest of the world” category, this category has the

¹ Additional explanation will be provided in the next subsection.

Figure 8.6: Number of alerts during the day of the incident (April 8, 2010) for different sizes of regional collaborations.

Figure 8.7: Impact of the size of the participating ASes on the number of alerts. For each degree threshold we choose 10 ASes with a degree equal to or greater than the applied threshold.

highest detection rate. The main reason for this is that many of these ASes have more vantage points closer to China Telecom than NA-based and EU-based ASes may have, and therefore have better visibility of the route announcements made by China Telecom. This observation mirrors the insights provided by our hijack prevention results (Section 8.3) that show that ASes deploying protection mechanisms close to the attacker provide the best protection.

While none of the regional collaborations performs as good as global collaboration, the value of regionally deployed solutions should not be underestimated, especially as there is no solution that has seen widespread deployment yet. These results show that careful regional deployment, possibly with a few complementing ASes from other regions, may provide a significant step in the right direction.

Figures 8.7(a) and 8.7(b) show the number of alerts as a function of the degree threshold for regional collaborations in NA and EU, respectively. As for the global results, for each degree threshold, we randomly pick 10 ASes per alliance.

We again observe stronger degree (size) dependence for prefix hijack detection (“new origins”) than for subprefix hijack detection (“new prefixes”). While the large ASes in NA in general provide more alerts than the smallest ASes in NA, it is very interesting that the very top ASes see a drop in the number of alerts they raise. It is also interesting that the large ASes in EU detect fewer attacks than the smaller ASes in EU. As the above ASes are in the same region, our previous explanations (in Section 8.4.2) regarding the relative differences in coverage seen by ASes in *different regions* no longer apply here. In the *same region*, the size-based differences may instead be related to the standard route export policy. In particular, malicious routes (learnt from a peer or provider) are typically exported only to customers. Therefore, malicious routes learnt by mid-tier ASes may not reach their providers (typically large ASes).

8.5 Interception and imposture detection

8.5.1 Methodology and datasets

To provide insights into the impact of regional collaboration for detecting interception and imposture attacks, we have extended the evaluation of CrowdSec [87] to account for locality of the collaborating network entities. CrowdSec is designed to raise alerts about RTT anomalies and help detect interception and imposture attacks. In the case of an interception attack (Figure 2.6(c)) the RTTs typically increase during an attack, whereas the RTTs during an imposture attack (Figure 2.6(b)) can either increase or decrease, depending on the relative locality of the attacker, victim, and detector.

In CrowdSec the end clients passively collect RTT measurements while in contact with different candidate victim IP addresses (or prefixes). The client applies an outlier detection test to raise an alert if the new RTT measurement deviates significantly from previously observed RTT measurements. These alerts are shared with other CrowdSec clients, and the individual alerts are combined using statistical test methods such as a binomial test that takes into account the likelihood of N clients observing significant deviations in RTT measurements to the same prefix, given past observations [87].

For the evaluation presented here, passively collected RTT values are simulated by extracting RTTs from (active) traceroute measurements performed by PlanetLab² nodes as part of the iPlane [32] project. In particular, we use daily RTT measurements associated with 106 NA-based nodes, 79 EU-based nodes, and 36 nodes located in other parts of the world. For the most part we use a month’s worth of training data (e.g., 278,690 successful traceroutes during July 2014) and evaluate the performance of different detection techniques for the following week, during which we simulate different

²PlanetLab, <https://www.planet-lab.org/>

Figure 8.8: Detection tradeoff, shown as the tradeoff between the detection rate during attack and false alert rate during normal circumstances, for varying percentages of affected nodes.

attack combinations. In total, we simulate 15,279 interception attacks and 62,576 imposture attacks per set of sample detectors, and report results averaged over 10 such sample sets. While we only present interception results, the results for imposture attacks are similar. In each simulation, detector nodes and affected nodes are selected randomly within each region.

8.5.2 Global and location-based evaluation

Figure 8.8 shows a comparison of tradeoffs in detection rate during a simulated attack (y-axis) and the false alert rate under normal circumstances (x-axis), when all global vantage points are collaborating (Global) and when only those in North America (NA) or Europe (EU) collaborate. We include results for when all (100%), half (50%), or none (0%) of the potential detector nodes are affected. The case when no nodes (0%) are affected is included only as a reference point, and captures the false positive rates during normal circumstances.

Interestingly, the 106 NA-based nodes achieve a tradeoff that is almost as good as the larger global collaboration (with 221 nodes). For example, a global collaboration that allowed a false alert rate of 10^{-2} would achieve a detection rate of 80%, if 50% of the nodes were affected. In the same scenario, the NA-based nodes achieve a 70% detection rate and the EU-based nodes achieve a 40% detection rate.

Note, however, that the size of the collaboration may play a big role. In Figure 8.9 we present a regional comparison while keeping the number of detector nodes fixed at 20 and 30. It turns out that NA-based nodes provide much better detection than EU-based nodes, even when taking alliance size into account, and in fact outperform a global alliance. Part of the reason for these differences may be differences in the variability of RTTs. Another possible contributing factor that we have discovered is that some EU-based

Figure 8.9: Detection tradeoff, when keeping the number of detectors fixed. Here, 50% of the nodes are assumed to be affected.

Figure 8.10: Detection tradeoff, shown as the tradeoff between the detection rate during attack and false alert rate during normal circumstances, for varying numbers of detectors. Here, 50% of the nodes are assumed to be affected.

routes (even between two EU-based nodes) go through NA even under “normal” circumstances. In such cases, attacks by networks outside EU may not result in noticeable changes in the RTTs.

8.5.3 Scale of collaboration

In general, regardless of the locality of the alliance, we have observed significant advantages to larger alliances. This is illustrated in Figure 8.10. Here, we show the alert-rate tradeoff for collaborations of different sizes when including nodes that are randomly selected from all global nodes (Figure 8.10(a)) vs. only North America (Figure 8.10(b)). Related to scale, we note that there are benefits to larger numbers of RTT measurements, as this helps to filter out anomalies. While region-based analysis of this aspect is omitted here, we refer the interested reader to our global results [87].

8.6 Related work

A large number of security mechanisms have been proposed to secure Internet routing. As described in Section 8.2, this includes prefix hijack prevention mechanisms based on prefix filtering [11, 150], crypto-based solutions such as RPKI [21] and ROVER [48], hijack detection mechanisms based on changes in prefix origins observed in AS-PATH announcements (e.g., PHAS [22], PrefiSec [105], and PG-BGP [23]), and route hijack detection mechanisms using either passive RTT measurements [87] or active traceroute measurements [29, 105]. Rather than proposing new mechanisms, we evaluate the effectiveness of three broad classes of such mechanisms when they are only partially deployed. We place particular focus on the geographic locality of the collaborating ASes or network entities, while also considering the impact of the collaboration scale and the size of participating ASes.

While partial deployment of BGP security mechanisms has been considered in prior literature [23, 24, 26, 155], the geographic location of participants is almost always ignored. Instead, carefully selected ASes have typically been used to demonstrate the potential of the individual techniques.

For example, Avramopoulos *et al.* [24] demonstrate good protection of a participant’s outgoing and incoming traffic using only the top-5 tier-1 ASes in the world. The authors propose that the participant ASes announce each other’s address spaces so as to coax non-participants into picking routes leading to nearby participants in the group. Once the traffic arrives at a participant then it uses a secure overlay to deliver the traffic to its original destination. The authors suggest that the security gain is improved for the group members when several tier-1 ASes are included in the group that uses the proposed extensions.

Others have relied on the top-tier ASes to demonstrate the effectiveness of PG-BGP [23], path validation protocols such as S-BGP and BGPSec [155], and incentive strategies for deployment of S*BGP [26]. None of these works consider the impact of locality of the ASes that are deploying the security mechanisms. Karlin *et al.* [23] propose PG-BGP, which treats unfamiliar routes cautiously, adopts potential bogus routes slowly, and allows time for secondary processes to check their validity. The objective of their evaluation is to see how well PG-BGP performs for these different deployment strategies. The authors find that PG-BGP can protect 97% of the considered ASs from malicious prefix routes and 85% from bogus sub-prefix routes when deployed only on the 62 “core” ASs, where the authors label an AS as a core AS if it has peer-to-peer relationships with fifteen or more neighbors.

Lychev *et al.* [155] study the security benefits of partially deployed path validation protocols such as S-BGP [19], SoBGP [20], and BGPSec [51]. The authors consider different routing models that vary in the degree of preference given to the secure routes. When choosing secure routes is not preferred, the authors suggest that having tier-2 ASes as early adopters gives better security compared to having tier-1 ASes deploy these solutions.

Similar to these results, our evaluation also suggests that having only large ASes collaborate with each other may result in worse attack detection results compared to the cases in which lower tier ASes are also included in the collaboration.

Gill *et al.* [26] evaluate strategies to drive the deployment of crypto-based routing security protocols such as S-BGP [19] and SoBGP [20]. The authors suggest that the deployment of these protocols would be greatly improved by ASes breaking ties (in their route path selection) in favor of secure paths, and by governments and industry groups concentrating their regulatory efforts or financial incentives on convincing a small set of early adopters to deploy S-BGP and SoBGP. They evaluate their deployment strategy under assumptions of different deployment costs for S-BGP and SoBGP, and show that the fraction of ASes deploying routing security mechanisms increases significantly if ASes with high degree deploy the security mechanisms.

We are not the first to study the impact of the number of participating ASes [25] or their node degree [27]. For example, Suchara *et al.* [27] analyze security gains as a function of increasing the node degree of the ASes that use a BGP security mechanism that filters malicious routes. Similarly to our results, they find significant benefits to deploying the mechanism at high-degree ASes at the core of the Internet.

Gersch *et al.* [25] analyze the effect of increasing the number of ASes using attack prevention techniques. Their results nicely show how the average number of polluted ASes decreases as the number of participating ASes (with higher degrees) increases. Similar to our evaluation, they show that the gains are not significant if a few hundred randomly selected ASes deploy the routing security mechanisms. Again, none of these works consider which geographic region each AS maps to. This can be an important factor when it comes to legislation and other political incentives.

Much work has also been done to understand the slow adoption of RPKI and other solutions [26, 156]. Other orthogonal but interesting work in this domain has focused on design of AS reputation systems that use control-plane information to capture short-lived routes that are often used by malicious ASes [157].

Finally, the original simulation framework used in Section 8.3 has also been used by Karlin *et al.* [23]. We extend the simulator to take into account the geographic locations of the attackers, victims, and collaborating participants. In Sections 8.4 and 8.5 we extend and generalize our prior evaluation frameworks for PrefiSec [105] and CrowdSec [87]. Again, neither of these systematically evaluates the value of scale and size in the context of locality-restricted collaboration. This chapter evaluates three such broad classes of mechanisms.

8.7 Conclusions

Despite BGP's vulnerabilities and the ongoing increase in routing attacks, no universally deployed security solution to such attacks exists. Using simulations based on real measurement data, we have presented a systematic evaluation of three broad classes of prevention and detection techniques. We have focused on the impact that regional, rather than global, deployment could have on their ability to prevent/detect attacks, as well as the impact of AS size of the (regional or global) participants, and the number of ASes that deploy the techniques. While prefix hijack prevention (Section 8.3) and detection (Section 8.4) benefit greatly from deployment close to the source of an attack, it is encouraging to see cases with all three classes of techniques where regional deployment provides substantial benefits. We even find some cases where regional deployment achieves most of the benefits achievable through global deployment, and note that regional deployment with carefully selected participants (e.g., based on AS size) can outperform global deployments that are less carefully planned. Another interesting observation is that the largest ASes may end up providing worse detection than mid-sized ASes, which may see a richer set of bogus announcements. This stands in contrast to deploying hijack prevention mechanisms, for which large ASes appear to provide the greatest benefit. The reason for the better detection rates of mid-sized ASes in the case of hijack detection mechanisms is that they may see a richer set of bogus route announcements. Due to these differences, the best AS selection may therefore depend on whether the system is designed for prevention or detection. We have focused on one class of techniques at a time. Interesting future work could weigh the benefits of the different approaches against each other for different collaboration constellations.

Chapter 9

Conclusions and Future Work

Security is of paramount importance in network and communication systems. However, many systems and protocols used on the Internet were not designed with security in mind. This has resulted in significant vulnerabilities. One glaring example of this is BGP, the de facto interdomain routing protocol used on the Internet, which is vulnerable to large-scale attacks and routing incidents such as the China Telecom incident discussed in Chapter 3 of this thesis. Motivated by these vulnerabilities and by current trends, this thesis makes major contributions towards improving the security of Internet routing, as well as towards protecting edge networks against attacks from other Internet domains. After presenting a case-based study of the China Telecom routing incident, we introduce PrefiSec and CrowdSec, two different collaborative mechanisms for detecting attacks that target BGP on the Internet. We also introduce TRAP, a collaborative mechanism for evaluating e-mail servers, and discuss how it can be used alongside PrefiSec. Finally, we also present an evaluation of previously proposed hijack detection and prevention mechanisms from the perspectives of scale, size, and locality. In the following we summarize our results and discuss potential future works.

9.1 Thesis summary

In Chapter 3, we characterized the China Telecom incident, which occurred on April 8, 2010, and involved a very large scale routing anomaly. We used this incident as a case study, to understand what can be learned about large-scale routing anomalies using public datasets, and about what types of data should be collected to diagnose routing anomalies.

Comparing geographic distribution of the prefixes that were hijacked to global distribution of prefixes in general, our study does not find evidence for an intentional attack. If this had been an attack on a specific country, then one might have expected that a disproportionately high number of prefixes

from that country would have been attacked. This was not the case. Our study also highlighted the fact that out of approximately 50,000 prefixes announced by China Telecom, less than 1% were sub-prefix announcements. In addition, we also showed how providers inadvertently aided in the interception of their customers' traffic by making incorrect decisions in choosing paths to prefixes. In summary, our study supports the conclusion that the incident was a leak of random prefixes in the routing table, though it does not rule out malicious intent.

The case study clearly highlights how the intricacies of Internet routing policies and how the interplay between different network operators can enable attacks such as interception attacks. Such an understanding is extremely important when designing security mechanisms that are intended to protect against routing incidents. In fact, this study also helped us in the design and evaluation of the different routing security mechanisms presented in later chapters. This case study also illustrates that collaboration among network operators can play an important role in detecting and mitigating such attacks.

In Chapter 4, we presented PrefiSec, a framework that enables collaborative evaluation of networks and the prefixes associated with each network. We maintain two registries in PrefiSec: an AS registry and a prefix registry. The AS registry is based on the Chord DHT. For the prefix registry, we modified the Chord DHT to allow for efficient IP prefix lookups. This modification also helped in getting consistent IP prefix lookup results irrespective of which node in the overlay network had initiated the query.

We showed that PrefiSec is scalable and has low communication overhead. Our simulations showed that the average number of end-to-end Chord lookups scales sub-logarithmically as the number of participants in the alliance overlay network increases. PrefiSec provides integrated basic services such as efficient and distributed IP-to-prefix mappings, prefix-to-AS mappings, AS-to-prefix mappings, and AS-to-AS relationships mappings.

Chapter 5 presented mechanisms and policies built with the help of the basic services of PrefiSec. These mechanisms enable collaboration among network entities to detect and protect against a wide range of *network-centric* attacks, including routing attacks and wide-area attacks against edge networks. Our framework combined control-plane data and data-plane data monitoring techniques, and we presented mechanisms to detect prefix hijacks, sub-prefix hijacks, and interception attacks on BGP.

Using a case-based analysis of the China Telecom incident, we showed that our framework would have detected most, if not all, of the anomalous route announcements during the incident. We also showed that the number of prefix hijack alerts scales well as the number of alliance members increases. On a normal day (exemplified by April 7, 2010) the number of prefix alerts decreases from 209 to 138 as the alliance size increases from one to six members. Similarly, the number of sub-prefix hijack alerts decreases from 1,949 to 1,354 as the alliance size increases.

With our framework, we can detect possible cases of interceptions on the Internet. Moreover, we showed how our framework and collaboration among nodes in the alliance network can help reduce the number of valid announcements that might look suspicious at first glance. In particular, we showed that the number of cases that must be manually validated decreases as more members share information with each other. We also showed that the overhead can be decreased by combining previous lookups with knowledge of IP-to-prefix mappings.

PrefiSec combines control-plane and data-plane data to detect attacks on the Internet. By shifting parts of the processing to participant nodes we enable decentralization and show how PrefiSec is scalable. For example, in contrast to existing systems such as PHAS [22] and PG-BGP [23], tracking of AS-prefix pairings is moved to participant nodes instead of performing it at a central location.

Chapter 6 presented a specialized framework that enables host-based evaluation of mail servers. Similar to PrefiSec, TRAP is based on a DHT-based overlay network. TRAP employs a distributed reputation-based system that can help e-mail recipients determine if messages are spam or not. We presented an evaluation of a reputation system. In TRAP, every e-mail server is implicitly assigned multiple holder nodes. Our evaluation showed that trust value divergence tends to increase over time at different trust holder nodes, but remained very small compared to the magnitude of the trust value. Our results also showed that the divergence in trust values is significant only when the trust values are in transition, otherwise trust value divergence is negligible.

We envision that TRAP and other specialized systems could be used in tandem with PrefiSec to evaluate prefixes and networks. For example, PrefiSec could map the malicious hosts detected by TRAP to appropriate prefixes and networks. This way TRAP could help enhance the service provided by PrefiSec.

In Chapter 7, we presented CrowdSec, a crowd-based approach to raise alerts for possible interception and imposture attacks, which are otherwise hard to detect. We showed that our approach provides an attractive tradeoff between attack detection rates (when there is an attack) and alert rates under normal conditions. For example, with 60 detectors (of which 50% are affected) we can achieve a detection rate of 50% while maintaining an alert rate (during normal circumstances) below 10^{-4} . This is more than two orders of magnitude of improvement over a single node acting on its own. This also demonstrates the power of the information sharing and distributed detection achieved with the CrowdSec approach.

We also present an analysis of which set of nodes are the best detectors when identifying the attacks. We extend our system model to determine the set of detector nodes most likely to be affected by an attack and then select the set of detector nodes based on their distance to the victim node. While we see benefits from biasing the detector node selection to intermediate

skews, picking detectors uniformly at random also achieves a relatively nice tradeoff even with the bias in selection of affected nodes.

Although interception and imposture attacks are not reported often, inability to detect these attacks does not imply that such attacks are not common. In fact, it is a great concern that some interception attacks have continued over several months without any detection. In light of these observations, our crowd-based mechanism for detecting such malicious interception and imposture attacks provides a timely and relevant contribution. The main challenge in detecting these attacks is that simple hijack alerts are not enough to identify them. To detect these attacks and collect evidence, one also needs to follow up such alerts with active measurements, such as traceroutes, while the attack is underway. Given the long-lasting duration of some of the previously observed attacks, integrating such data collection into CrowdSec appears highly feasible, but is left for future work.

It should be noted that although we have focused on the context of routing attacks, our proposed mechanisms can also be used to detect other types of hijacks, anomalies, and misconfigurations including hijacked DNS responses. This type of infrastructure could also be leveraged for other services, including detecting cases of differentiated service between customers (when the networks are not allowed to do so), detecting instances and the prevalence of censorship, identifying bottlenecks in the network, so as to help improve network performance and take appropriate actions when needed.

In Chapter 8, we present an evaluation of hijack prevention and detection mechanisms, focusing on scale, size, and locality aspects of the participants. This work answers the question of how the detection/prevention rates of previously proposed routing security mechanisms are affected when they are deployed in particular geographical regions. Although these are fundamental questions, this has not been considered in previous work related to routing security mechanisms.

Simulation-based evaluation of hijack prevention techniques shows that when all ASes with a degree of at least 20 participate, the security gain is consistently over 45%. Similarly, when high degree nodes (with at least 50 neighbors) collaborate, the security gains are consistently over 40%. We also observed that even with few participating ASes, regional deployment can sometimes produce better results compared to random global deployment.

We evaluated the effectiveness of hijack detection schemes using data from the China Telecom incident as an example. The results are rather interesting. For example, we observed that the alerts for the China Telecom incident are higher when the degree threshold is small, and the number of alerts is lower when large ASes collaborate. Such observations were made from both global and local perspectives. From the local perspective, such observations can be explained by the standard route export policy used by ASes in which routes are not exported to provider networks, hence resulting in lower alert rates when large ASes collaborate.

We also found that local deployment of BGP security mechanisms has a

notable impact. This finding is interesting and relevant, since such local deployment can be legally enforced. Given the current stalemate, where none of the previously proposed mechanisms is widely used despite many routing security mechanisms having been proposed and their potential demonstrated, we believe our work offers new insights into a very interesting avenue going forward in implementing these solutions.

It would be naïve to say that we have addressed all issues related to routing attacks and attacks on the edge networks in this thesis. However, the work in this thesis definitely provides significant contributions towards securing networks and users from such attacks. We believe our contributions improve understanding of routing attacks and of the pros and cons of regionally deploying routing security mechanisms. We also present mechanisms that enable collaboration between ASes, routers, and users in a scalable manner. This work also enables interesting future work, which will be discussed in the following section.

9.2 Future work

We present open research problems and point out interesting future work.

- It is believed that the relationships between ASes do not change very often and remain stable over long time periods. It would be interesting to analyze the longitudinal data from public inter-domain routers, decode the relationships among ASes, and maintain records of these relationships. Path hijack attacks often result in the announcement of fake AS relationships, and one can compare newly announced relationships with records. The challenge here is how to best decode the relationships among ASes using routing data [158], or perhaps more importantly, how to best do this in a distributed way such that the mechanisms can be effectively integrated into a collaborative alliance framework. There is existing work [54] that focuses on anomaly detection based on geographical distances among ASes in AS-PATH and that categorizes ASes into core or periphery ASes. Vigna *et al.* [54] have evaluated the response of their system by creating a static topology of Internet AS connections and using geographical information from *whois* services (which can be incomplete, incorrect, and stale). Decentralizing the approach of anomaly detection based on geographical distances among ASes and fitting them into PrefiSec remains as future work. Such classification can be further extended to capture relationships between ASes, including classification into provider-customer, customer-provider, peering, and sibling relationships.
- In our current evaluation of our proposed traceroute-based path hijack detection mechanism, we use public looking glass servers to perform traceroutes to hosts within the prefix block that we suspect are being

targeted with path hijack attacks. Using looking glass servers has some disadvantages. In particular, the rate at which we can query these servers is often limited. These challenges could potentially be overcome by using open research platforms such as PlanetLab¹ [159]. However, while PlanetLab nodes may allow traceroutes to be performed at a faster rate, new challenges arise. For example, with many nodes to choose from, how should PlanetLab nodes be selected to achieve the best possible detection capabilities?

- Different networks add different value to the alliance. Interesting future work includes determining effective methods to evaluate nodes' behavior and usefulness to the alliance network. Such evaluation can help to identify nodes that would be particularly valuable to add to the alliance network. One could also design a payment system, where nodes make/receive payments based on their contributions to the alliance.
- Future work can also include active data collection of malicious activities, such as spamming and scanning attacks. For example, honeypots could be deployed to collect spam mails and detect spam sources.² Scanning attacks could be identified using network traces [66]. Such datasets could then be used to compare host-based and prefix-based approaches. For example, instead of existing methods of blacklisting individual IP addresses, the IP addresses could be mapped back to IP prefix ranges, and then those IP prefix ranges could be blacklisted.
- The bootstrapping aspect of PrefiSec can be studied. PrefiSec requires the participation of existing ASes. It cannot be expected that all participant ASes would start using PrefiSec at once. It is important to study how PrefiSec could be bootstrapped and how it could be expanded to include other ASes. Interesting questions here include how to incentivize participation, both for early adopters, as well as for later adopters.

---

¹PlanetLab, <http://www.planet-lab.org/>

²Project Honey Pit, "Help stop spammers before they even get your address", <http://www.projecthoneypot.org/>, Aug. 2016

Bibliography

- [1] International Telecommunication Union, “Measuring the Information Society Report,” Nov. 2013. [Online]. Available: <http://www.itu.int/en/ITU-D/Statistics/Documents/publications/misr2015/MISR2015-ES-E.pdf>
- [2] P. Gill, M. Arlitt, N. Carlsson, A. Mahanti, and C. Williamson, “Characterizing organizational use of web-based services: Methodology, challenges, observations, and insights,” *ACM Transactions on the Web*, vol. 5, no. 4, pp. 19:1–19:23, Oct. 2011.
- [3] Symantec, “Consumer cybercrime estimated at \$110 billion annually,” Sep. 2012. [Online]. Available: http://www.symantec.com/about/news/release/article.jsp?prid=20120905_02
- [4] R. Anderson, C. Barton, R. Boehme, R. Clayton, M. van Eeten, M. Levi, T. Moore, and S. Savage, “Measuring the cost of cybercrime,” in *Proc. WEIS*, Berlin, Germany, Jun. 2012.
- [5] J. Hong, “The state of phishing attacks,” *Communications of the ACM*, vol. 55, no. 1, pp. 74–81, Jan. 2012.
- [6] H. Khurana, M. Hadley, N. Lu, and D. Frincke, “Smart-grid security issues,” *IEEE Security & Privacy*, vol. 8, no. 1, pp. 81–85, Jan. 2010.
- [7] phys.org, “The cyberattack on Ukraine’s power grid is a warning of what’s to come,” Jan. 2016. [Online]. Available: <http://phys.org/news/2016-01-cyberattack-ukraine-power-grid.html>
- [8] ISO/IEC, “Information technology-security techniques-information security management systems-overview and vocabulary,” Feb. 2016. [Online]. Available: [http://standards.iso.org/ittf/PubliclyAvailableStandards/c066435_ISO_IEC_27000_2016\(E\).zip](http://standards.iso.org/ittf/PubliclyAvailableStandards/c066435_ISO_IEC_27000_2016(E).zip)
- [9] R. Hiran, N. Carlsson, and P. Gill, “Characterizing large-scale routing anomalies: A case study of the China Telecom incident,” in *Proc. PAM*, Hong Kong, China, Mar. 2013.
- [10] Dyn research, “The new threat: Targeted internet traffic misdirection,” Nov. 2013. [Online]. Available: <http://research.dyn.com/2013/11/mitm-internet-hijacking/>

- [11] K. Butler, T. Farley, P. McDaniel, and J. Rexford, "A survey of BGP security issues and solutions," *Proc. IEEE*, vol. 98, no. 1, pp. 100–122, Jan. 2010.
- [12] A. Pilosov and T. Kapela, "Stealing the Internet: An Internet-scale man in the middle attack," Live demonstration at Defcon 16, Las Vegas, NV, Aug. 2008.
- [13] A. Ramachandran, A. Dasgupta, N. Feamster, and K. Weinberger, "Spam or ham?: characterizing and detecting fraudulent "not spam" reports in web mail systems," in *Proc. CEAS*, Perth, Australia, Sept. 2011.
- [14] N. Carlsson and M. Arlitt, "Leveraging organizational etiquette to improve Internet security," in *Proc. IEEE ICCCN*, Zurich, Switzerland, Aug. 2010.
- [15] S. Goldberg, "Why is it taking so long to secure Internet routing?" *ACM Queue*, vol. 12, no. 8, pp. 327–338, Oct. 2014.
- [16] H. Ringberg, A. Soule, and M. Caesar, "Evaluating the potential of collaborative anomaly detection," Tech. Rep., 2008.
- [17] J. Schlamp, G. Carle, and E. W. Biersack, "A forensic case study on as hijacking: The attacker's perspective," *ACM SIGCOMM Computer Communication Review*, vol. 43, no. 2, pp. 5–12, Apr 2013.
- [18] Y. Zhang, E. Cooke, and Z. M. Mao, "Internet-scale malware mitigation: Combining intelligence of the control and data plane," in *Proc. ACM Workshop on Recurring Malcode*, Fairfax, VA, USA, Nov. 2006.
- [19] S. Kent, C. Lynn, and K. Seo, "Secure Border Gateway Protocol (S-BGP)," *IEEE Journal on Selected Areas in Communications*, vol. 18, no. 4, pp. 582–592, Apr. 2000.
- [20] R. White, "Securing BGP through secure origin BGP," *The Internet Protocol Journal*, vol. 6, no. 3, pp. 15–22, Sep. 2003.
- [21] M. Lepinski and S. Kent, "An infrastructure to support secure Internet routing," RFC 6480 (Informational), IETF, Feb. 2012. [Online]. Available: <http://www.ietf.org/rfc/rfc6480.txt>
- [22] M. Lad, D. Massey, D. Pei, Y. Wu, B. Zhang, and L. Zhang, "PHAS: a prefix hijack alert system," in *Proc. USENIX Security Symp.*, Vancouver, Canada, Jul. 2006.
- [23] J. Karlin, S. Forrest, and J. Rexford, "Pretty good BGP: Improving BGP by cautiously adopting routes," in *Proc. IEEE ICNP*, Santa Barbara, CA, Nov. 2006.

- [24] I. Avramopoulos, M. Suchara, and J. Rexford, “How small groups can secure interdomain routing,” Princeton University, Tech. Rep., Nov. 2007.
- [25] J. Gersch, D. Massey, and C. Papadopoulos, “Incremental deployment strategies for effective detection and prevention of BGP origin hijacks,” in *Proc. IEEE ICDCS*, Madrid, Spain, Jun. 2014.
- [26] P. Gill, M. Schapira, and S. Goldberg, “Let the market drive deployment: A strategy for transitioning to BGP security,” in *Proc. ACM SIGCOMM*, Toronto, Ontario, Canada, Aug. 2011.
- [27] M. Suchara, I. Avramopoulos, and J. Rexford, “Securing BGP incrementally,” in *Proc. ACM CoNEXT*, New York, NY, Dec. 2007.
- [28] N. Shahmehri, D. Byers, and R. Hiran, “Trap: Open decentralized distributed spam filtering,” in *Proc. TrustBus*, Toulouse, France, Aug/Sep. 2011.
- [29] C. Zheng, L. Ji, D. Pei, J. Wang, and P. Francis, “A light-weight distributed scheme for detecting IP prefix hijacks in real-time,” in *Proc. ACM SIGCOMM*, Kyoto, Japan, Aug. 2007.
- [30] R. Hiran, N. Carlsson, and N. Shahmehri, “Does scale, size, and locality matter? evaluation of collaborative BGP security mechanisms,” in *Proc. IFIP Networking*, Vienna, Austria, May 2016.
- [31] University of Oregon, “Route views project,” <http://www.routeviews.org/>.
- [32] H. Madhyastha, T. Isdal, M. Piatek, C. Dixon, T. Anderson, A. Krishnamurthy, and A. Venkataramani, “iPlane: An information plane for distributed services,” in *Proc. OSDI*, Seattle, WA, Nov. 2006.
- [33] R. Jain, *The art of computer systems performance analysis: techniques for experimental design, measurement, simulation, and modeling*. J. Wiley, Apr. 1991.
- [34] I. Stoica, R. Morris, D. Karger, M. F. Kaashoek, and H. Balakrishnan, “Chord: A scalable peer-to-peer lookup service for Internet applications,” in *Proc. ACM SIGCOMM*, San Diego, CA, Aug. 2001.
- [35] P. Garca, C. Pairot, R. Mondjar, J. Pujol, H. Tejedor, and R. Rallo, “Planetsim: A new overlay network simulation framework,” in *Proc. SEM*, Linz, Austria, Sep. 2004.
- [36] V. Paxson, “Strategies for sound Internet measurement,” in *Proc. ACM SIGCOMM IMC*, Taormina, Sicily, Italy, Oct. 2004.

- [37] FreePastry, “Pastry: A substrate for peer-to-peer applications.” [Online]. Available: <http://www.freepastry.org/>
- [38] J. Hawkinson and T. Bates, “Guidelines for creation, selection, and registration of an autonomous system (AS),” RFC 1930, IETF, Mar. 1996. [Online]. Available: <http://tools.ietf.org/html/rfc1930>
- [39] H. Khosravi and T. Anderson, “Requirements for separation of IP control and forwarding,” RFC 3654, IETF, Nov. 2003. [Online]. Available: <http://www.ietf.org/rfc/rfc3654.txt>
- [40] Y. Rekhter and T. Li, “A Border Gateway Protocol 4 (BGP-4),” RFC 1771, IETF, Mar. 1995. [Online]. Available: <http://www.ietf.org/rfc/rfc1771.txt>
- [41] P. Gill, M. Schapira, and S. Goldberg, “A survey of interdomain routing policies,” *SIGCOMM Computer Communication Review*, vol. 44, no. 1, pp. 28–34, Jan. 2014.
- [42] M. Caesar and J. Rexford, “BGP routing policies in ISP networks,” *IEEE Network*, vol. 19, no. 6, pp. 5–11, Nov. 2005.
- [43] E. Chen and J. Stewart, “A Framework for Inter-Domain Route Aggregation,” RFC 2519, IETF, Feb. 1999. [Online]. Available: <https://tools.ietf.org/html/rfc2519>
- [44] Dyn Research. (2008) Pakistan hijacks youtube. [Online]. Available: <http://research.dyn.com/2008/02/pakistan-hijacks-youtube-1/>
- [45] A. Arnbak and S. Goldberg, “Loopholes for circumventing the constitution: Unrestrained bulk surveillance on Americans by collecting network traffic abroad,” in *Proc. HOTPETS*, Amsterdam, Netherlands, Jul. 2014.
- [46] C. Alaettinoglu, C. Villamizar, E. Gerich, D. Kessens, D. Meyer, T. Bates, D. Karrenberg, and M. Terpstra, “Routing Policy Specification Language (RPSL),” RFC 2622, IETF, Jun. 1999. [Online]. Available: <http://www.ietf.org/rfc/rfc2622.txt>
- [47] J. Jung, E. Sit, H. Balakrishnan, and R. Morris, “DNS performance and the effectiveness of caching,” *IEEE/ACM Transactions on Networking*, vol. 10, no. 5, pp. 589–603, Oct. 2002.
- [48] J. Gersch and D. Massey, “ROVER: route origin verification using DNS,” in *Proc. IEEE ICCCN*, Nassau, Bahamas, Jul/Aug. 2013.
- [49] R. Arends, R. Austein, M. Larson, D. Massey, and S. Rose, “DNS Security Introduction and Requirements,” RFC 4033, IETF, Mar. 2005. [Online]. Available: <https://www.ietf.org/rfc/rfc4033.txt>

- [50] D. Cooper, E. Heilman, K. Brogle, L. Reyzin, and S. Goldberg, “On the risk of misbehaving RPKI authorities,” in *Proc. ACM HotNets*, College Park, MD, Nov. 2013.
- [51] M. Lepinski and K. Sriram, “BGPSEC protocol specification,” Internet-Draft, IETF, Aug. 2016. [Online]. Available: <https://tools.ietf.org/html/draft-ietf-sidr-bgpsec-protocol-18>
- [52] B. Kuerbis and M. L. Mueller, “Negotiating a new governance hierarchy: An analysis of the conflicting incentives to secure Internet routing,” *Communications & Strategies*, vol. 1, no. 81, pp. 125–142, Mar. 2011.
- [53] X. Zhao, D. Pei, L. Wang, D. Massey, A. Mankin, S. F. Wu, and L. Zhang, “An analysis of BGP multiple origin AS (MOAS) conflicts,” in *Proc. IMW*, San Francisco, CA, Nov. 2001.
- [54] C. Kruegel, D. Mutz, W. Robertson, and F. Valeur, “Topology-based detection of anomalous BGP messages,” in *Proc. RAID*, Pittsburgh, PA, Sep. 2003.
- [55] L. Subramanian, V. Roth, I. Stoica, S. Shenker, and R. H. Katz, “Listen and whisper: security mechanisms for BGP,” in *Proc. NSDI*, San Francisco, CA, Mar. 2004.
- [56] Z. Zhang, Y. Zhang, Y. C. Hu, Z. M. Mao, and R. Bush, “iSPY: Detecting IP prefix hijacking on my own,” in *Proc. ACM SIGCOMM*, Seattle, WA, Aug. 2008.
- [57] E. L. Wong, P. Balasubramanian, L. Alvisi, M. G. Gouda, and V. Shmatikov, “Truth in advertising: Lightweight verification of route integrity,” in *Proc. ACM PODC*, Portland, OR, Aug. 2007.
- [58] H. Ballani, P. Francis, and X. Zhang, “A study of prefix hijacking and interception in the Internet,” in *Proc. ACM SIGCOMM*, Kyoto, Japan, Aug. 2007.
- [59] X. Shi, Y. Xiang, Z. Wang, X. Yin, and J. Wu, “Detecting prefix hijackings in the Internet with Argus,” in *Proc. ACM IMC*, Boston, MA, Nov. 2012.
- [60] X. Hu and Z. M. Mao, “Accurate real-time identification of IP prefix hijacking,” in *Proc. IEEE S&P*, Oakland, CA, May 2007.
- [61] Symantec, “Internet Security Threat Report,” Symantec Corporation, Apr. 2016. [Online]. Available: http://www.symantec.com/security_response/publications/threatreport.jsp

- [62] Z. Durumeric, E. Wustrow, and J. A. Halderman, “Zmap: Fast Internet-wide scanning and its security applications,” in *USENIX Security*, Washington, D.C., Aug. 2013.
- [63] V. Yegneswaran, P. Barford, and J. Ullrich, “Internet intrusions: Global characteristics and prevalence,” in *Proceedings ACM SIGMETRICS*, San Diego, CA, USA, Jun. 2003.
- [64] M. Allman, V. Paxson, and J. Terrell, “A brief history of scanning,” in *Proc. ACM IMC*, San Diego, CA, Oct. 2007.
- [65] E. Raftopoulos, E. Glatz, X. Dimitropoulos, and A. Dainotti, “How dangerous is Internet scanning? a measurement study of the aftermath of an Internet-wide scan,” in *Workshop on Traffic Monitoring and Analysis*, Barcelona, Spain, Apr. 2015.
- [66] M. Arlitt, N. Carlsson, P. Gill, A. Mahanti, and C. Williamson, “Characterizing intelligence gathering and control on an edge network,” *ACM Transactions on Internet Technology*, vol. 11, no. 1, pp. 2:1–2:26, Jul. 2011.
- [67] R. Rodrigues and P. Druschel, “Peer-to-peer systems,” *Communications of the ACM*, vol. 53, no. 10, pp. 72–82, Oct. 2010.
- [68] A. I. T. Rowstron and P. Druschel, “Pastry: Scalable, decentralized object location, and routing for large-scale peer-to-peer systems,” in *Proc. IFIP/ACM Middleware*, Heidelberg, Germany, Nov. 2001.
- [69] B. Zhao, L. Huang, J. Stribling, S. Rhea, A. Joseph, and J. Kubiawicz, “Tapestry: a resilient global-scale overlay for service deployment,” *IEEE Journal on Selected Areas in Communications*, vol. 22, pp. 41–53, Jan. 2004.
- [70] S. Ratnasamy, P. Francis, M. Handley, R. Karp, and S. Shenker, “A scalable content-addressable network,” *ACM Computer Communication Review*, vol. 31, pp. 161–172, Aug. 2001.
- [71] B. Cohen, “Incentives build robustness in bittorrent,” in *Workshop on Economics of Peer-to-Peer systems*, Berkeley, CA, Jun. 2003.
- [72] S. Nakamoto, “Bitcoin: A peer-to-peer electronic cash system,” 2009. [Online]. Available: <https://bitcoin.org/bitcoin.pdf>
- [73] S. King and S. Nadal, “PPCoin: peer-to-peer crypto-currency with proof-of-stake,” 2012. [Online]. Available: <http://peercoin.net/assets/paper/peercoin-paper.pdf>
- [74] M. Castro, P. Druschel, A.-M. Kermarrec, and A. Rowstron, “Scribe: a large-scale and decentralized application-level multicast infrastructure,” *IEEE Journal on Selected Areas in Communications*, vol. 20, no. 8, pp. 1489 – 1499, Oct. 2002.

- [75] S. Iyer, A. Rowstron, and P. Druschel, "Squirrel: a decentralized peer-to-peer web cache," in *Proc. ACM PODC*, Monterey, CA, Jul. 2002.
- [76] P. Druschel and A. Rowstron, "PAST: a large-scale, persistent peer-to-peer storage utility," in *Proc. HotOS*, Elmau/Oberbayern, Germany, May. 2001.
- [77] A. Mislove, A. Post, C. Reis, P. Willmann, P. Druschel, D. S. Wallach, X. Bonnaire, P. Sens, J.-M. Busca, and L. Arantes-Bezerra, "POST: a secure, resilient, cooperative messaging system," in *Proc. HotOS*, Lihue, Hawaii, May. 2003.
- [78] N. Carlsson, C. Williamson, A. Hirt, and M. Jacobson Jr, "Performance modelling of anonymity protocols," *Performance Evaluation*, vol. 69, no. 12, pp. 643–661, Dec. 2012.
- [79] R. Newman, *Computer Security: Protecting Digital Resources*, 1st ed. Jones and Bartlett Publishers, Inc., 2009.
- [80] D. Dash, B. Kveton, J. M. Agosta, E. Schooler, J. Chandrashekar, A. Bachrach, and A. Newman, "When gossip is good: Distributed probabilistic inference for detection of slow network intrusions," in *Proc. AAAI*, Boston, MA, Jul. 2006.
- [81] D. Moore, C. Shannon, G. Voelker, and S. Savage, "Internet Quarantine: Requirements for Containing Self-Propagating Code," in *Proc. IEEE INFOCOM*, San Francisco, CA, Mar/Apr. 2003.
- [82] C. V. Zhou, C. Leckie, and S. Karunasekera, "A survey of coordinated attacks and collaborative intrusion detection," *Computers and Security*, vol. 29, no. 1, pp. 124 – 140, Feb. 2010.
- [83] S. R. Snapp, J. Brentano, G. V. Dias, T. L. Goan, L. T. Heberlein, C. lin Ho, K. N. Levitt, B. Mukherjee, S. E. Smaha, T. Grance, D. M. Teal, and D. Mansur, "DIDS (Distributed Intrusion Detection System) - motivation, architecture, and an early prototype," in *Proc. National Computer Security Conference*, Washington, D.C., Oct. 1991.
- [84] A. K. Ganame, J. Bourgeois, R. Bidou, and F. Spies, "A global security architecture for intrusion detection on computer networks," *Computers & Security*, vol. 27, no. 12, pp. 30 – 47, Mar. 2008.
- [85] P. A. Porras and P. G. Neumann, "EMERALD: Event Monitoring Enabling Responses to Anomalous Live Disturbances," in *Proc. NISSC*, Baltimore, MD, Oct. 1997.
- [86] M. Locasto, J. J. Parekh, A. D. Keromytis, and S. J. Stolfo, "Towards collaborative security and P2P intrusion detection," in *Proc. IEEE IAW*, West Point, NY, Jun. 2005.

- [87] R. Hiran, N. Carlsson, and N. Shahmehri, "Crowd-based detection of routing anomalies on the Internet," in *Proc. IEEE CNS*, Florence, Italy, Sep. 2014.
- [88] A. Ramachandran and N. Feamster, "Understanding the network-level behavior of spammers," in *Proc. ACM SIGCOMM*, Pisa, Italy, Sep. 2006.
- [89] Y. Chen, K. Hwang, and W.-S. Ku., "Collaborative detection of DDoS attacks over multiple network domains," *IEEE Transactions on Parallel and Distributed Systems*, vol. 18, no. 12, pp. 1649–1662, Dec. 2007.
- [90] V. Sekar, N. Duffield, O. Spatscheck, J. van der Merwe, and H. Zhang, "LADS: large-scale automated DDOS detection system," in *Proc. USENIX ATC*, Boston, MA, Jun. 2006.
- [91] V. Yegneswaran, P. Barford, and S. Jha, "Global intrusion detection in the DOMINO overlay system," in *Proc. NDSS*, San Diego, CA, Feb. 2004.
- [92] J. McGibney and D. Botvich, "A trust based system for enhanced spam filtering," *Journal of Software*, vol. 3, no. 5, pp. 55–64, May. 2008.
- [93] A. Haeberlen, I. Avramopoulos, J. Rexford, and P. Druschel, "NetReview: detecting when interdomain routing goes wrong," in *Proc. NSDI*, Boston, MA, Apr. 2009.
- [94] D. Blumenthal, P. Brookes, R. Cleveland, J. Fiedler, P. Mulloy, W. Reinsch, D. Shea, P. Videnieks, M. Wessel, and L. Wortzel, "Report to Congress of the US-China Economic and Security Review Commission," Report, Nov. 2010. [Online]. Available: http://origin.www.uscc.gov/sites/default/files/annual_reports/2010-Report-to-Congress.pdf
- [95] V. Khare, Q. Ju, and B. Zhang, "Concurrent prefix hijacks: Occurrence and impacts," in *Proc. ACM IMC*, Boston, MA, Nov. 2012.
- [96] Y. Chi, R. Oliveira, and L. Zhang, "Cyclops: The Internet AS-level observatory," *ACM Computer Communication Review*, vol. 38, no. 5, pp. 5–16, Oct. 2008.
- [97] B. Ager, N. Chatzis, A. Feldmann, N. Sarrar, S. Uhlig, and W. Willinger, "Anatomy of a large European IXP," in *Proc. ACM SIGCOMM*, Helsinki, Finland, Aug. 2012.
- [98] R. Oliveira, D. Pei, W. Willinger, B. Zhang, and L. Zhang, "Quantifying the completeness of the observed Internet AS-level structure,"

UCLA Computer Science Department - Technical Report TR-080026-2008, Sep. 2008.

- [99] Z. M. Mao, J. Rexford, J. Wang, and R. H. Katz, "Towards an accurate AS-level traceroute tool," in *Proc. ACM SIGCOMM*, Karlsruhe, Germany, Aug. 2003.
- [100] L. Gao and J. Rexford, "Stable Internet routing without global coordination," *IEEE/ACM Transactions on Networking*, vol. 9, no. 6, pp. 307–317, Dec. 2001.
- [101] E. Gregori, A. Improta, L. Lenzini, L. Rossi, and L. Sani, "On the incompleteness of the AS-level graph: a novel methodology for BGP route collector placement," in *Proc. ACM IMC*, Boston, MA, Nov. 2012.
- [102] P. Gill, M. Schapira, and S. Goldberg, "Modeling on quicksand: Dealing with the scarcity of ground truth in interdomain routing data," *ACM Computer Communication Review*, vol. 42, no. 1, pp. 40–46, Jan. 2012.
- [103] Symantec, "2013 Internet Security Threat Report," Symantec Corporation, Apr. 2013. [Online]. Available: http://www.symantec.com/security_response/publications/threatreport.jsp
- [104] S. Katti, B. Krishnamurthy, and D. Katabi, "Collaborating against common enemies," in *Proc. IMC*, Berkeley, CA, Oct. 2005.
- [105] R. Hiran, N. Carlsson, and N. Shahmehri, "PrefiSec: A distributed alliance framework for collaborative BGP monitoring and prefix-based security," in *Proc. ACM CCS Workshop on Information Sharing and Collaborative Security*, Scottsdale, AZ, Nov. 2014.
- [106] N. Feamster, H. Balakrishnan, J. Rexford, A. Shaikh, and J. van der Merwe, "The case for separating routing from routers," in *Proc. ACM SIGCOMM Workshop on Future Directions in Network Architecture*, Portland, OR, Aug./Sep. 2004.
- [107] J. Scudder, R. Fernando, and S. Stuart, "BGP Monitoring Protocol," Internet-Draft, IETF, Oct. 2012. [Online]. Available: <https://tools.ietf.org/html/draft-ietf-grow-bmp-07>
- [108] J. Fan, J. Xu, M. H. Ammar, and S. B. Moon, "Prefix-preserving IP address anonymization: measurement-based security evaluation and a new cryptography-based scheme," *Computer Networks*, vol. 46, no. 2, pp. 253–272, Oct. 2004.
- [109] A. Gupta, B. Liskov, and R. Rodrigues, "Efficient routing for peer-to-peer overlays," in *Proc. NSDI*, San Francisco, CA, Mar. 2004.

- [110] I. Gupta, K. Birman, P. Linga, A. Demers, , and R. van Renesse, “Kelips: Building an efficient and stable P2P DHT through increased memory and background overhead,” in *Proc. IPTPS*, Berkeley, CA, Feb. 2003.
- [111] Y. Zhang and M. Pourzandi, “Studying impacts of prefix interception attack by exploring BGP AS-path prepending,” in *Proc. IEEE ICDCS*, Macau, China, Jun. 2012.
- [112] P. Marchetta, W. de Donato, and A. Pescapé, “Detecting third-party addresses in traceroute IP paths,” in *Proc. ACM SIGCOMM*, Helsinki, Finland, Aug. 2012.
- [113] B. Augustin, B. Krishnamurthy, and W. Willinger, “IXPs: mapped?” in *Proc. ACM IMC*, Chicago, IL, Nov. 2009.
- [114] Y. Shavitt, E. Shir, and U. Weinsberg, “Near-deterministic inference of AS relationships,” in *Proc. ConTEL*, Zagreb, Croatia, Jun. 2009.
- [115] V. Paxson, “Bro: a system for detecting network intruders in real-time,” *Computer Networks*, vol. 31, no. 23-24, pp. 2435–2463, Dec. 1999.
- [116] C. Duma, N. Shahmehri, and G. Caronni, “Dynamic trust metrics for peer-to-peer systems,” in *Proc. DEXA*, Copenhagen, Denmark, Aug. 2005.
- [117] C. Duma, M. Karresand, N. Shahmehri, and G. Caronni, “A trust-aware, P2P-based overlay for intrusion detection,” in *Proc. DEXA*, Krakow, Poland, Sep. 2006.
- [118] D. L. Mills, “Network time protocol version 4 reference and implementation guide,” Technical Report, University of Delaware, Jun. 2006.
- [119] L. Xiong and L. Liu, “PeerTrust: Supporting Reputation-Based Trust for Peer-to-Peer Electronic Communities,” *IEEE Transactions on Knowledge and Data Engineering*, vol. 16, no. 7, pp. 843–857, Jul. 2004.
- [120] K. Hoffman, D. Zage, and C. Nita-Rotaru, “A Survey of Attack and Defense Techniques for Reputation Systems,” *ACM Computing Surveys*, vol. 42, no. 1, pp. 1:1–1:31, Dec. 2009.
- [121] J. R. Douceur, “The Sybil attack,” in *Proc. IPTPS*, Cambridge, MA, Mar. 2002.
- [122] D. S. Wallach, “A survey of peer-to-peer security issues,” in *Proc. ISSS*, Tokyo, Japan, Nov. 2003.

- [123] Y. Lee and J.-S. Kim, "Characterization of large-scale SMTP traffic: the coexistence of the poisson process and self-similarity," in *Proc. IEEE MASCOTS*, Baltimore, MD, Sep. 2008.
- [124] R. Bhagwan, S. Savage, and G. M. Voelker, "Understanding availability," in *Proc. IPTPS*, Berkeley, CA, Feb. 2003.
- [125] M. Castro, M. Costa, and A. Rowstron, "Performance and dependability of structured peer-to-peer overlays," in *IEEE/IFIP DSN*, Florence, Italy, Jun. 2004.
- [126] SpamAssassin, "The apache spamassassin project." [Online]. Available: <http://spamassassin.apache.org/>
- [127] Y. Gong and Q. Chen, "Research of spam filtering based on bayesian algorithm," in *Proc. ICCASM*, Taiyuan, China, Oct. 2010.
- [128] V. V. Prakash and A. J. O'Donnell, "A reputation-based approach for efficient filtration of spam," White paper, Apr. 2007. [Online]. Available: <http://www.cloudmark.com/en/whitepapers/reputation-based-approach-for-efficient-filtration-of-spam>
- [129] E. Damiani, S. De Capitani di Vimercati, S. Paraboschi, and P. Samarati, "P2p-based collaborative spam detection and filtering," in *Proc. IEEE P2P*, Zurich, Switzerland, Aug. 2004.
- [130] J. S. Kong, B. A. Rezaei, N. Sarshar, V. P. Roychowdhury, and P. O. Boykin, "Collaborative spam filtering using e-mail networks," *Computer*, vol. 39, no. 8, pp. 67–73, Aug. 2006.
- [131] F. Zhou, L. Zhuang, B. Y. Zhao, L. Huang, A. D. Joseph, and J. Kubiatowicz, "Approximate Object Location and Spam Filtering on Peer-to-peer Systems," in *Proc. ACM/IFIP/USENIX Middleware*, Rio de Janeiro, Brazil, Jun. 2003.
- [132] A. Gray and M. Haahr, "Personalised, Collaborative Spam Filtering," in *Proc. CEAS*, Mountain View, CA, USA, Jul. 2004.
- [133] K. Li, Z. Zhong, and L. Ramaswamy, "Privacy-Aware Collaborative Spam Filtering," *IEEE Transactions on Parallel and Distributed Systems*, vol. 20, no. 5, pp. 725–739, May. 2009.
- [134] G. Mo, W. Zhao, H. Cao, and J. Dong, "Multi-agent Interaction Based Collaborative P2P System for Fighting Spam," in *Proc. IEEE/WIC/ACM IAT*, Hong Kong, China, Dec. 2006.
- [135] P. Chirita, J. Diederich, and W. Nejdl, "MailRank: Using Ranking for Spam Detection," in *Proc. CIKM*, Bremen, Germany, Oct./Nov. 2005.

- [136] N. Foukia, L. Zhou, and C. Neuman, “Multilateral decisions for collaborative defense against unsolicited bulk e-mail,” in *Proc. iTrust*, Pisa, Italy, May. 2006.
- [137] F. E. Grubbs, “Sample criteria for testing outlying observations,” *The Annals of Mathematical Statistics*, vol. 21, no. 1, pp. 27–58, Mar. 1950.
- [138] J. Aikat, J. Kaur, F. D. Smith, and K. Jeffay, “Variability in tcp round-trip times,” in *Proc ACM IMC*, Miami Beach, FL, Oct. 2003.
- [139] M. Dhawan, J. Samuel, R. Teixeira, C. Kreibich, M. Allman, N. Weaver, and V. Paxson, “Fathom: A browser-based network measurement platform,” in *Proc. ACM IMC*, Boston, MA, Nov. 2012.
- [140] W. Li, R. K. Mok, R. K. Chang, and W. W. Fok, “Appraising the delay accuracy in browser-based network measurement,” in *Proc. ACM IMC*, Barcelona, Spain, Oct. 2013.
- [141] A. Dhamdhere, R. Teixeira, C. Dovrolis, and C. Diot, “Netdiagnoser: Troubleshooting network unreachabilities using end-to-end probes and routing data,” in *Proc. ACM CoNEXT*, New York, NY, Dec. 2007.
- [142] L. Quan, J. Heidemann, and Y. Pradkin, “Detecting Internet outages with precise active probing,” USC/Information Sciences Institute, Tech. Rep., May 2012.
- [143] E. Katz-Bassett, H. V. Madhyastha, J. P. John, A. Krishnamurthy, D. Wetherall, and T. Anderson, “Studying black holes in the Internet with Hubble,” in *Proc. NSDI*, San Francisco, California, 2008.
- [144] A. Dainotti, R. Amman, E. Aben, and K. Claffy, “Extracting benefit from harm: using malware pollution to analyze the impact of political and geophysical events on the Internet,” *ACM SIGCOMM Computer Communication Review*, vol. 42, no. 1, pp. 31–39, Jan. 2012.
- [145] C. Kreibich, N. Weaver, B. Nechaev, and V. Paxson, “Netalyzer: Illuminating the edge network,” in *Proc. ACM IMC*, Melbourne, Australia, Nov. 2010.
- [146] M. Arlitt, N. Carlsson, C. Williamson, and J. Rolia, “Passive crowd-based monitoring of world wide web infrastructure and its performance,” in *Proc. IEEE ICC*, Ottawa, Canada, Jun. 2012.
- [147] P. Shankar, Y.-W. Huang, P. Castro, B. Nath, and L. Iftode, “Crowds replace experts: Building better location-based services using mobile social network interactions,” in *Proc. IEEE PerCom*, Lugano, Switzerland, Mar. 2012.

- [148] A. Faggiani, E. Gregori, L. Lenzini, V. Luconi, and A. Vecchio, “Smartphone-based crowdsourcing for network monitoring: Opportunities, challenges, and a case study,” *IEEE Communications*, vol. 52, no. 1, pp. 106–113, Jan. 2014.
- [149] D. R. Choffnes, F. E. Bustamante, and Z. Ge, “Crowdsourcing service-level network event monitoring,” in *Proc. ACM SIGCOMM*, New Delhi, India, Aug./Sep. 2010.
- [150] T. Bates, E. Gerich, L. Joncheray, J.-M. Jouanigot, D. Karrenberg, M. Terpstra, and J. Yu, “Representation of IP routing policies in a routing registry,” RFC 1786, IETF, Mar. 1995. [Online]. Available: <https://tools.ietf.org/html/rfc1786>
- [151] S. Kent, “An infrastructure supporting secure Internet routing,” in *Public Key Infrastructure*, 2006, vol. 4043, pp. 116–129.
- [152] P.-A. Vervier, O. Thonnard, and M. Dacier, “Mind your blocks: On the stealthiness of malicious BGP hijacks,” in *Proc. NDSS*, San Diego, CA, Feb. 2015.
- [153] L. Gao and J. Rexford, “Stable Internet routing without global coordination,” *IEEE/ACM Transactions on Networking*, vol. 9, no. 6, pp. 681–692, Dec. 2001.
- [154] X. Dimitropoulos, D. Krioukov, M. Fomenkov, B. Huffaker, Y. Hyun, k. claffy, and G. Riley, “AS relationships: Inference and validation,” *ACM SIGCOMM Computer Communication Review*, vol. 37, no. 1, pp. 29–40, Jan. 2007.
- [155] R. Lychev, S. Goldberg, and M. Schapira, “BGP security in partial deployment: Is the juice worth the squeeze?” in *Proc. ACM SIGCOMM*, Hong Kong, China, Aug. 2013.
- [156] M. Wahlisch, R. Schmidt, T. C. Schmidt, O. Maennel, S. Uhlig, and G. Tyson, “RiPKI: The tragic story of RPKI deployment in the Web ecosystem,” in *Proc. ACM HotNets*, Philadelphia, PA, Nov. 2015.
- [157] M. Konte, R. Perdisci, and N. Feamster, “ASwatch: An AS reputation system to expose bulletproof hosting ASes,” in *Proc. ACM SIGCOMM*, London, United Kingdom, Aug. 2015.
- [158] L. Gao, “On inferring autonomous system relationships in the Internet,” *IEEE/ACM Transactions on Networking*, vol. 9, no. 6, pp. 733–745, Dec. 2001.
- [159] L. Peterson, T. Anderson, D. Culler, and T. Roscoe, “A blueprint for introducing disruptive technology into the internet,” *ACM Computer Communication Review*, vol. 33, no. 1, pp. 59–64, Jan. 2003.

Dissertations

Linköping Studies in Science and Technology

Linköping Studies in Arts and Science

Linköping Studies in Statistics

Linköping Studies in Information Science

Linköping Studies in Science and Technology

- No 14 **Anders Haraldsson:** A Program Manipulation System Based on Partial Evaluation, 1977, ISBN 91-7372-144-1.
- No 17 **Bengt Magnhagen:** Probability Based Verification of Time Margins in Digital Designs, 1977, ISBN 91-7372-157-3.
- No 18 **Mats Cedwall:** Semantisk analys av process-beskrivningar i naturligt språk, 1977, ISBN 91-7372-168-9.
- No 22 **Jaak Urmi:** A Machine Independent LISP Compiler and its Implications for Ideal Hardware, 1978, ISBN 91-7372-188-3.
- No 33 **Tore Risch:** Compilation of Multiple File Queries in a Meta-Database System 1978, ISBN 91-7372-232-4.
- No 51 **Erland Jungert:** Synthesizing Database Structures from a User Oriented Data Model, 1980, ISBN 91-7372-387-8.
- No 54 **Sture Häggglund:** Contributions to the Development of Methods and Tools for Interactive Design of Applications Software, 1980, ISBN 91-7372-404-1.
- No 55 **Pär Emanuelson:** Performance Enhancement in a Well-Structured Pattern Matcher through Partial Evaluation, 1980, ISBN 91-7372-403-3.
- No 58 **Bengt Johnsson, Bertil Andersson:** The Human-Computer Interface in Commercial Systems, 1981, ISBN 91-7372-414-9.
- No 69 **H. Jan Komorowski:** A Specification of an Abstract Prolog Machine and its Application to Partial Evaluation, 1981, ISBN 91-7372-479-3.
- No 71 **René Reboh:** Knowledge Engineering Techniques and Tools for Expert Systems, 1981, ISBN 91-7372-489-0.
- No 77 **Östen Oskarsson:** Mechanisms of Modifiability in large Software Systems, 1982, ISBN 91-7372-527-7.
- No 94 **Hans Lunell:** Code Generator Writing Systems, 1983, ISBN 91-7372-652-4.
- No 97 **Andrzej Lingas:** Advances in Minimum Weight Triangulation, 1983, ISBN 91-7372-660-5.
- No 109 **Peter Fritzson:** Towards a Distributed Programming Environment based on Incremental Compilation, 1984, ISBN 91-7372-801-2.
- No 111 **Erik Tengvald:** The Design of Expert Planning Systems. An Experimental Operations Planning System for Turning, 1984, ISBN 91-7372-805-5.
- No 155 **Christos Levcououlos:** Heuristics for Minimum Decompositions of Polygons, 1987, ISBN 91-7870-133-3.
- No 165 **James W. Goodwin:** A Theory and System for Non-Monotonic Reasoning, 1987, ISBN 91-7870-183-X.
- No 170 **Zebo Peng:** A Formal Methodology for Automated Synthesis of VLSI Systems, 1987, ISBN 91-7870-225-9.
- No 174 **Johan Fagerström:** A Paradigm and System for Design of Distributed Systems, 1988, ISBN 91-7870-301-8.
- No 192 **Dimitar Driankov:** Towards a Many Valued Logic of Quantified Belief, 1988, ISBN 91-7870-374-3.

- No 213 **Lin Padgham:** Non-Monotonic Inheritance for an Object Oriented Knowledge Base, 1989, ISBN 91-7870-485-5.
- No 214 **Tony Larsson:** A Formal Hardware Description and Verification Method, 1989, ISBN 91-7870-517-7.
- No 221 **Michael Reinfrank:** Fundamentals and Logical Foundations of Truth Maintenance, 1989, ISBN 91-7870-546-0.
- No 239 **Jonas Löwgren:** Knowledge-Based Design Support and Discourse Management in User Interface Management Systems, 1991, ISBN 91-7870-720-X.
- No 244 **Henrik Eriksson:** Meta-Tool Support for Knowledge Acquisition, 1991, ISBN 91-7870-746-3.
- No 252 **Peter Eklund:** An Epistemic Approach to Interactive Design in Multiple Inheritance Hierarchies, 1991, ISBN 91-7870-784-6.
- No 258 **Patrick Doherty:** NML3 - A Non-Monotonic Formalism with Explicit Defaults, 1991, ISBN 91-7870-816-8.
- No 260 **Nahid Shahmehri:** Generalized Algorithmic Debugging, 1991, ISBN 91-7870-828-1.
- No 264 **Nils Dahlbäck:** Representation of Discourse-Cognitive and Computational Aspects, 1992, ISBN 91-7870-850-8.
- No 265 **Ulf Nilsson:** Abstract Interpretations and Abstract Machines: Contributions to a Methodology for the Implementation of Logic Programs, 1992, ISBN 91-7870-858-3.
- No 270 **Ralph Rönquist:** Theory and Practice of Tense-bound Object References, 1992, ISBN 91-7870-873-7.
- No 273 **Björn Fjellborg:** Pipeline Extraction for VLSI Data Path Synthesis, 1992, ISBN 91-7870-880-X.
- No 276 **Staffan Bonnier:** A Formal Basis for Horn Clause Logic with External Polymorphic Functions, 1992, ISBN 91-7870-896-6.
- No 277 **Kristian Sandahl:** Developing Knowledge Management Systems with an Active Expert Methodology, 1992, ISBN 91-7870-897-4.
- No 281 **Christer Bäckström:** Computational Complexity of Reasoning about Plans, 1992, ISBN 91-7870-979-2.
- No 292 **Mats Wirén:** Studies in Incremental Natural Language Analysis, 1992, ISBN 91-7871-027-8.
- No 297 **Mariam Kamkar:** Interprocedural Dynamic Slicing with Applications to Debugging and Testing, 1993, ISBN 91-7871-065-0.
- No 302 **Tingting Zhang:** A Study in Diagnosis Using Classification and Defaults, 1993, ISBN 91-7871-078-2.
- No 312 **Arne Jönsson:** Dialogue Management for Natural Language Interfaces - An Empirical Approach, 1993, ISBN 91-7871-110-X.
- No 338 **Simin Nadjm-Tehrani:** Reactive Systems in Physical Environments: Compositional Modelling and Framework for Verification, 1994, ISBN 91-7871-237-8.
- No 371 **Bengt Savén:** Business Models for Decision Support and Learning. A Study of Discrete-Event Manufacturing Simulation at Asea/ABB 1968-1993, 1995, ISBN 91-7871-494-X.

- No 375 **Ulf Söderman:** Conceptual Modelling of Mode Switching Physical Systems, 1995, ISBN 91-7871-516-4.
- No 383 **Andreas Kågedal:** Exploiting Groundness in Logic Programs, 1995, ISBN 91-7871-538-5.
- No 396 **George Fodor:** Ontological Control, Description, Identification and Recovery from Problematic Control Situations, 1995, ISBN 91-7871-603-9.
- No 413 **Mikael Pettersson:** Compiling Natural Semantics, 1995, ISBN 91-7871-641-1.
- No 414 **Xinli Gu:** RT Level Testability Improvement by Testability Analysis and Transformations, 1996, ISBN 91-7871-654-3.
- No 416 **Hua Shu:** Distributed Default Reasoning, 1996, ISBN 91-7871-665-9.
- No 429 **Jaime Villegas:** Simulation Supported Industrial Training from an Organisational Learning Perspective - Development and Evaluation of the SSIT Method, 1996, ISBN 91-7871-700-0.
- No 431 **Peter Jonsson:** Studies in Action Planning: Algorithms and Complexity, 1996, ISBN 91-7871-704-3.
- No 437 **Johan Boye:** Directional Types in Logic Programming, 1996, ISBN 91-7871-725-6.
- No 439 **Cecilia Sjöberg:** Activities, Voices and Arenas: Participatory Design in Practice, 1996, ISBN 91-7871-728-0.
- No 448 **Patrick Lambrix:** Part-Whole Reasoning in Description Logics, 1996, ISBN 91-7871-820-1.
- No 452 **Kjell Orsbom:** On Extensible and Object-Relational Database Technology for Finite Element Analysis Applications, 1996, ISBN 91-7871-827-9.
- No 459 **Olof Johansson:** Development Environments for Complex Product Models, 1996, ISBN 91-7871-855-4.
- No 461 **Lena Strömbäck:** User-Defined Constructions in Unification-Based Formalisms, 1997, ISBN 91-7871-857-0.
- No 462 **Lars Degerstedt:** Tabulation-based Logic Programming: A Multi-Level View of Query Answering, 1996, ISBN 91-7871-858-9.
- No 475 **Fredrik Nilsson:** Strategi och ekonomisk styrning - En studie av hur ekonomiska styrsystem utformas och används efter företagsförvärv, 1997, ISBN 91-7871-914-3.
- No 480 **Mikael Lindvall:** An Empirical Study of Requirements-Driven Impact Analysis in Object-Oriented Software Evolution, 1997, ISBN 91-7871-927-5.
- No 485 **Göran Forslund:** Opinion-Based Systems: The Cooperative Perspective on Knowledge-Based Decision Support, 1997, ISBN 91-7871-938-0.
- No 494 **Martin Sköld:** Active Database Management Systems for Monitoring and Control, 1997, ISBN 91-7219-002-7.
- No 495 **Hans Olsén:** Automatic Verification of Petri Nets in a CLP framework, 1997, ISBN 91-7219-011-6.
- No 498 **Thomas Drakengren:** Algorithms and Complexity for Temporal and Spatial Formalisms, 1997, ISBN 91-7219-019-1.
- No 502 **Jakob Axelsson:** Analysis and Synthesis of Heterogeneous Real-Time Systems, 1997, ISBN 91-7219-035-3.
- No 503 **Johan Ringström:** Compiler Generation for Data-Parallel Programming Languages from Two-Level Semantics Specifications, 1997, ISBN 91-7219-045-0.
- No 512 **Anna Moberg:** Närhet och distans - Studier av kommunikationsmönster i satellitkontor och flexibla kontor, 1997, ISBN 91-7219-119-8.
- No 520 **Mikael Ronström:** Design and Modelling of a Parallel Data Server for Telecom Applications, 1998, ISBN 91-7219-169-4.
- No 522 **Niclas Ohlsson:** Towards Effective Fault Prevention - An Empirical Study in Software Engineering, 1998, ISBN 91-7219-176-7.
- No 526 **Jochim Karlsson:** A Systematic Approach for Prioritizing Software Requirements, 1998, ISBN 91-7219-184-8.
- No 530 **Henrik Nilsson:** Declarative Debugging for Lazy Functional Languages, 1998, ISBN 91-7219-197-x.
- No 555 **Jonas Hallberg:** Timing Issues in High-Level Synthesis, 1998, ISBN 91-7219-369-7.
- No 561 **Ling Lin:** Management of 1-D Sequence Data - From Discrete to Continuous, 1999, ISBN 91-7219-402-2.
- No 563 **Eva L Ragnemalm:** Student Modelling based on Collaborative Dialogue with a Learning Companion, 1999, ISBN 91-7219-412-X.
- No 567 **Jörgen Lindström:** Does Distance matter? On geographical dispersion in organisations, 1999, ISBN 91-7219-439-1.
- No 582 **Vanja Josifovski:** Design, Implementation and Evaluation of a Distributed Mediator System for Data Integration, 1999, ISBN 91-7219-482-0.
- No 589 **Rita Kovordányi:** Modeling and Simulating Inhibitory Mechanisms in Mental Image Reinterpretation - Towards Cooperative Human-Computer Creativity, 1999, ISBN 91-7219-506-1.
- No 592 **Mikael Ericsson:** Supporting the Use of Design Knowledge - An Assessment of Commenting Agents, 1999, ISBN 91-7219-532-0.
- No 593 **Lars Karlsson:** Actions, Interactions and Narratives, 1999, ISBN 91-7219-534-7.
- No 594 **C. G. Mikael Johansson:** Social and Organizational Aspects of Requirements Engineering Methods - A practice-oriented approach, 1999, ISBN 91-7219-541-X.
- No 595 **Jörgen Hansson:** Value-Driven Multi-Class Overload Management in Real-Time Database Systems, 1999, ISBN 91-7219-542-8.
- No 596 **Niklas Hallberg:** Incorporating User Values in the Design of Information Systems and Services in the Public Sector: A Methods Approach, 1999, ISBN 91-7219-543-6.
- No 597 **Vivian Vimarlund:** An Economic Perspective on the Analysis of Impacts of Information Technology: From Case Studies in Health-Care towards General Models and Theories, 1999, ISBN 91-7219-544-4.
- No 598 **Johan Jenvald:** Methods and Tools in Computer-Supported Taskforce Training, 1999, ISBN 91-7219-547-9.
- No 607 **Magnus Merkel:** Understanding and enhancing translation by parallel text processing, 1999, ISBN 91-7219-614-9.
- No 611 **Silvia Coradeschi:** Anchoring symbols to sensory data, 1999, ISBN 91-7219-623-8.
- No 613 **Man Lin:** Analysis and Synthesis of Reactive Systems: A Generic Layered Architecture Perspective, 1999, ISBN 91-7219-630-0.
- No 618 **Jimmy Tjäder:** Systemimplementering i praktiken - En studie av logiker i fyra projekt, 1999, ISBN 91-7219-657-2.
- No 627 **Vadim Engelson:** Tools for Design, Interactive Simulation, and Visualization of Object-Oriented Models in Scientific Computing, 2000, ISBN 91-7219-709-9.

- No 637 **Esa Falkenroth:** Database Technology for Control and Simulation, 2000, ISBN 91-7219-766-8.
- No 639 **Per-Arne Persson:** Bringing Power and Knowledge Together: Information Systems Design for Autonomy and Control in Command Work, 2000, ISBN 91-7219-796-X.
- No 660 **Erik Larsson:** An Integrated System-Level Design for Testability Methodology, 2000, ISBN 91-7219-890-7.
- No 688 **Marcus Bjärelund:** Model-based Execution Monitoring, 2001, ISBN 91-7373-016-5.
- No 689 **Joakim Gustafsson:** Extending Temporal Action Logic, 2001, ISBN 91-7373-017-3.
- No 720 **Carl-Johan Petri:** Organizational Information Provision - Managing Mandatory and Discretionary Use of Information Technology, 2001, ISBN 91-7373-126-9.
- No 724 **Paul Scerri:** Designing Agents for Systems with Adjustable Autonomy, 2001, ISBN 91 7373 207 9.
- No 725 **Tim Heyer:** Semantic Inspection of Software Artifacts: From Theory to Practice, 2001, ISBN 91 7373 208 7.
- No 726 **Pär Carlshamre:** A Usability Perspective on Requirements Engineering - From Methodology to Product Development, 2001, ISBN 91 7373 212 5.
- No 732 **Juha Takkinen:** From Information Management to Task Management in Electronic Mail, 2002, ISBN 91 7373 258 3.
- No 745 **Johan Åberg:** Live Help Systems: An Approach to Intelligent Help for Web Information Systems, 2002, ISBN 91-7373-311-3.
- No 746 **Rego Granlund:** Monitoring Distributed Teamwork Training, 2002, ISBN 91-7373-312-1.
- No 757 **Henrik André-Jönsson:** Indexing Strategies for Time Series Data, 2002, ISBN 917373-346-6.
- No 747 **Anneli Hagdahl:** Development of IT-supported Interorganisational Collaboration - A Case Study in the Swedish Public Sector, 2002, ISBN 91-7373-314-8.
- No 749 **Sofie Pilemalm:** Information Technology for Non-Profit Organisations - Extended Participatory Design of an Information System for Trade Union Shop Stewards, 2002, ISBN 91-7373-318-0.
- No 765 **Stefan Holmlid:** Adapting users: Towards a theory of use quality, 2002, ISBN 91-7373-397-0.
- No 771 **Magnus Morin:** Multimedia Representations of Distributed Tactical Operations, 2002, ISBN 91-7373-421-7.
- No 772 **Pawel Pietrzak:** A Type-Based Framework for Locating Errors in Constraint Logic Programs, 2002, ISBN 91-7373-422-5.
- No 758 **Erik Berglund:** Library Communication Among Programmers Worldwide, 2002, ISBN 91-7373-349-0.
- No 774 **Choong-ho Yi:** Modelling Object-Oriented Dynamic Systems Using a Logic-Based Framework, 2002, ISBN 91-7373-424-1.
- No 779 **Mathias Broxvall:** A Study in the Computational Complexity of Temporal Reasoning, 2002, ISBN 91-7373-440-3.
- No 793 **Asmus Pandikow:** A Generic Principle for Enabling Interoperability of Structured and Object-Oriented Analysis and Design Tools, 2002, ISBN 91-7373-479-9.
- No 785 **Lars Hult:** Publika Informationstjänster. En studie av den Internetbaserade encyklopedins bruksegenskaper, 2003, ISBN 91-7373-461-6.
- No 800 **Lars Taxén:** A Framework for the Coordination of Complex Systems' Development, 2003, ISBN 91-7373-604-X
- No 808 **Klas Gäre:** Tre perspektiv på förväntningar och förändringar i samband med införande av informationssystem, 2003, ISBN 91-7373-618-X.
- No 821 **Mikael Kindborg:** Concurrent Comics - programming of social agents by children, 2003, ISBN 91-7373-651-1.
- No 823 **Christina Ölvingson:** On Development of Information Systems with GIS Functionality in Public Health Informatics: A Requirements Engineering Approach, 2003, ISBN 91-7373-656-2.
- No 828 **Tobias Ritzau:** Memory Efficient Hard Real-Time Garbage Collection, 2003, ISBN 91-7373-666-X.
- No 833 **Paul Pop:** Analysis and Synthesis of Communication-Intensive Heterogeneous Real-Time Systems, 2003, ISBN 91-7373-683-X.
- No 852 **Johan Moe:** Observing the Dynamic Behaviour of Large Distributed Systems to Improve Development and Testing - An Empirical Study in Software Engineering, 2003, ISBN 91-7373-779-8.
- No 867 **Erik Herzog:** An Approach to Systems Engineering Tool Data Representation and Exchange, 2004, ISBN 91-7373-929-4.
- No 872 **Aseel Berglund:** Augmenting the Remote Control: Studies in Complex Information Navigation for Digital TV, 2004, ISBN 91-7373-940-5.
- No 869 **Jo Skåmedal:** Telecommuting's Implications on Travel and Travel Patterns, 2004, ISBN 91-7373-935-9.
- No 870 **Linda Askenäs:** The Roles of IT - Studies of Organising when Implementing and Using Enterprise Systems, 2004, ISBN 91-7373-936-7.
- No 874 **Annika Flycht-Eriksson:** Design and Use of Ontologies in Information-Providing Dialogue Systems, 2004, ISBN 91-7373-947-2.
- No 873 **Peter Bunus:** Debugging Techniques for Equation-Based Languages, 2004, ISBN 91-7373-941-3.
- No 876 **Jonas Mellin:** Resource-Predictable and Efficient Monitoring of Events, 2004, ISBN 91-7373-956-1.
- No 883 **Magnus Bång:** Computing at the Speed of Paper: Ubiquitous Computing Environments for Healthcare Professionals, 2004, ISBN 91-7373-971-5
- No 882 **Robert Eklund:** Disfluency in Swedish human-human and human-machine travel booking dialogues, 2004, ISBN 91-7373-966-9.
- No 887 **Anders Lindström:** English and other Foreign Linguistic Elements in Spoken Swedish. Studies of Productive Processes and their Modelling using Finite-State Tools, 2004, ISBN 91-7373-981-2.
- No 889 **Zhiping Wang:** Capacity-Constrained Production-inventory systems - Modelling and Analysis in both a traditional and an e-business context, 2004, ISBN 91-85295-08-6.
- No 893 **Pernilla Qvarfordt:** Eyes on Multimodal Interaction, 2004, ISBN 91-85295-30-2.
- No 910 **Magnus Kald:** In the Borderland between Strategy and Management Control - Theoretical Framework and Empirical Evidence, 2004, ISBN 91-85295-82-5.
- No 918 **Jonas Lundberg:** Shaping Electronic News: Genre Perspectives on Interaction Design, 2004, ISBN 91-85297-14-3.
- No 900 **Mattias Arvola:** Shades of use: The dynamics of interaction design for sociable use, 2004, ISBN 91-85295-42-6.
- No 920 **Luis Alejandro Cortés:** Verification and Scheduling Techniques for Real-Time Embedded Systems, 2004, ISBN 91-85297-21-6.
- No 929 **Diana Szentivanyi:** Performance Studies of Fault-Tolerant Middleware, 2005, ISBN 91-85297-58-5.

- No 933 **Mikael Cäker:** Management Accounting as Constructing and Opposing Customer Focus: Three Case Studies on Management Accounting and Customer Relations, 2005, ISBN 91-85297-64-X.
- No 937 **Jonas Kvarnström:** TALplanner and Other Extensions to Temporal Action Logic, 2005, ISBN 91-85297-75-5.
- No 938 **Bourhane Kadmiry:** Fuzzy Gain-Scheduled Visual Servoing for Unmanned Helicopter, 2005, ISBN 91-85297-76-3.
- No 945 **Gert Jervan:** Hybrid Built-In Self-Test and Test Generation Techniques for Digital Systems, 2005, ISBN: 91-85297-97-6.
- No 946 **Anders Arpteg:** Intelligent Semi-Structured Information Extraction, 2005, ISBN 91-85297-98-4.
- No 947 **Ola Angelsmark:** Constructing Algorithms for Constraint Satisfaction and Related Problems - Methods and Applications, 2005, ISBN 91-85297-99-2.
- No 963 **Calin Curescu:** Utility-based Optimisation of Resource Allocation for Wireless Networks, 2005, ISBN 91-85457-07-8.
- No 972 **Björn Johansson:** Joint Control in Dynamic Situations, 2005, ISBN 91-85457-31-0.
- No 974 **Dan Lawesson:** An Approach to Diagnosability Analysis for Interacting Finite State Systems, 2005, ISBN 91-85457-39-6.
- No 979 **Claudiu Duma:** Security and Trust Mechanisms for Groups in Distributed Services, 2005, ISBN 91-85457-54-X.
- No 983 **Sorin Manolache:** Analysis and Optimisation of Real-Time Systems with Stochastic Behaviour, 2005, ISBN 91-85457-60-4.
- No 986 **Yuxiao Zhao:** Standards-Based Application Integration for Business-to-Business Communications, 2005, ISBN 91-85457-66-3.
- No 1004 **Patrik Haslum:** Admissible Heuristics for Automated Planning, 2006, ISBN 91-85497-28-2.
- No 1005 **Aleksandra Tešanovic:** Developing Reusable and Reconfigurable Real-Time Software using Aspects and Components, 2006, ISBN 91-85497-29-0.
- No 1008 **David Dinka:** Role, Identity and Work: Extending the design and development agenda, 2006, ISBN 91-85497-42-8.
- No 1009 **Iakov Nakhimovski:** Contributions to the Modeling and Simulation of Mechanical Systems with Detailed Contact Analysis, 2006, ISBN 91-85497-43-X.
- No 1013 **Wilhelm Dahllöf:** Exact Algorithms for Exact Satisfiability Problems, 2006, ISBN 91-85523-97-6.
- No 1016 **Levon Saldamli:** PDEModelica - A High-Level Language for Modeling with Partial Differential Equations, 2006, ISBN 91-85523-84-4.
- No 1017 **Daniel Karlsson:** Verification of Component-based Embedded System Designs, 2006, ISBN 91-85523-79-8.
- No 1018 **Ioan Chisalita:** Communication and Networking Techniques for Traffic Safety Systems, 2006, ISBN 91-85523-77-1.
- No 1019 **Tarja Susi:** The Puzzle of Social Activity - The Significance of Tools in Cognition and Cooperation, 2006, ISBN 91-85523-71-2.
- No 1021 **Andrzej Bednarski:** Integrated Optimal Code Generation for Digital Signal Processors, 2006, ISBN 91-85523-69-0.
- No 1022 **Peter Aronsson:** Automatic Parallelization of Equation-Based Simulation Programs, 2006, ISBN 91-85523-68-2.
- No 1030 **Robert Nilsson:** A Mutation-based Framework for Automated Testing of Timeliness, 2006, ISBN 91-85523-35-6.
- No 1034 **Jon Edvardsson:** Techniques for Automatic Generation of Tests from Programs and Specifications, 2006, ISBN 91-85523-31-3.
- No 1035 **Vaida Jakoniene:** Integration of Biological Data, 2006, ISBN 91-85523-28-3.
- No 1045 **Genevieve Gorrell:** Generalized Hebbian Algorithms for Dimensionality Reduction in Natural Language Processing, 2006, ISBN 91-85643-88-2.
- No 1051 **Yu-Hsing Huang:** Having a New Pair of Glasses - Applying Systemic Accident Models on Road Safety, 2006, ISBN 91-85643-64-5.
- No 1054 **Åsa Hedenskog:** Perceive those things which cannot be seen - A Cognitive Systems Engineering perspective on requirements management, 2006, ISBN 91-85643-57-2.
- No 1061 **Cécile Åberg:** An Evaluation Platform for Semantic Web Technology, 2007, ISBN 91-85643-31-9.
- No 1073 **Mats Grindal:** Handling Combinatorial Explosion in Software Testing, 2007, ISBN 978-91-85715-74-9.
- No 1075 **Almut Herzog:** Usable Security Policies for Runtime Environments, 2007, ISBN 978-91-85715-65-7.
- No 1079 **Magnus Wahlström:** Algorithms, measures, and upper bounds for Satisfiability and related problems, 2007, ISBN 978-91-85715-55-8.
- No 1083 **Jesper Andersson:** Dynamic Software Architectures, 2007, ISBN 978-91-85715-46-6.
- No 1086 **Ulf Johansson:** Obtaining Accurate and Comprehensive Data Mining Models - An Evolutionary Approach, 2007, ISBN 978-91-85715-34-3.
- No 1089 **Traian Pop:** Analysis and Optimisation of Distributed Embedded Systems with Heterogeneous Scheduling Policies, 2007, ISBN 978-91-85715-27-5.
- No 1091 **Gustav Nordh:** Complexity Dichotomies for CSP-related Problems, 2007, ISBN 978-91-85715-20-6.
- No 1106 **Per Ola Kristensson:** Discrete and Continuous Shape Writing for Text Entry and Control, 2007, ISBN 978-91-85831-77-7.
- No 1110 **He Tan:** Aligning Biomedical Ontologies, 2007, ISBN 978-91-85831-56-2.
- No 1112 **Jessica Lindblom:** Minding the body - Interacting socially through embodied action, 2007, ISBN 978-91-85831-48-7.
- No 1113 **Pontus Wärnestål:** Dialogue Behavior Management in Conversational Recommender Systems, 2007, ISBN 978-91-85831-47-0.
- No 1120 **Thomas Gustafsson:** Management of Real-Time Data Consistency and Transient Overloads in Embedded Systems, 2007, ISBN 978-91-85831-33-3.
- No 1127 **Alexandru Andrei:** Energy Efficient and Predictable Design of Real-time Embedded Systems, 2007, ISBN 978-91-85831-06-7.
- No 1139 **Per Wikberg:** Eliciting Knowledge from Experts in Modeling of Complex Systems: Managing Variation and Interactions, 2007, ISBN 978-91-85895-66-3.
- No 1143 **Mehdi Amirijoo:** QoS Control of Real-Time Data Services under Uncertain Workload, 2007, ISBN 978-91-85895-49-6.
- No 1150 **Sanny Syberfeldt:** Optimistic Replication with Forward Conflict Resolution in Distributed Real-Time Databases, 2007, ISBN 978-91-85895-27-4.
- No 1155 **Beatrice Alenljung:** Envisioning a Future Decision Support System for Requirements Engineering - A Holistic and Human-centred Perspective, 2008, ISBN 978-91-85895-11-3.

- No 1156 **Artur Wilk:** Types for XML with Application to Xcerpt, 2008, ISBN 978-91-85895-08-3.
- No 1183 **Adrian Pop:** Integrated Model-Driven Development Environments for Equation-Based Object-Oriented Languages, 2008, ISBN 978-91-7393-895-2.
- No 1185 **Jörgen Skågeby:** Gifting Technologies - Ethnographic Studies of End-users and Social Media Sharing, 2008, ISBN 978-91-7393-892-1.
- No 1187 **Imad-Eldin Ali Abugessaisa:** Analytical tools and information-sharing methods supporting road safety organizations, 2008, ISBN 978-91-7393-887-7.
- No 1204 **H. Joe Steinhauer:** A Representation Scheme for Description and Reconstruction of Object Configurations Based on Qualitative Relations, 2008, ISBN 978-91-7393-823-5.
- No 1222 **Anders Larsson:** Test Optimization for Core-based System-on-Chip, 2008, ISBN 978-91-7393-768-9.
- No 1238 **Andreas Borg:** Processes and Models for Capacity Requirements in Telecommunication Systems, 2009, ISBN 978-91-7393-700-9.
- No 1240 **Fredrik Heintz:** DyKnow: A Stream-Based Knowledge Processing Middleware Framework, 2009, ISBN 978-91-7393-696-5.
- No 1241 **Birgitta Lindström:** Testability of Dynamic Real-Time Systems, 2009, ISBN 978-91-7393-695-8.
- No 1244 **Eva Blomqvist:** Semi-automatic Ontology Construction based on Patterns, 2009, ISBN 978-91-7393-683-5.
- No 1249 **Rogier Woltjer:** Functional Modeling of Constraint Management in Aviation Safety and Command and Control, 2009, ISBN 978-91-7393-659-0.
- No 1260 **Gianpaolo Conte:** Vision-Based Localization and Guidance for Unmanned Aerial Vehicles, 2009, ISBN 978-91-7393-603-3.
- No 1262 **AnnMarie Ericsson:** Enabling Tool Support for Formal Analysis of ECA Rules, 2009, ISBN 978-91-7393-598-2.
- No 1266 **Jiri Trnka:** Exploring Tactical Command and Control: A Role-Playing Simulation Approach, 2009, ISBN 978-91-7393-571-5.
- No 1268 **Bahlol Rahimi:** Supporting Collaborative Work through ICT - How End-users Think of and Adopt Integrated Health Information Systems, 2009, ISBN 978-91-7393-550-0.
- No 1274 **Fredrik Kuivinen:** Algorithms and Hardness Results for Some Valued CSPs, 2009, ISBN 978-91-7393-525-8.
- No 1281 **Gunnar Mathiason:** Virtual Full Replication for Scalable Distributed Real-Time Databases, 2009, ISBN 978-91-7393-503-6.
- No 1290 **Viacheslav Izosimov:** Scheduling and Optimization of Fault-Tolerant Distributed Embedded Systems, 2009, ISBN 978-91-7393-482-4.
- No 1294 **Johan Thapper:** Aspects of a Constraint Optimisation Problem, 2010, ISBN 978-91-7393-464-0.
- No 1306 **Susanna Nilsson:** Augmentation in the Wild: User Centered Development and Evaluation of Augmented Reality Applications, 2010, ISBN 978-91-7393-416-9.
- No 1313 **Christer Thörn:** On the Quality of Feature Models, 2010, ISBN 978-91-7393-394-0.
- No 1321 **Zhiyuan He:** Temperature Aware and Defect-Probability Driven Test Scheduling for System-on-Chip, 2010, ISBN 978-91-7393-378-0.
- No 1333 **David Broman:** Meta-Languages and Semantics for Equation-Based Modeling and Simulation, 2010, ISBN 978-91-7393-335-3.
- No 1337 **Alexander Siemers:** Contributions to Modelling and Visualisation of Multibody Systems Simulations with Detailed Contact Analysis, 2010, ISBN 978-91-7393-317-9.
- No 1354 **Mikael Asplund:** Disconnected Discoveries: Availability Studies in Partitioned Networks, 2010, ISBN 978-91-7393-278-3.
- No 1359 **Jana Rambusch:** Mind Games Extended: Understanding Gameplay as Situated Activity, 2010, ISBN 978-91-7393-252-3.
- No 1373 **Sonia Sangari:** Head Movement Correlates to Focus Assignment in Swedish, 2011, ISBN 978-91-7393-154-0.
- No 1374 **Jan-Erik Källhammer:** Using False Alarms when Developing Automotive Active Safety Systems, 2011, ISBN 978-91-7393-153-3.
- No 1375 **Mattias Eriksson:** Integrated Code Generation, 2011, ISBN 978-91-7393-147-2.
- No 1381 **Ola Leifler:** Affordances and Constraints of Intelligent Decision Support for Military Command and Control - Three Case Studies of Support Systems, 2011, ISBN 978-91-7393-133-5.
- No 1386 **Soheil Samii:** Quality-Driven Synthesis and Optimization of Embedded Control Systems, 2011, ISBN 978-91-7393-102-1.
- No 1419 **Erik Kuiper:** Geographic Routing in Intermittently-connected Mobile Ad Hoc Networks: Algorithms and Performance Models, 2012, ISBN 978-91-7519-981-8.
- No 1451 **Sara Stymne:** Text Harmonization Strategies for Phrase-Based Statistical Machine Translation, 2012, ISBN 978-91-7519-887-3.
- No 1455 **Alberto Montebelli:** Modeling the Role of Energy Management in Embodied Cognition, 2012, ISBN 978-91-7519-882-8.
- No 1465 **Mohammad Saifullah:** Biologically-Based Interactive Neural Network Models for Visual Attention and Object Recognition, 2012, ISBN 978-91-7519-838-5.
- No 1490 **Tomas Bengtsson:** Testing and Logic Optimization Techniques for Systems on Chip, 2012, ISBN 978-91-7519-742-5.
- No 1481 **David Byers:** Improving Software Security by Preventing Known Vulnerabilities, 2012, ISBN 978-91-7519-784-5.
- No 1496 **Tommy Färnqvist:** Exploiting Structure in CSP-related Problems, 2013, ISBN 978-91-7519-711-1.
- No 1503 **John Wilander:** Contributions to Specification, Implementation, and Execution of Secure Software, 2013, ISBN 978-91-7519-681-7.
- No 1506 **Magnus Ingmarsson:** Creating and Enabling the Useful Service Discovery Experience, 2013, ISBN 978-91-7519-662-6.
- No 1547 **Wladimir Schamai:** Model-Based Verification of Dynamic System Behavior against Requirements: Method, Language, and Tool, 2013, ISBN 978-91-7519-505-6.
- No 1551 **Henrik Svensson:** Simulations, 2013, ISBN 978-91-7519-491-2.
- No 1559 **Sergiu Rafiliu:** Stability of Adaptive Distributed Real-Time Systems with Dynamic Resource Management, 2013, ISBN 978-91-7519-471-4.
- No 1581 **Usman Dastgeer:** Performance-aware Component Composition for GPU-based Systems, 2014, ISBN 978-91-7519-383-0.
- No 1602 **Cai Li:** Reinforcement Learning of Locomotion based on Central Pattern Generators, 2014, ISBN 978-91-7519-313-7.
- No 1652 **Roland Samlaus:** An Integrated Development Environment with Enhanced Domain-Specific

- Interactive Model Validation, 2015, ISBN 978-91-7519-090-7.
- No 1663 **Hannes Uppman:** On Some Combinatorial Optimization Problems: Algorithms and Complexity, 2015, ISBN 978-91-7519-072-3.
- No 1664 **Martin Sjölund:** Tools and Methods for Analysis, Debugging, and Performance Improvement of Equation-Based Models, 2015, ISBN 978-91-7519-071-6.
- No 1666 **Kristian Stavåker:** Contributions to Simulation of Modelica Models on Data-Parallel Multi-Core Architectures, 2015, ISBN 978-91-7519-068-6.
- No 1680 **Adrian Lifa:** Hardware/ Software Codesign of Embedded Systems with Reconfigurable and Heterogeneous Platforms, 2015, ISBN 978-91-7519-040-2.
- No 1685 **Bogdan Tanasa:** Timing Analysis of Distributed Embedded Systems with Stochastic Workload and Reliability Constraints, 2015, ISBN 978-91-7519-022-8.
- No 1691 **Håkan Warnquist:** Troubleshooting Trucks – Automated Planning and Diagnosis, 2015, ISBN 978-91-7685-993-3.
- No 1702 **Nima Aghae:** Thermal Issues in Testing of Advanced Systems on Chip, 2015, ISBN 978-91-7685-949-0.
- No 1715 **Maria Vasilevskaya:** Security in Embedded Systems: A Model-Based Approach with Risk Metrics, 2015, ISBN 978-91-7685-917-9.
- No 1729 **Ke Jiang:** Security-Driven Design of Real-Time Embedded System, 2016, ISBN 978-91-7685-884-4.
- No 1733 **Victor Lagerkvist:** Strong Partial Clones and the Complexity of Constraint Satisfaction Problems: Limitations and Applications, 2016, ISBN 978-91-7685-856-1.
- No 1734 **Chandan Roy:** An Informed System Development Approach to Tropical Cyclone Track and Intensity Forecasting, 2016, ISBN 978-91-7685-854-7.
- No 1746 **Amir Aminifar:** Analysis, Design, and Optimization of Embedded Control Systems, 2016, ISBN 978-91-7685-826-4.
- No 1747 **Ekhiotz Vergara:** Energy Modelling and Fairness for Efficient Mobile Communication, 2016, ISBN 978-91-7685-822-6.
- No 1748 **Dag Sonntag:** Chain Graphs – Interpretations, Expressiveness and Learning Algorithms, 2016, ISBN 978-91-7685-818-9.
- No 1768 **Anna Vapen:** Web Authentication using Third-Parties in Untrusted Environments, 2016, ISBN 978-91-7685-753-3.
- No 1778 **Magnus Jandinger:** On a Need to Know Basis: A Conceptual and Methodological Framework for Modelling and Analysis of Information Demand in an Enterprise Context, 2016, ISBN 978-91-7685-713-7.
- No 1798 **Rahul Hiran:** Collaborative Network Security: Targeting Wide-area Routing and Edge-network Attacks, 2016, ISBN 978-91-7685-662-8.

Linköping Studies in Arts and Science

- No 504 **Ing-Marie Jonsson:** Social and Emotional Characteristics of Speech-based In-Vehicle Information Systems: Impact on Attitude and Driving Behaviour, 2009, ISBN 978-91-7393-478-7.
- No 586 **Fabian Segelström:** Stakeholder Engagement for Service Design: How service designers identify and communicate insights, 2013, ISBN 978-91-7519-554-4.

- No 618 **Johan Blomkvist:** Representing Future Situations of Service: Prototyping in Service Design, 2014, ISBN 978-91-7519-343-4.
- No 620 **Marcus Mast:** Human-Robot Interaction for Semi-Autonomous Assistive Robots, 2014, ISBN 978-91-7519-319-9.
- No 677 **Peter Berggren:** Assessing Shared Strategic Understanding, 2016, ISBN 978-91-7685-786-1.
- No 695 **Mattias Forsblad:** Distributed cognition in home environments: The prospective memory and cognitive practices of older adults, 2016, ISBN 978-91-7685-686-4.

Linköping Studies in Statistics

- No 9 **Davood Shahsavani:** Computer Experiments Designed to Explore and Approximate Complex Deterministic Models, 2008, ISBN 978-91-7393-976-8.
- No 10 **Karl Wahlin:** Roadmap for Trend Detection and Assessment of Data Quality, 2008, ISBN 978-91-7393-792-4.
- No 11 **Oleg Sysoev:** Monotonic regression for large multivariate datasets, 2010, ISBN 978-91-7393-412-1.
- No 13 **Agné Burauskaite-Harju:** Characterizing Temporal Change and Inter-Site Correlations in Daily and Sub-daily Precipitation Extremes, 2011, ISBN 978-91-7393-110-6.

Linköping Studies in Information Science

- No 1 **Karin Axelsson:** Metodisk systemstrukturerings- att skapa samstämmighet mellan informationssystem-arkitektur och verksamhet, 1998, ISBN-9172-19-296-8.
- No 2 **Stefan Cronholm:** Metodverktyg och användbarhet - en studie av datorstödd metodbaserad systemutveckling, 1998, ISBN-9172-19-299-2.
- No 3 **Anders Avdic:** Användare och utvecklare - om anveckling med kalkylprogram, 1999, ISBN-91-7219-606-8.
- No 4 **Owen Eriksson:** Kommunikationskvalitet hos informationssystem och affärsprocesser, 2000, ISBN 91-7219-811-7.
- No 5 **Mikael Lind:** Från system till process - kriterier för processbestämning vid verksamhetsanalys, 2001, ISBN 91-7373-067-X.
- No 6 **Ulf Melin:** Koordination och informationssystem i företag och nätverk, 2002, ISBN 91-7373-278-8.
- No 7 **Pär J. Ågerfalk:** Information Systems Actability - Understanding Information Technology as a Tool for Business Action and Communication, 2003, ISBN 91-7373-628-7.
- No 8 **Ulf Seigerroth:** Att förstå och förändra systemutvecklingsverksamheter - en taxonomi för metautveckling, 2003, ISBN 91-7373-736-4.
- No 9 **Karin Hedström:** Spår av datoriseringens värden – Effekter av IT i äldreomsorg, 2004, ISBN 91-7373-963-4.
- No 10 **Ewa Braf:** Knowledge Demanded for Action - Studies on Knowledge Mediation in Organisations, 2004, ISBN 91-85295-47-7.
- No 11 **Fredrik Karlsson:** Method Configuration method and computerized tool support, 2005, ISBN 91-85297-48-8.
- No 12 **Malin Nordström:** Styrbar systemförvaltning - Att organisera systemförvaltningsverksamhet med hjälp av effektiva förvaltningsobjekt, 2005, ISBN 91-85297-60-7.

- No 13   **Stefan Holgersson:** Yrke: POLIS - Yrkeskunskap, motivation, IT-system och andra förutsättningar för polisarbete, 2005, ISBN 91-85299-43-X.
- No 14   **Benneth   Christiansson,   Marie-Therese Christiansson:** Mötet mellan process och komponent - mot ett ramverk för en verksamhetsnära kravspecifikation vid anskaffning av komponent-baserade informationssystem, 2006, ISBN 91-85643-22-X.