University of Luxembourg
Faculty of Science, Technology and Communication

Doctoral Thesis

Network-Enabled Capability:

Modelling and Measuremnt

Author:

David Goergen

Supervisor:

Prof. Dr. Thomas Engel
Prof. Dr. Reinhard Posch
Dr. Habil. Radu State

A thesis submitted in fulfilment of the requirements for the degree of Ph.D

in the

Communicative Systems Laboratory
Computer Science and Communications Research Unit

August 13, 2014

To Brigitte and Romain

Acknowledgements

This work has been realized in collaboration with the SECAN-Lab at the University of Luxembourg. First of all I want to thank my local advisers and supervisors, Prof. Dr. Thomas Engel and Dr. Habil. Radu State, for giving me the opportunity to be part of their team and the support during the period of my Ph.D.. Secondly I would like to thank Prof. Dr. Reinhard Posch, CIO of the Federal Republic of Austria, for his continuous support and advises over the whole period. A special thanks also goes to Carlo Simon who provided me with numerous advices and hours of productive discussions. Special thanks are due also to Dr. Vijay Gurbani, distinguished expert at Bell Laboratories, for his permanent advice and guidance on the Application Traffic Layer Optimization protocol related research.

I would like to thank every member of the SECAN-Lab team for the warm and excellent work environment. A special thank you to Dominic Dunlop for proofreading and reviewing all the scientific contributions and enabling the use of the Hadoop Cluster.

Finally I would like to thank my family and friends for their continuous support over the whole period. Their dedication and motivation were an essential encouragement.

Abstract

In public safety many actors work together to achieve a common goal. However, communication among these actors is not always possible. Most use their own communication system and infrastructure, which are not always compatible with other participants'. A goal of this research is to analyse the current situation in terms of the capabilities of the different systems and to propose a solution for the successful interoperability between the different actors in the public safety arena by evaluating both current and improved communication infrastructures with a given scenario using a network simulator. To achieve this goal we analyse the capabilities of broadband communication infrastructure to create a model that describes its capabilities. We also analyse and evaluate the requirements for a communication infrastructure that can handle the daily communication flows, as well as the exceptional traffic that occurs during crises. Our evaluation and analysis is performed on country-level data sets that collect data over a certain period of time.

Contents

List of Figures

List of Tables

Introduction

Communication plays an essential role in our modern connected world, whether in checking the latest news or posting the latest update via social media. During crisis situations however, communication among involved parties as well as with victims becomes crucial, as recent events such as hurricanes Katrina and Sandy or the flooding in central Europe in 2013 of have shown. Public safety agencies often rely on a dedicated communication infrastructure tailored to their needs. Communication among public safety agencies such as fire fighters, police and medical personnel is often handled through command and control centres that coordinate the different resources at their disposal. Direct interoperability between the different communication infrastructure is not always possible but can be essential if some parts of the system fails.

To improve interoperability we investigate the possible improvement of existing communication infrastructures by proposing to incorporate other means of communication, e.g. private cellular communication and other possible communication technologies, rendering the existing architecture more robust and reliable in the case of a major crisis. We focus only on major crises, as communication in smaller and minor crises is minimal because it is the daily routine of public safety agencies to deal with these. A major crisis may have a natural origin, such as an earthquake or flood, but it could also be a man-made event such as a plane crash or terrorist attack. A common factor in all these scenarios is the potential implication of a large number of victims. Consequentially, public safety organisations have developed mass casualty plans to deal with this kind of event in order to efficiently coordinate their available resources. These plans describe in detail which agencies need to be involved and how the agencies should interact with each other. The impact on the communication architecture however is not always tackled as the agencies rely mostly on their own dedicated architectures. These are often outdated and offer only limited services, such as one way communication between two partners. Compared to state-of-the-art mobile cellular networks that offer video and audio calls, conference calls and data services they definitely need improvement. To illustrate how these additional services can contribute to more efficient crisis management we evaluate them in such a

mass casualty scenario. We aim to develop a framework for crisis management coordinators which enables them to improve the decision-making process though training and simulation of possible events, as well as forensic analysis of past event and the lessons learned. A typical decision process is initiated by the detection of a major incident in within a monitored region. Upon detecting an incident the coordinator must take decisions and actions according to their previously-developed mass casualty plan. Each decision taken will have an impact on subsequent decisions, and so needs a thorough understanding of the situation at hand. As lives are at stake the decisions and actions are time-critical and therefore must be as accurate and timely as possible. This real-time analysis and processing gathers a large of amount of data. The collected data needs to be carefully analysed in order contribute and support the decision-making process. In order to take a decision, a crisis coordinator needs all the information possible about the occurring incident. We propose to support the gathering and aggregation of information by analysing the Call Data Records (CDRs) of mobile phone operator, as described in chapter 4. These CDRs are typically used by operators to ensure correct billing of their customers. Further analysis of a CDR corpus has allowed us to detect patterns in them and to correlate the observed patterns with possible events or incidents.

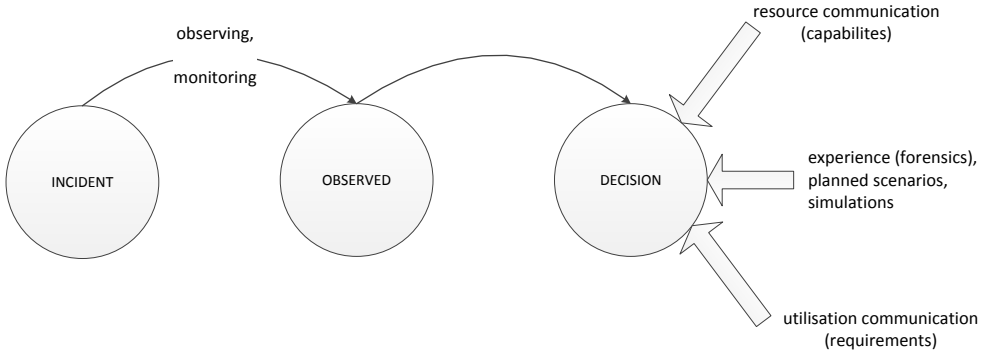


FIGURE 1: Situation awareness cycle

To further increase our confidence in the observed patterns we propose to aggregate these information with public and open information platforms. Due to our modern lifestyle, people share information with their "followers" to keep them updated about interesting

events. These message flows are often distributed before the initial incident report arrives at the crisis management centre. If these flows are carefully mined and analysed, they can represent an important source of information. Research in this area has proved to be helpful in analysing the situation at hand.

Once the crisis management centre has gathered all relevant information about the incident, it can begin to dispatch the public safety agencies. Since we focus on the communication flows among these agencies, we need to model the communication between them and the coordinators as well as among themselves. If we presume the involvement of private cellular networks we also need to take into account the daily traffic generated by their customers. Here again the CDR corpus can provide essential information on network utilisation as well as the mobility patterns of the connected users. The modelled population traffic and the traffic generated by the rescue services will represent the requirements that our communication infrastructure has to cope with. Knowing the requirements means that we also need knowledge of the network capabilities. These will represent the resources available in the communication architecture. With this information in hand the coordinators can compute the capability of their communication infrastructure.

This dissertation is structured as follows. Chapter 1 deals with alternative means of communication such as Twitter. In this chapter we explore the possible integration of Twitter into public safety communication processes and show the benefits of using this kind of information. However such systems also have some inherent drawbacks that should be mentioned. In chapter 2 we investigate how to retrieve network capability information of an established communication infrastructure as well as modelling and profiling the requirements of the communication flows over a given communication infrastructure. The exploration and analysis of the captured data is used to construct models that can be integrated into a simulation environment and serve as ground truth. Chapter 3 focuses on recent activities in overlay network implementation of communication infrastructures over Internet and their possible benefits to public safety communication. These new approaches show some interesting advantages when it come to the propagation of content. Since these systems are quite novel, there are some research questions that must be tackled such as their vulnerability to cyber-attacks and the implementation of regulation policies for such traffic. Analysing the data discussed in chapter 2 to create realistic models is explored in chapter 4. We analyse a country-level data set from a mobile phone operator in order to detect possible early signs of larger incidents in the communication flows. This data can also be used to create models of how the communication changes during a crisis and its impact on the whole communication infrastructure. Chapter 5 deals with the use of public data to create overlay maps of the underlying communication infrastructure. These can provide essential information to create a precise overview of

the network capabilities of a communication infrastructure. Finally chapter 6 concludes this thesis and provides some final remarks.

Chapter 1

Communication in an Emergency Situation

1.1 Monitoring social media services

People tend to share information with their peers in order to inform them about their current situation typically, by using social media services (e.g. sharing their latest news over Facebook or Twitter). This has become a huge part of our social life. During crises this behaviour is even more pronounced because it allows people to reassure their peers (followers and friends) about their well-being in a quick and easy manner. Recently social media services have been also used for another purpose during crises: that of informing oneself about the current evolution of the crisis. During crises people have shown that they are able to organise themselves in order to provide support to others. Examples are the flood in Germany in 2013 where people spontaneously organised sandbags to protect themselves from the flood or the distribution of goods for people cut off by the flood[1]. Such behaviour has increased over the last few years. From the point of view of research this is interesting, as people tend to prefer the use of social media over the use of traditional media (e.g. telephone). Crisis management can also profit from this kind of information as the population tends to share information with each other. This information could be mined and used during the decision process, as it provides near real-time information on the situation to crisis management centres. However an essential question is how trustworthy information shared over this new medium is, and how to filter the messages relevant in a crisis situation from huge amount of the general chatter. General opinion is currently split on whether crisis management officers should use this

[1] http://etzrodt.wordpress.com/2013/06/06/hochwasser-in-dresden-2013-und-die-rolle-sozialer-medien/

information or not. Social media has another inherent problem: the language and syntax used are very informal, meaning that traditional language processing techniques must be adapted to the situation at hand.In particular Twitter streams, known as tweets, are written in a condensed style as tweets are limited to 140 characters. Consequently the syntax employed by its users is highly informal and contains many abbreviations and grammatical errors. People tend to use the phonetic sound of letters and numbers to represent whole syllables of words. This makes traditional language processing techniques even more impractical. However 140 character limit allows information to be share quickly and concisely with followers, who can in turn share it more widely.

As people tend to follow others over social media, it is increasingly suggested that crisis management and public agencies should have official accounts to also inform the population as other media might not be reachable or be out of service. The authors in [1] show that an official account managed by people who know how to use social media has proven to be effective during and after a crisis, if the operators follow best practices. In [2] the author discusses the ethical implications of including social media in crisis management. The author supports his claims by illustrating the advantages and the disadvantages of such inclusion.

To extract and use relevant information from the general flow of information, operators need to filter out only the flows they are concerned with. Taking Twitter as an example, there are on average 241 million active users over one month and about 500 million tweets sent every day[2]. Analysing this amount of data manually is time-consuming and not practical during a crisis situation. In [3] the authors survey several techniques for event detection in Twitter streams and provide an overview of investigative techniques. They classify each technique based on three characteristics: the type of event, the method used to detect the event, and whether an event is new or not. For the determination of the type of event they further distinguish between *specified*, where some information about the event is already known (i.e. time, venue) and *unspecified* where no prior information about the event is available. The detection of an event is further divided into *unsupervised* and *supervised* techniques. Supervised detection techniques require a certain amount of tagged data, in our case Twitter streams, in order to learn which streams are important and which can be neglected. In the case of unsupervised learning, the algorithm tries to detect and infer patterns from the original input stream without prior knowledge. The last characteristic focuses on whether new events should be detected using knowledge from previously learned events in the live data streams or if previously undetected events should be discovered using historical data.

[2]Twitter usage statistics 01.02.2014: https://about.twitter.com/company

A crisis is commonly defined by three distinct parameters known as w^3 questions. The first is the actual event or incident which corresponds the question ***What*** *is actually happening?*. The next question focusses on the actual location: ***Where*** *is the incident?*. Finally the time of the event is established: ***When*** *did the event happen?*. Taking a deeper look at some of the techniques employed in event detection on Twitter streams involves analysing the tweets themselves, as they are the actual source of information we are interested in. This will help to answer the first question of what is actually happening. Natural language processing (NLP) techniques are most commonly used to extract relevant information from tweets. NLP uses language-specific features, which allow the algorithm to process it in a similar fashion to a human being, enabling the algorithm to infer the semantics of a sentence and assigning a meaning to it. We present several different techniques of NLP that have been used to investigate Twitter event and news flow detection before, during and after crises [4, 5]. Figure 1.1 below and Table 1.1 on page 12 give an overview of the topic of event detection and classification.

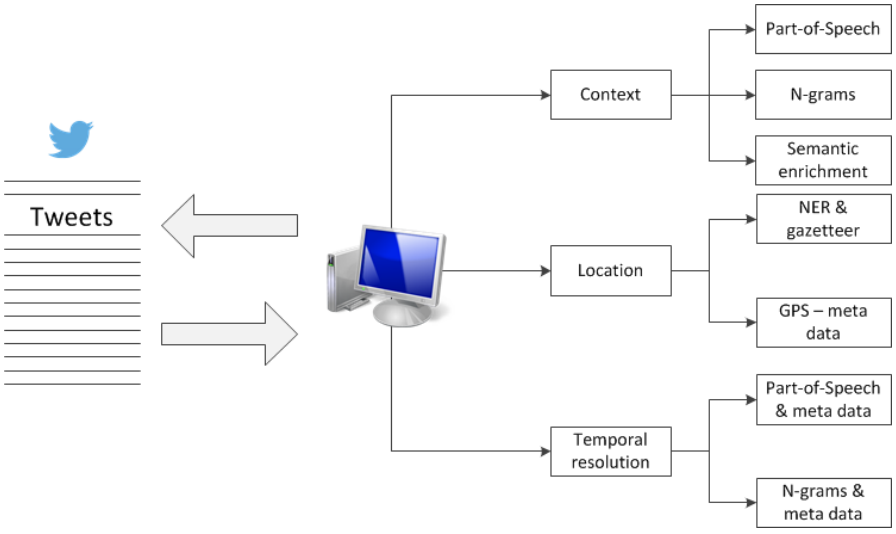


FIGURE 1.1: Twitter analysis

The method described in [4] focuses on classification of disaster by extracting features from tweets using n-grams. To determine the n-grams the authors pre-process the original tweets to remove all the stop words and punctuation. Subsequent analysis includes additional Twitter-specific features such as the presence and number of *hashtags* mentioned in a tweet, as well as the presence of *mentions* and the number of mention included in a tweet. Hashtags are a kind of keyword to which other participants of the ongoing discussion can refer to in order to be included in a search on the specified hashtag, thereby grouping all relevant tweets in the ongoing discussion. Hashtags are helpful during crises, as they can quickly filter out unwanted general chatter. They are most usefull when there is a common agreement on the establishment of a hashtag beforehand; however this is not often the case for crises, which are random in their appearance.

It is not guaranteed that messages using the hashtag related to a crisis will contain only information which crisis management officials and people involved find relevant, as the example in [6] shows. The authors tested their method by using a data set generated through the Twitter API. The data set contained tweets relating tpseveral different kinds of disasters distributed around the globe. Using worldwide data is difficult, as in general proposed methods focus only on a single language. For formal and well-written text, machine translation can go some way towards solving this issue but for sparse and malformed text such as tweets it must be adapted in order to produce useful results. The analysis separated the data set into a learning and a testing set using K-fold cross validation including a time-split parameter to guarantee that the learning phase did not include future knowledge on events not present in the learning time frame. The learning set was manually annotated using two criteria. Firstly a binary classifier: was the tweet related to a disaster?; and secondly a multi-class classifier: what kind of disaster was it? The annotation was performed via a crowd-sourced platform using a majority vote of the annotators to determine the classification. The resulting classifiers were tested and analysed using Support Vector Machines (SVMs) [7] to classify the test data sets. The results of the experiments showed that using n-grams only, a success rate of 50% was obtained for the binary classifier if the SVM started with 10% of the original data set. The additional Twitter specific features only brought significant improvement to the classification task if 50% of the data set was used during the learning process. Due to the nature of hashtags, namely that they are specific to a certain event, the classification process is not able to generalise them for use in other events. The authors explain that only general hashtags such as #CDCemergency improve the classification task when combined with the n-gram based method and only then over a sufficiently large learning set. Their results indicate that it is important to have a certain diversity of tweets that mention a certain location, as the classification process could falsely classify a tweet due to to limited information at the learning stage. Another topic of interest, also discussed in [4], is to be able to classify events that have yet to arise. To do this, the authors used their annotated data set and excluded one of the available categories of disasters, then evaluated whether the binary classifier was able to detect the omitted class based on the knowledge of other disaster types. Here again Twitter-specific features were included to analyse their impact on the classification. An observation from the previous experiment was again confirmed, namely that crises often feature disaster-specific hashtags, which are not very helpful in the categorisation process. Hashtags are helpful when a disaster of the same type occurs, as they already exist. Another observation is that tweets could successfully be categorised of as crisis related, even if the given type of crises was not featured in the learning process. This maked the monitoring Twitter for crisis event detection interesting, as we can learn from past history in order to detect new events in the live stream.

The authors of [5] follow another path. They focus on classifying tweets not based on the disaster type, but rather on the overall topic covered by the tweets, extracting relevant information to increase overall situation awareness. This classification of tweets allows them to improve general situation awareness such that unrelated tweets are discarded during a crisis by content-based filtering and extracting only the parts of the remaining tweets that are considered relevant. The authors' previous work established an ontology that further explains the different categories of topics covered [8]. They also rely on the effectiveness of crowd-sourcing for annotating their data set. For this purpose they used two distinct data set: The first collected the tweets during a tornado that severely damaged Joplin, MO in 2011, while the second contained tweets sent during hurricane Sandy which hit the United States in 2012. As their analysis focused on historical data they were able to extract the tweets relevant to each disaster by using the respective hashtags, #joplin for the tornado and #sandy and #nyc for Hurricane Sandy. Based on their previous work they attempted to determine if it is possible to detect a new kind of disaster by learning from previous ones. Once the annotation process had finished they proceeded to the extraction of relevant information fragments. A conditional random field algorithm [9], a machine learning technique proven successful in the extracting of information fragments, is used to label each word in a tweet marking the information fragment for further processing. These fragments are classified using a set of multi-label classifiers defined in [8]. The classification process is performed using the Waikato Environment for Knowledge Analysis (WEKA) [10] using naïve Bayes classifiers consisting of n-grams and Part-Of-Speech(POS) taggers. Experiments conducted on each data set showed good results when they were used individually. The main challenge however was to be able to use one data set and learn from a previous one. The experiments showed a decreasing detection rate when the Joplin data set was used for training and tested on the Sandy data set. The hit ratio, the ratio of detected examples deemed correctly tagged by human test subjects however remained stable and was not affected. The conclusion drawn are similar to those of [4]: some of the word used are event specific and cannot be transferred to other type of disaster, particularly when describing the event itself independent of the damage it caused. A final experiment showed that, if the system was trained completely on the Joplin set and only a small portion of the Sandy set was used in the learning process it could adapt to the situation, leading to improvement in the detection process.

The authors of [11] show that keyword-based filtering can be further enhanced using semantic expansion. The system developed by the authors begins searching for relevant tweets as soon as a new incident message is generated through a public incident announcement system. This initial message is dissected using Named Entity Recognition to extract location, person, organisation, etc. together with the incident at hand and

semantically expand the search parameters for related tweets. This semantic expansion algorithm takes into account whether the tweets link to a web page and further expands the search parameters of an incident by analysing the linked page. The expansion of the search parameter is added to the original parameters using a weighted facet-value approach for each additional parameter. The classification of the relevant tweets is then performed using a rule-based approach. The authors compare their approach using several data sets and show that it offers an improvement compared to purely keyword base approaches.

Location of an incident is another important feature with regard to crisis management, as it provides essential information about the area around the incident. Twitter has implemented a location information service, which is turned off by default. If this feature is turned on, the position from which a tweet was sent can easily be mapped. Analysis in [12] shows that only a small proportion of Twitter users use this optional service. However, recent studies such as [13] have shown that many users inadvertently share their current location, either by the tweet itself or in the metadata of attached documents(e.g. photos). Crisis management can profit from this information, as it can give accurate situation reports, raising the overall situation awareness for decision takers, even before first responders from public safety agencies are on site.

Named Entity Recognition(NER) algorithms are used in [14] and [15] to extract names of locations from tweets and georeference them using open gazetteers such as the GeoNames database[3] to retrieve their geographical coordinates. NER classifies each word of a sentence into a possible class of similar expression such as names of persons or organisations, values, temporal expression (e.g. today, tomorrow, evening). However, NER algorithms are domain-specific meaning that, in order to work well for the extraction for words within tweets for crisis management purposes, they need to be adapted to the specific language used. This can be difficult for tweets, as they often contain misspelling and informal speech. In [15] the authors leverage the use of a reduced data set from the GeoNames database. The authors argue that the data set can be reduced to inhabited locations only, ignoring places where users would normally not be found, so reducing the time needed to match a given location in the data set. In [16] the authors take into account tweets that reveal their position either by divulging their GPS coordinates in the Twitter metadata or by extracting the user account's profile location and querying its GPS coordinates using Google maps. They assume that the majority of the tweets posted about a major incident are sent by users who live in close proximity to the incident, making the location provided in the user profile accurate enough. This assumption may be correct for earthquakes for example but it may not be applicable to other crises.

[3]GeoNames Database: http://www.geonames.org

Another important step is to analyse the temporal indications in the tweet metadata. This is important if we want to use the data provided in real-time analysis. It should however be clear that the time indication only helps if combined with the location and an actual description of what is happening. In [16] the authors illustrate that, in case of an earthquake, by using distinct keyword-based queries, they are able to determine the location of the epicentre quite accurately. Secondly, by using the same procedure they are able to determine the approximate trajectory of a hurricane. Their method considers Twitter users to be sensory devices deployed on a large scale, although compared to more conventional sensor networks, they find that Twitter users are noisier due to their behaviour (e.g. some users might not share information because they are asleep or have no network connectivity). In order to compensate for this noisy behaviour the authors apply well-known techniques: Kalman filters [17] and particle filters [18], which have shown good results in location estimation in pervasive and ubiquitous computing.

The authors of [19] use Twitter streams to established a calendar of important events. While this is not necessarily related to crisis management they present a method to map the information contained in a tweet to extract the date and time of an event if abstract concepts such as "tomorrow" or "next Monday" are used. They use Part-Of-Speech (POS) tagging and Named Entity Recognition(NER) to extract the information fragments needed to construct their calendar. They leverage the use of an in-domain NER tagger as tweets often capitalise words to emphasize their importance, complicating exact segmentation for a classical NER tagger. They also use a rule-based temporal expression resolver to associate an exact date to the information contained in a tweet. Their classification model is based on latent variable extraction, which classifies the tweet to an appropriate category by relying solely on the information contained in the tweet itself. The authors compare their method with a supervised classification algorithm as a baseline, using the F_1 score of each method as a metric for comparison, and show that their method significantly outperforms the baseline.

authors	**year**	**monitored period**	**events**	**methods**
Karimi et al. [4]	2013	Dec 2010 - Nov 2012	Various catastrophes	WEKA based classification using n-grams and Part-of-Speech analysis
Imran et al. [5]	2013	2011 / 2012	Joplin tornado (2011) and Hurricane Sandy (2012)	SVM based classification using n-grams and Twitter features
Abel et al. [11]	2012	TREC corpus 2011	Multiple incidents	Named Entity Recognition and semantic expansion of message
Valkanas and Gunopulos [12]	2012	•	•	Geolocation extraction from tweets
MacEachren et al. [14]	2011	•	•	Location determination using customised Named Entity extractor
Valkanas et al. [15]	2013	•	•	Matching extracted location with gazetteer
Sakaki et al. [16]	2010	Aug-Oct 2009	Earthquakes in Japan	Spatio-temporal analysis using Kalman and particle filters
Ritter et al. [19]	2012	$\approx 10^8$ tweets up to Nov. 2011	Multiple events	Resolving temporal expression with accurate date/time information

TABLE 1.1: Overview of presented research

1.2 Dissemination of crisis-relevant information

Analysing and monitoring tweets in order to increase the situational awareness of a crisis is not the only application of Twitter that public safety agencies can use. Twitter is primarily a platform to share information about the current situation, often answering the question "What is happening right now?". Twitter is designed to let people quickly post and share status updates, and allow followers and friends to respond to them in a fast an convenient manner. In [20] the authors analyse the current utilisation of Twitter by different governmental bodies. @GOVsites, now renamed into @GOVworld, is a non-exhaustive list of governmental agencies with official Twitter accounts. The authors sampled a set of 60 accounts over a six-month period to determine the way in which these governmental bodies use Twitter. The purpose of their analysis was to determine whether governmental agencies tend to use Twitter for the sole purpose of information dissemination, or whether they also take advantage of the Twitter architecture to initiate an interactive dialogue between themselves and their followers. The authors therefore categorise tweets of their sample set applying the four typical models from public relations: press release, public information, two-way asymmetrical, and two-way symmetrical. This lets them determine whether a tweet is intended to simply inform people in general about a certain topic, or if instead there is some kind of dialogue between them and the followers. If the latter is the case they further distinguish between asymmetric and symmetric communication. In asymmetric communication the dialogue does not take place directly between the involved parties but rather by means of polls and surveys to gather opinion and provide feedback. Symmetric communication on the other hand is an ongoing dialogue between the involved parties. The outcome of the research is that governmental bodies primarily use their official Twitter accounts to share general information with the public. The authors noticed however that agencies also tend to use the two-way communication nature of Twitter in order to stay in contact with their followers. Public relations literature also suggests and favours the use of a more dialogue-based exchange. Both modes, public information sharing and two-way communication, can be of use in time of crisis.

When discussing the dissemination of information via Twitter and other social media, the reliability and trustworthiness of the information contained in the shared message becomes an issue. In Twitter there are two modes of sharing information: one can either post it on the public wall where anybody can read and comment it; or one can share it only with the people following your account. The latter case can be considered more trustworthy as generally people tend to befriend people they already know and oh whose trustworthiness hey have a positive opinion of. However, in this thesis we are discussing the possibility of using Twitter as a means of communication in times of

crisis. The authors of [21] discuss how willing people are to share information received via different media. They analyse whether people's tendency to share information depends on the medium through which the information is received or whether the message itself is important. They additionally analyse whether emotional content tends to increase the willingness to share information relative to informal public notification. They also discuss whether the established reputation of an organisation might be tarnished depending on medium via which a message was received. Their analysis is based on an experiment conducted with around 2,000 people, which consisted of displaying a message issued by an organisation and assessing how the test subjects reacted to it. The media used to evaluate the impact were a classic medium — here a newspaper article, a organisation's blog and their official Twitter account. A fourth option was a combination: a Twitter message which would lead the person to a blog entry containing further information. The results obtained confirmed some of the authors' hypotheses while others were disproved. An interesting observation was that the participants in the experiment were more willing to share information obtained through the classical medium than that obtained via blog or Twitter. This might be explained by the nature of the medium itself. While blogs and Twitter accounts are often managed by corporation employees, a newspaper is regarded as neutral and the message has no perceived bias. Another interesting observation was that negative opinions and comments were less prevalent if the message contained only general information when compared to an emotional message which expressed either an apology or sympathy. The experiment also showed that participants already actively using Twitter were more willing to share information compared to bloggers and people using neither service.

In [22], the behaviour of people was analysed in order to determine whether Twitter could be used in crisis scenarios, and how the population reacts to information shared over this medium. The authors were able to show that there is an increase in Twitter messages originated and forwarded when comparing the sample set of users before and after a crisis. They analysed a data set of tweets around the 2011 Tōhoku Earthquake and evaluated the change in behaviour of the users before and after the earthquake. The results obtained indicate that people using Twitter to share information will also do this during a crisis and may even increase the amount of information that they forward via Twitter, whereas automated systems which normally send a volume of messages almost come to a halt. They also analysed whether people tend to use forwarding mechanism after a crisis when they had not used them before. The results indicated that those who had not used forwarding mechanism before also did not use them to share information during or after a crisis. The results also showed that forwarded information was mainly information relayed from mass media and general users. An interesting observation was of how false rumours were handed. The authors showed that, although only a

few false rumours appeared, these were quite efficiently dealt with by the Twitter users themselves. The results showed that the amount of counter-information acting against the further spreading of false rumours outweighed the original number of rumours. This is particularly interesting when considering the overall reliability of information circulating on Twitter: people themselves take care of the false information spreading on the network.

The authors of [23] compared a tool which they developed to gather information related to emergency event from web feeds with the Emergency Situation Awareness (ESA) platform [24, 25], a system that collect similar information from Twitter. The developed tool gathers web feeds from various emergency services and organisations, focusing on information about crises such as cyclones, fires and others. The authors analysed both systems by monitoring a specific time-frame containing emergency events and compared the chronological occurrences of tweets with the information published through web feeds. The results indicated that tweets often occur before an actual web feed is published and the tweets often contain, despite being limited to 140 characters, relevant information for those following the tweets. Furthermore, tweets also contain images and links to further information. The paper concludes that Twitter, even though public, can be a useful addition to public emergency services for the dissemination and collection of information. The conclusion is that Twitter can be considered a good source of information, but one should always consider using it in combination with other media to collect data or to disseminate information.

In [26] the authors analyse the willingness of people to share information focusing particularly on sharing information over Twitter. To determine the willingness to disseminate information, the authors conducted several experiment to analyse the behaviour of the test candidates. They argue that people are more willing to share information if they find themselves in close proximity to a crisis centre than if they are further away from same situation. They also assume that people are more willing to share information if they are themselves involved than if they simply imagine being someone else in the same situation. People tend to interpret pieces of information and associate their personal feelings before sharing these pieces. Most crises are associated with negative feelings, as lives are in danger leading to sorrow and despair among victims and people involved. The authors further make the assumption that messages with negative feelings are more often shared than messages with positive feelings such as relief and happiness. Like [21] they also tested the impact of the source of information on willingness to share, but using sample messages from individuals and mass media. The experiment tested this assumption by asking the subjects how likely they would share a message received via social media. The experiment revealed no statistically relevant dependence on the source of the information. Some of the authors' assumption were validated, however. People are more willing to share information if they are personally involved in a crisis situation

than if they imagine someone else being involved. The assumption that emotionally negative information is disseminated more than positive messages was also confirmed. The distance assumption was only partially confirmed, resulting in a suggestion that further investigations should be carried out future work.

1.3 Implementations

The Emergency Situation Awareness[4] [24, 25] platform developed at CSIRO in Australia offers a complete system architecture for monitoring and capturing the continuous flows of tweets using both the Twitter search and a streaming API. The platform has a modular design, where each module is responsible for a distinct task. One module handles the capturing of tweets while another processes the captured tweets to detect possible candidate incidents showing a sudden increase or burst in messages occurring. A third module detects these bursts by unigram analysis and word occurrence of the tweets to create a statistical model of the event. When an official deems the incident worthy of further investigation, another module will cluster all relevant tweets using common labels. The continuous flow of tweets is then further filtered based on the selected event using Support Vector Machines to remove tweets containing no relevant information about the incident under investigation. The platform also embodies an incident visualisation tool, which extracts the physical location of the emitted tweet, either by using its geo-tagged metadata or by using the author's profile location. By using Name Entity Recognition they can also map organisation, places on a map, and time and date to allow a spatio-temporal analysis of the relevant tweets. The platform also provides a module which allows the forensic analysis of captured tweets.

The Senseplace2[5] [14, 27] platform developed at PennState University allows its user to search for tweets using keywords as an initial filter to narrow the search. It queries the Twitter feed using its API and stores the retrieved tweets in a database for further processing. The initial query can then be further refined by using a time line control, which offers a slider based bar to specify the time period that the user wants to analyse. A heat bar provides further decision support by highlighting periods of intensified messages flows. Named Entity Recognition is also used to extract location and other information. Senseplace2 also offers a map on which the tweets most relevant according to the search parameters are visualised by frequency of occurrence and by location. The platform is designed to increase situational awareness by allowing a quick exploration of relevant data through an easy to use interface. The list of retrieved tweets can be filtered by relevance, time or location. It also provides a word cloud of the most frequent words in

[4] https://esa.csiro.au/
[5] http://www.geovista.psu.edu/SensePlace2/

the tweets. This allows further narrowing of the search parameter and quick exploration of data.

The Twitcident[6] [11, 28] platform developed at TU Delft in the Netherlands processes messages obtained via the P2000 emergency broadcasting system. This network[7] is exclusively deployed in the Netherlands. It features only one-way communication among pagers using Motorola's FLEX protocol. The messages are sent in a text form similar to SMS. Due to the lack of encryption, the messages can be captured by any receiver implementing the protocol. When a new incident is reported through the P2000 network, the Twitcident platform just creates a new incident profile. This profile is composed of essential information extracted from the original message such as location, incident type, etc. These pieces of information are stored in a facet-value based scheme where the facet is a key that represents the type of information stored and the value represents the extracted information. Each facet-value pair is also associated with a weight representing the importance of the variable. Having generated the profile, the platform then begins to semantically expand the initial features set by analysing Twitter streams and extracting further parameters through Named Entity Recognition, enhancing and narrowing the underlying search of relevant tweets. Associated links in tweets are also processed in a similar fashion to further enhance the search algorithm. The method shows a significant improvement compared to purely key-word based system. The platform has been used on several occasions and proved its usefulness in real crisis situations [29, 30].

[6] http://twitcident.com/

[7] Live monitor of the P2000 network: http://monitor.livep2000.nl/

Chapter 2

Simulation, Emulation and Measurement of Broadband Capabilities

2.1 Overview

In public safety many actors work together in times of crisis to achieve a common goal and to save lives. However communication between these actors is not always possible or it could be that the crisis has rendered some of the systems useless. Most public safety organisations use their own communication system and infrastructure, which offers limited to no compatibility with other actors in the field. Inter-agency communication is often handled by a command and control centre were the managing entity gathers all relevant information before executing actions. We will not focus on how this communication could be optimised but rather the incorporation of the different communication infrastructures involved into a large heterogeneous network of networks. This allows seamless communication between every actor involved, independent of its organisation allegiance. Analysing current communication infrastructures in terms of the capabilities of the different systems is therefore a preliminary step in understanding how the different actors involved in public safety operate. This will enable us to envision a solution which provides interoperability between different actors' infrastructures in public safety. Envisioning such a large network can be quite cost intensive if done without prior testing and evaluation. We therefore opt instead to simulate heterogeneous environment in order to illustrate the benefits of such a system. This allows us to propose and test changes on the current model of the infrastructure without touching the physical systems already in place.

As mentioned above we first characterise the people involved in a large scale event and the requirements they have in order to fulfil their mission objectives. This also involves analysing the communication infrastructure currently in place and determining if there are potential weak points. Generally speaking the most commonly involved parties in a crisis are public safety agencies such as police, fire-fighters and medical personnel. Depending on the scale of the event, other parties such as military or civil protection might also be involved. Private companies might also offer expertise or personnel to help to cope with certain situations. All these actors have different needs and requirements, which are reflected in the use of agency-specific communication infrastructures. Public safety also often involves the commitment of voluntary personnel such as rural fire brigades and medical personnel provided by the population in the area. Another important factor that should be taken into account is cross-border cooperation between different countries. This is especially important when a crisis occurs near a national border.

Governments' own communication infrastructures are not the only ones to take into consideration when creating a heterogeneous architecture. Private network operators, such as mobile phone providers, also maintain communication infrastructures, which could play a crucial role in times of crisis. Their priority inevitably lies with their customers, but an intelligent prioritisation scheme for using the operator's network for crisis situations could be available to governmental and non-governmental agencies. Combining different communication infrastructures also raises other issues such as information leakage to unauthorized personnel or privacy and security concerns. Consequently strong legal policies need to be in place to guarantee correct information handling and exchange.

2.2 Vision

Analysing many different communication infrastructures can be time and cost intensive. We therefore opt for a simulation environment in which potential users can test their communication infrastructure and simulate any given condition related to public safety. This allows for a thorough analysis of the infrastructure in different simulated incident scenarios without taking the risk of harming the workflow of any public safety agency during a real crisis. We leverage the use of a heterogeneous communication infrastructure consisting of several independent, often overlapping, networks that are merged together to form a large network of networks which should enhance interoperability and reliability. Network managers and operational end users will also benefit from such a simulation environment, because it enabled them to have a global view of their current communication infrastructures and it could be an aid in developing plans and scenarios which could be

tested in advance of a real crisis arising. Future versions of the simulation environment could incorporate emulation of real physical devices that would deliver real-time measurement of the communication infrastructure which would further enhance confidence in the result produced. In addition, using the simulation during actual crises could support the decision-making process, as it would allow testing decisions and their possible outcomes and impacts.

This chapter reviews three main areas related to crisis management. Firstly, the analysis and measurement of network capabilities and communication infrastructures is discussed in section 2.3. In section 2.3.1 we discuss how to measure the capabilities of existing network architectures and the lessons that can be learned. Section 2.3.2 deals with how to model users and their behaviour. This provides a concrete view of the requirements for the network infrastructure, and models of how users behave in different situations. To the best of the authors knowledge, there has been little or no research on traffic models and the simulation of heterogeneous public safety networks. Therefore section 2.3.1 and 2.3.2 are used as a foundation for section 2.4, which deals with how to utilise network capabilities and user behaviour data in simulations and/or emulations of communication infrastructures. The capabilities serve as a reference model of the actual communication infrastructure, while the user and behaviour analysis is used to simulate actual utilisation of the modelled communication infrastructure. Figure 2.1 illustrates how simulation and emulation allow speedy resolution if shortcomings are detected, and also aid in planning long-term future investment and its assessing impact on the current communication infrastructure.

2.3 Requirements, regulation and measuring capabilities

2.3.1 Measuring broadband data

In order to analyse the capabilities of any communication infrastructure one needs to obtain measurements from the actual installation. To do this we need data collected over certain period of time in order to have an overview how the communication infrastructure behaves. This will allow us to create models which can be used as references during simulations. Obtaining such measurements also has other benefits, as it provides the opportunity to bootstrap new communication infrastructures based on previously-collected network capabilities. Obtaining this kind of data is not always easy, as, for privacy and business reasons, network operators are not eager to share information about their system architecture. Furthermore, distributing sensors evenly to measure the performance and capabilities of a system requires careful planning, as they should not interfere with

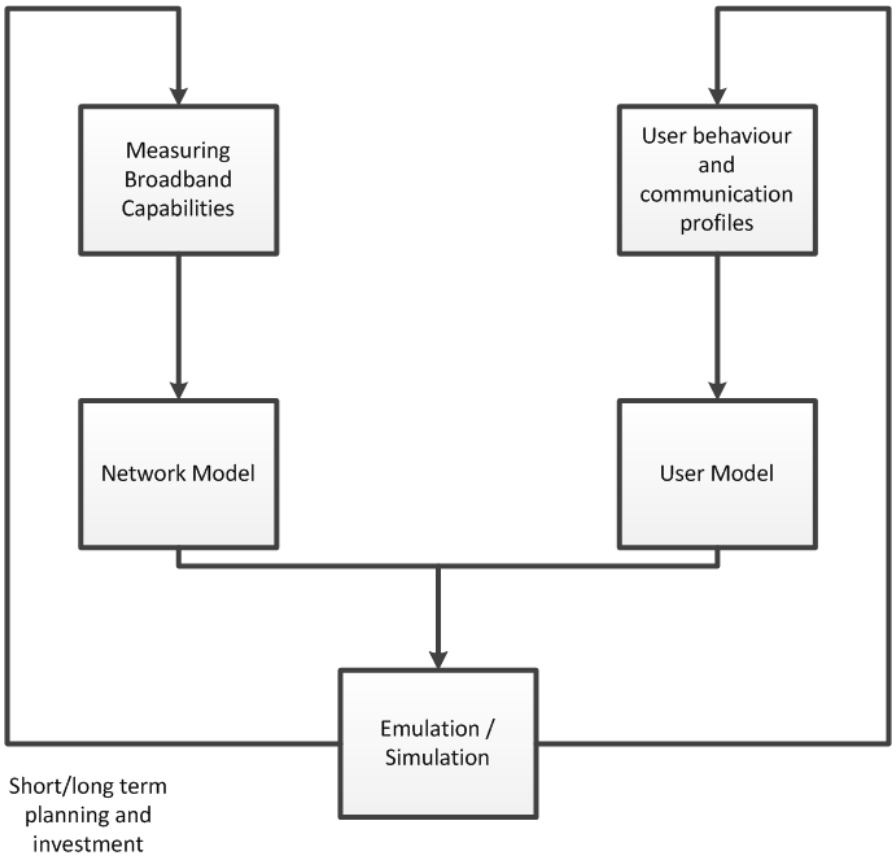
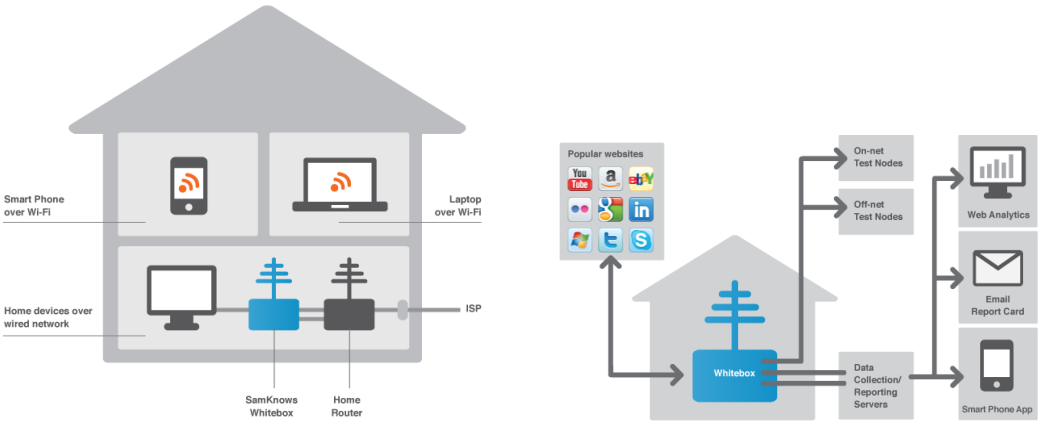


FIGURE 2.1: Vision

the daily traffic of the system, thereby biasing the measurements and inconveniently the operator. The US Federal Communication Commission (FCC) pioneered such efforts by measuring the residential broadband network through the Measuring Broadband America (MBA) programme which as initiated in 2010. The goal of the programme is to collect data with the support of national broadband service providers to measure the actual capabilities of their national cabled network infrastructure. The FCC publishes the raw anonymised data to encourage research. Since such data sets are not currently available in Luxembourg, our study uses the FCC MBA data set from 2012. In order to collect as much data as possible the FCC cooperated with SamKnows, an international statistics and analytics firm, to create a data set containing several different metrics from voluntary collaborators from among the residential population of the United States. Lately Samkowns also started offered a similar study outside of the United States. Volunteers were provides with off-the-shelf routers, so called Whiteboxes, with pre-installed firmware developed by SamKnows that handles the measurement and reporting of the data without interfering with volunteer's daily activities or requiring their intervention. The devices were deployed in each volunteer's own network, as illustrated in Figure 2.2[1]. The software-implemented measurement experiments were scheduled to run periodically.

[1]Figure source: https://www.samknows.com/broadband/how-it-works

The data was then collected, analysed and published as an annual report by the FCC. The experiments also allow any participant in the project to verify if the data measured by the experiments corresponds to the service level advertised by their Internet service provider. As private individuals are involved, the data needs to be anonymised to preserve privacy and secrecy of the data: all identifying information, e.g. name, street address, IP address, customer tier, is scrubbed from the original raw data before making it publicly available. More information on the MBA programme is available in the FCC MBA technical appendix[31].



(A) Deployment (B) Measure

FIGURE 2.2: Deployment of the SamKnows Whitebox to conduct the measurements

Several scheduled experiment were performed during the studies. Each experiment measured a different metric, with data being captured over a whole year, leading to quite huge data volumes. For 2012, the amount of raw data distributed over all the tables totals 291 GB. Conducting several different experiments provided a diversified view of the underlying network's capabilities. The tests included basic speed tests, measuring the upload and download speed of the connection, together with application testing features such as web browsing and VoIP capabilities. Figure 2.3 shows the different measurements taken during the experiments. Each table represents a different experiment. Each record in each table is identified by an opaque invariant unit identification number, *unit_id*, an anonymised identifier of the device of the volunteer participating in the study. The tables constituting the FCC data set are the following:

- *curr_webget* provides data on accessing the ten most popular websites in the US (e.g. Amazon, CNN, YouTube). The test measures the time to access the main page of each website, also capturing the total number of bytes for each website. This allows for an ongoing evaluation of the performance of the underlying network infrastructure, taking into account the benefits of Content Delivery Networks and other enhancing features.

- *curr_dns* also uses ten predefined well-known websites to measure the DNS resolution time for each of them. This test allows the ISP's DNS resolvers to be evaluated.

- *curr_netusage* captures the total amount of data received and transmitted from the user, as well as the number of bytes transmitted and received during an experiment.

- *curr_ping* contacts two different test nodes, one maintained by SamKnows outside any ISP network and one within an ISP's network, measuring the round trip time to each node using ICMP pings. This setup allows for comparison and evaluation of the ISP network infrastructure compared to other networks and provide a mean of validating the measured data.

- *curr_httpgetmt* and *curr_httppostmt* measure the actual speed of the connection. The *curr_httpgetmt* measures the retrieval time for a 1 GB binary file from a HTTP server. Once this file is retrieved it is immediately discarded on the client side. The test runs for a fixed duration and measures the average cumulative speed of the download channel. *curr_httppostmt* proceeds similarly but from the client side. It generates a binary file and sends it to a receiving server to measure the upload speed of the connection. The tests take into account single and multiple TCP connections and the start-up period of the TCP connection. This start-up period is recorded but is not reflected in the actual measurement of the speed.

- *curr_udplatency* and *curr_udpjitter* are used to measure the latency of the network infrastructure by sending UDP packets to a target node at regular interval. The packet is declared as lost if it has not been received back at the sender within two seconds. In this way , the test measures the average, minimum and maximum round trip times.

- *curr_videostream* simulates the use of a continuous stream of voice or video. The test assumes that users typically search for a stable connection which allows them to maintain a low or non-existent buffer underrun rate. The test captures the features of the buffer, i.e. buffer size, delay and the number of buffer underruns, as well as the connection itself, i.e. download throughput, jitter and latency.

For a more detailed view of each parameters captured by each table, please refer to Appendix B to see an exhaustive description of each captured feature. The analytics and results from these data sets are discussed in chapter 5

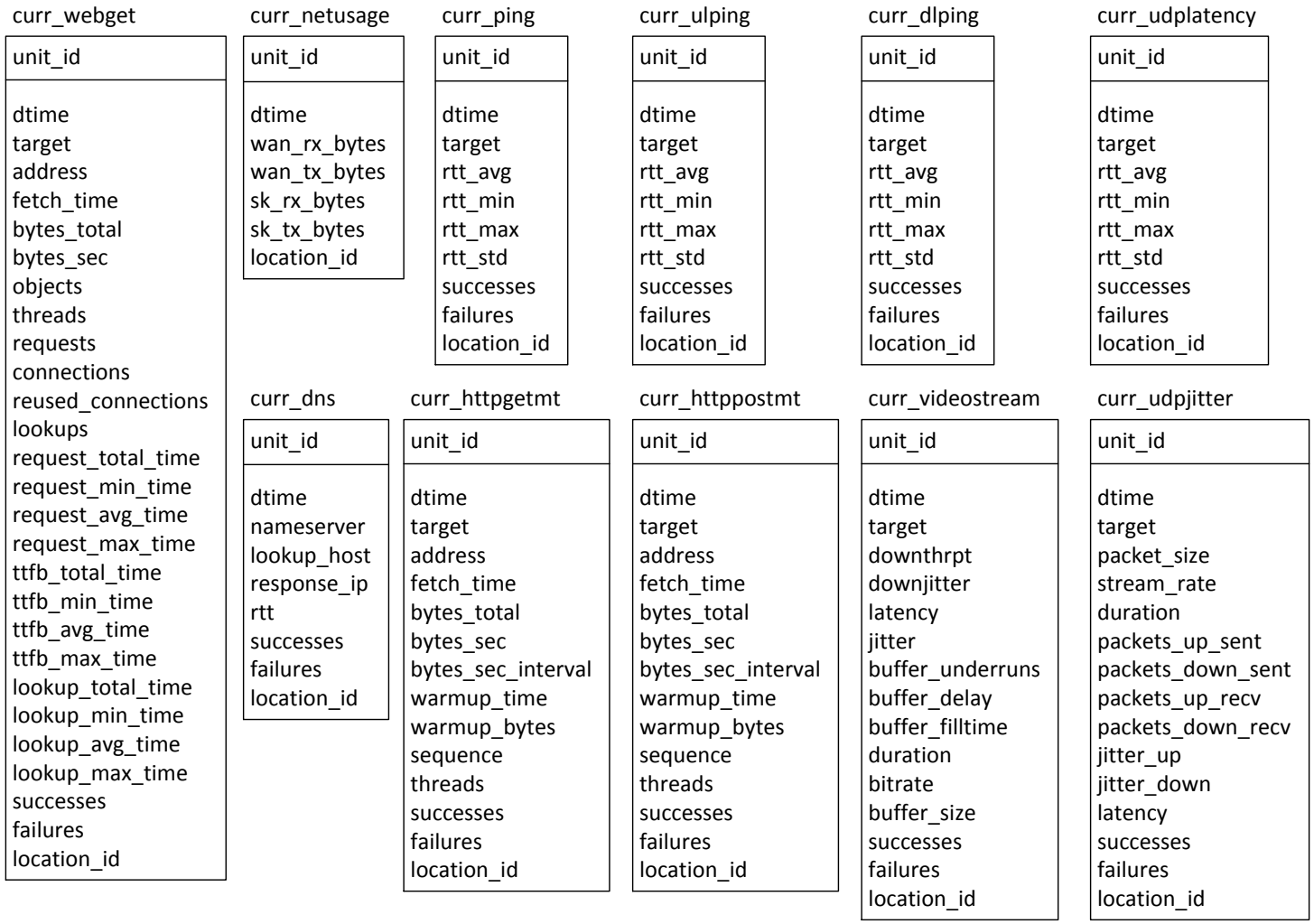


FIGURE 2.3: Datasets captured by the FCC

2.3.2 Modelling user mobility and voice communication

User profiles and behaviour are an integral part for our evaluation of the capability of a communication infrastructure. Detailed models of how the users use the system on a daily basis enables a precise and thorough analysis of the system. However, capturing behaviour and profiling users without their consent presents legal issues. Fortunately anonymised data sets are published by mobile operators to enable researchers to create models. However these models are always location-dependent and may not be transferable from one place to another. In our case we want to analyse the data for discovering user behaviours and creating user profiles during crises and quantifying how these differ from the normal cases. Such specific data is not easy to get, as crises are random, unpredictable events. Therefore the only solution is to capture communication traffic over an extended period of time and hope for a crisis to appear.

Like usage data above, using Call Data Records (CDRs) requires the permission of each operators and its customers, but not every operator wants to share information on its customers' behaviour. Privacy and anonymity of an operator's customers is a prime concern. As with the data set measuring broadband access, mobile phone operators are not eager to share information. Fortunately, we were granted access to an anonymised data set provided by Orange a subsidiary of France Telecom as part of the "Data for Development" D4D Challenge [32][2]. Orange offered access to an anonymised data set of captured mobile traffic over a period of 150 days between December 2011 and April 2012. The challenge was intended to improve everbody's quality of life by giving researchers the opportunity to analyse real-world data in order to create realistic models. The models can then be used to simulate better the effect of particular events on the communication infrastructure and the impact of user behaviour. Knowing how users react to certain incidents allows an early detection of events and reduces the time needed to resolve them. The D4D challenge also encourages the aggregation and use of other data sources in order enhance the data that it provides.

The capture period for the data was 150 days and contained CDRs for phone and text message exchanges between five million customers. CDRs are typically used by mobile phone operator to implement correct billing of services. In order to ensure consistency, only users currently subscribed to the mobile phone operator were considered and customers subscribing or cancelling their subscription were omitted. Also, only phone calls and short messages were captured. The customers were anonymised and the data was separated into four distinct data sets as shown in Figure 2.4. Each set had a specific purpose.

[2]D4D Challenge: http://www.d4d.orange.com/

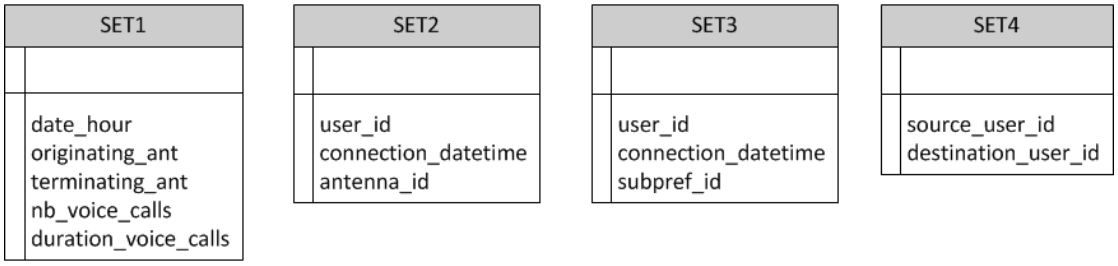


FIGURE 2.4: D4D data set

The first set, *SET1*, allowed us to analyse the base station-to-base station communication flows. Each record in this set comprises a time stamp during which the aggregated number of all ongoing calls and the overall duration of all the calls on an edge between a source and a destination base station is recorded. The set contains information on around 1200 individual nodes (base stations) measured on an hourly basis. Calls which lasted over an hour are contained in the record of the time slot in which they started and ignored in those following.

The second set, *SET2*, focuses on the individual trajectories of users and contains an anonymised user id as well as the base station to which the user is connected and a time stamp. 50,000 customers from the mobile operator where randomly selected to provide a consistent customer base. This set focuses on the mobility of end devices among base stations in order to determine the mobility patterns in the network itself.

The third set, *SET3*, is similar to the second but, instead of matching a given user to a base station it records the administrative sub-prefecture in which the user was at the given time. Here again 50,000 randomly-selected customers were used. This set enables hot-spot detection using real physical locations as it can be correlated to geographical location.

The final set, *SET4*, illustrates communication sub-graphs by capturing 5000 randomly-selected customers and keeping track of the communication among them throughout the whole period. This set focuses on social network analysis among the customer base.

Not mentioned in the description in the individual sets is a table containing the physical location of each base station in order to position them on a map.

This data set was used in our research, resulting in the scientific outcomes presented in chapter 4.

2.3.3 Regulation and standardisation

Our world is getting more and more digital every day. Governments follow this trend and are trying to shift all of the their administrative procedures to the digital domain. Citizens applying for documents, security checks at the airport, the opening of a bank account, all require information that is verified and authenticated by a public authority. This is usually done by issuing a specific document from the government containing a distinct marker that proves its authenticity. There needs to be agreement on how these marked documents should look and what information these documents contain in order to be compliant with other countries' procedures. Before the rise of the digital age, the authenticity of a single person was verifiable by examining his official document, i.e. personal identity card, driver's licence or passport. With the rise of the Internet and its popularity, the need for and electronic counterpart to such documents has become more and more pressing. Another important factor when considering digital identities is the reliability of the documents. There have been major efforts to provide such verifiable digital identity by governments. Among the pioneer and early adopters was the Austrian government with its regulation of digital identities in the Austrian eGovernment Act in 2004 [33]. While creating the necessary infrastructure in a single country seems to be feasible, there have been projects and research effort to create cross-border regulation and interoperability of the different digital identity schemes and their reliability.

The authors of [34, 35] leverage the use of electronic signatures in conjunction with governmental agencies. They describe a method that allows the coexistence of paper-based administrative procedures alongside electronically-enhanced procedures based on digital identities and signatures.They also developed a method that allows authentication of a printout that was generated using an electronic signature both for citizens and administrations. Their method involves printing all the relevant information for the validation of the document in a clean and concise manner.

In [36, 37] the authors explore the use of digital identities across the borders of administrative domains. For traditional face-to-face case, there are different means of identifying an individual, as mentioned above. However different administrative domains have different laws and regulations concerning identity. This is especially true when considering different countries. Countries implement regulations to their suit their own requirements and interoperability with other jurisdictions is not necessarily a prime concern. In [37] the authors describe a large scale pilot program driven by the European Union to promote and integrate the use of digital identities across borders. The STORK project (Secure idenTities acrOss boRders linKed) proposes a framework layered on top of nationally-implemented digital identification systems and provides and interoperability layer to allow cross-border validation and verification of digital identities.

Digital identities can be used during large-scale events such as catastrophes as a means of verifying the authenticity of the partners involved in communication flows. They can also be used by regulation policies in communication infrastructures of other governmental agencies to allow or restrict access tp particular items of information within these external networks. If we consider smaller countries with limited capabilities for handling large-scale catastrophes, then cross-border operation with the public safety agencies of neighbouring countries would benefit if such systems ans regulations already were in place.

Standardisation of is another important aspect. It simplifies interoperability and compatibility through predefined procedure and methods. The IETF (Internet Engineering Task Force)[3] has done and is doing much work concerning standardisation of the Internet. Promoting the use of open standards, the IETF is independent body, split into many working groups, each working on specific topics and standards. The ECRIT[4] working group deals with the resolution of emergency call placed over Internet technologies, and also explores the use of non-human initiated calls coming from sensors. Another example explored and used in this dissertation is the soon-to-be-standardized ALTO (Application-Layer Traffic Optimization) protocol[5] [38, 39]. This protocol focuses on an optimized pre-selection of peers in a P2P network by querying ALTO servers which provide additional information on the network topology. The protocol is based around two distinct maps, a network map and a cost map. The network map describes the underlying network architecture, while the cost map provides a metric for using the links described in the network map. This allows anyone (user, service provider or application) querying an ALTO server to get an estimation of th cost (bandwidth consumption, throughput, ...) of establishing a connection to a particular peer. The ALTO protocol is discussed further in Chapter 5

2.4 Simulation environments

Verifying setup and configuration on a large scale can be costly, and take some time. An alternative is to simulate a given setup with a network simulation tool, eliminating the need to acquire the specific hardware and invest time in building a test bed. One simply uses software to provide the specified functionality. There exists a vast number of different simulation tools from the very simple to the extremely complex. One of the most commonly used simulators in research is ns/2[40, 41]. Other well-known candidates are OPNET[42], Qualnet[43] from Scalable Networks and the newly released ns/3[44, 45],

[3] http://www.ietf.org/
[4] http://datatracker.ietf.org/wg/ecrit/
[5] http://datatracker.ietf.org/wg/alto/

the successor of ns/2. However one must always consider that simulation is simply an abstraction of the real world. Simulation of some problem that one may encounter in a real deployment might not be able to possible. There are already some issues concerning simulation tools, such as the quality of the random number generators used or the verification of certain implemented models. Simulation offers the possibility of repeatability, controllability of the environment and observability.

Performance comparison between the different simulation environment is a major criterion when selecting one simulator over another. Among determining criteria are CPU utilisation, memory consumption and scalability. The latter is a major factor in case of large-scale simulation involving thousands of network nodes. Selecting a network simulator always depends on the use-case to which it will be applied. In [46] the authors compare the performance of different network simulators by simulating an ad hoc on-demand distance vector (AODV) routing algorithm over varying network architectures and measuring the impact on memory consumption, CPU utilization and scalability. The authors determine that, according to their results, the recommended all-rounder is ns/3. The authors in [47] also compare different network simulators. As well as examining the computational performance the authors compare the overall simulation time needed to complete their defined scenario. The simulation time depends on several parameters, among them the number of nodes in the network. They determine that, depending on the underlying task, some simulators are a better choice than others, for example if the scalability of the network is a major factor they strongly recommend JiST, ns/3 and OMNeT++. They also conclude that ns/3 is a good all-round choice but at the time of their experiment, it had a limited library of simulation models.

Mobility pattern and models define how mobile nodes move inside the network during the simulation. This is especially important for mobile nodes such as notebooks and smartphones. Today being connected to a network has never been easier, as there is a multitude of different communication architectures at our disposal. All the simulators that we tested implement a certain number of mobility models, such as random walk or random waypoint model, although these are simple and generic as, considering the surface as a plane without any obstacles. There are however more complex mobility models, like the Manhattan grid that simulates the movement pattern occurring on a grid-based network similar to the road networks in cities across the United States. The mobility of nodes is also largely dependent on the mode of transportation. A pedestrian could use any kind of terrain, whereas, if the mobile node is mounted inside a car, mobility is limited to road network. Simple models offer some initial results when used in simulation, but often do not reflect reality. The more complex a mobility model is, the more accurate the results; however the computational complexity also increases. The use of an appropriate mobility model model has large impact on the simulation and the

results. The authors of [48] give an overview of existing mobility models and show the benefits and drawbacks of different models.

The network simulators presented in this section focus mostly on the correct simulation of packet transmission; the physical layer is heavily abstracted. Simulating a complete physical layer (PHY) can be complex as the PHY handles the way the bits are transformed into a signal form and back again, detected and corrected in case of bit errors. Most of these actions performed at the physical layer involvs mathematical operations on vectors and matrices. Depending on their size, these operations can become time- and resource-intensive. Implementing this in a packet simulator would quickly explode the computational complexity. Thus physical layers are often simplified to pure capture of throughput, error probability and delay, assuming that synchronisation is perfect between sender and receiver. This huge simplification has an important consequence for packet-domain models. Error in the physical frame are detected, but the simulators are not able to determine which bit is erroneous, even though this is relevant for upper-layer protocols. Depending on where the error occurs, the whole frame may need to be retransmitted. If however the errors are inside the payload, which could be an audio or video stream, then they could be neglected. These are important considerations that need to be taken into account when evaluating the results of a network simulator.

A different approach to assess communication in networks is the emulation of network components. In a network emulation, one or more physical end devices are attached to the network emulator and the end devices behave as they would when connected to a real network. This has the advantage of directly testing real devices and their actual behaviour. However, in terms of scalability, there are limitation due to the physical presence of the devices. There are efforts to create large-scale network emulator where it would be possible to emulate larger networks. A prominent and pioneer example is the Emulab[6] platform offered by the University of Utah. We focus however on the use of network simulation by presenting some of the available simulators.

2.4.1 Qualnet

Qualnet is the commercial version of the package previously known as GloMoSim[49, 50], a network simulator developed by Scalable Network Technologies. It is designed to simulate both wired and wireless network. A combination of both is also possible. Defining a scenario in Qualnet is relatively easy, but can also be time-consuming and complex. This seems contradictory but there is an simple explanation: while a scenario can easily be built through the rich GUI included in the software, the configuration of a

[6] https://www.emulab.net/

scenario can be quite time consuming as there are many properties that one can set on the various objects in a scenario, depending on the complexity one wishes to use during the simulation. As Qualnet is a commercial product, it comes with a wide varity of support and training offerings. However its major drawback, compared to the simulators presented below, is the high acquisition cost of licences.

2.4.2 Network Simulator 2, ns/2

ns/2 is open source software which is widely used in research and academia. Its implementation closely follows the structure of the OSI reference model. This means that each packet, during a simulation, traverses several layers namely: the network layer, link layer, MAC layer and the physical layer. In its standard distribution, ns/2 supports physical layers like those in LAN and WLAN. Other models of layers are available in the community. The same goes for routing algorithms. Due to the open source character of ns/2 anybody can encode models, but this leads to a major issue: one must carefully analyse any provided model in order to be sure that it behaves correctly. The same applies if one want to implement one's own model.Initially ns/2 could simulate only either wired or wireless networks; only after the introduction of a new component it was possible to perform hybrid simulations.

The development of personal scenarios is done through TCL script files which describe what the simulator does at which point. The whole design process is text-based: there is no GUI which one can use the describe the simulation and its components. The only visual representation that is available is the output, which can be visualize with a built-in tool, nam. This tool will graphically represent the run and results of the simulation as an animation.

2.4.3 Network Simulator 3, ns/3

ns/3 is the successor to the currently-available ns/2 and was developed as its replacement. Just like to ns/2, ns/3 is open source and has a community of researchers and developers extending its features. ns/3 was completely redesigned from scratch and therefore there is no backward compatibility with ns/2. ns/3 is also a text-based simulation tool and lacks a GUI for scenario description. However, while in ns/2, the TCL scripting language was used to describe scenarios, ns/3 chooses to follow another path and gives its users the possibility to either write the simulation scenario in C++ code or by using the Perl scripting language. ns/3 is designed such that both offer a similar syntax with minor differences and therefore no effort is require to learn two different languages as with its predecessor.

As ns/3 was a complete redesign, its architecture was revised and brought even closer to physical end devices[51, 52]. Efficient memory management and state-of-the-art programming paradigms were used to improve it relative to its predecessor. A further enhancement allows its simulation outputs to be analysed with popular tools like *tcpdump*[7] and *Wireshark*[8]. As mentioned above ns/3 is not compatible with ns/2, meaning that no community model, routing model or application simulators developed for ns/2 can be used. This a major issue, as ns/3 has not yet reached the level of maturity ot its predecessor and support is not as widespread as forns/2.

2.5 Real-world trials

Several real-world trials of systems for interoperability have taken place. In this subsection, we summarise two of them.

U2010 [53], Ubiquitous IP centric Government & Enterprise Next Generation Networks Vision 2010, was a European research project instigated by the University of Luxembourg, which focused on the interoperability of public safety communication networks. The core ideas behind U2010 were to offer the most suited means of communication available and to provide the most effective way to access incident-relevant information for anybody involved in an emergency situation be it of major or minor scale. U2010 used existing and new technologies, such as IPv6 to reach its goals. The project successfully demonstrated the seamless use of multiple communication architectures and the automatic redirection of traffic to other communication channels. Several other projects, such as SECRICOM[9] and FreeSIC [10] are working on following up on the experience and results gained from U2010.

The North Atlantic Treaty Organization has developed a framework[54] for the successful collaboration and interoperability between the communication and information systems of member countries and the NATO infrastructure. Mainly focused on military interoperability, it can also be adapted to public safety communication. Several NATO studies [55, 56, 57] show the importance of an interoperable communication and information infrastructure and how this can improve overall situational awareness and communication flows compared to the current situation.

[7] http://www.tcpdump.org/
[8] http://www.wireshark.org/
[9] http://www.secricom.eu/
[10] http://www.freesic.eu/

Chapter 3

Security Monitoring and Control for Content-Centric Networking

Content-Centric Networking (CCN) is one of the most promising research area for a future Internet and has therefore also potential to be used in Emergency networks. The goal of CCN is to obtain a more scalable, secure, collaborative Internet supporting context-aware services. However, as a new overlay infrastructure, CCN raises the need of a new monitoring architecture to assess security of CCN devices. In particular, the stateful nature of CCN routers introduces new attack threats that need to be addressed. This chapter proposes a monitoring approach for the instrumentation of CCN enabled network nodes. The rationale of this monitoring approach is demonstrated through real experimentations to detect and mitigate network level attacks against CCN. However, several key components that secures the current Internet are still missing in CCN, in particular a firewall able to enforce security policies. Therefore this chapter also introduces a comprehensive study of CCN security requirements from which a first CCN compliant firewall was designed, including the syntax and definition of rules. In particular, based on CCN features, the firewall can filter packets regarding both their authentication and the semantic of the content name. Finally a performance evaluation of the prototype firewall is also provided.

3.1 Introduction

Content-Centric Networking (CCN) is a new routing paradigm developed at PARC by Van Jacobson et al [58] but also known as Named Data Networking at a larger scale [59]. Based on the observation that today's communications are more oriented toward content retrieval (Web, P2P, etc.) than point-to-point communications (VoIP, IM, etc.)[60], CCN proposes to deeply revise the Internet architecture to best match its current usage. In a nutshell, contents are addressable, routable and authenticated, while their locations do not matter anymore. They can be replicated and stored (especially popular contents) on any CCN node. People looking for a content can securely retrieve it from the best locations available.

On one side, the client-server architecture needs more and more investments in expensive content delivery networks and server farms to be scalable. Agreements between ISPs (Internet Service Providers) and content providers tend to benefit to big web-actors that centralize user-generated contents. On the other side, the P2P paradigm makes an inefficient use of resources being mostly unaware of the physical location of the peers. In this context, we think that CCN could be an answer to the challenges that Internet will face in the near future and deserves research efforts from the community to properly investigate the applicability of this paradigm.

From a management point of view, Content-Centric Networking introduces new challenges. Firstly, it is hard for a content provider to monitor and control the diffusion of its content over the network after the initial release which leads to accountability issues because content can be distributed from any CCN node without requesting the original provider. While it is an important quest from a management point of view, we focus on the security aspects which present a more critical drawback of early deployment of CCN. Secondly, CCN routers are stateful as the route between a content and the requester has to be memorized. This stateful nature can lead to new possible DoS (Denial of Service) attacks which exploit the CCN routers limited memory size and these kind of attacks must be detected and mitigated. In this paper, we address this second issue because we think that security issues could highly decrease the appeal of the technology and reduce further research efforts. To that end, we:

1. model a DoS attacks on a CCN architecture
2. propose a monitoring Architecture for Content-Centric Networking which is able to detect them.

This new paradigm is promising and has been built from the experience gained from the current Internet. In particular, security aspects have been considered in the design which

includes, for example, the authentication of contents. However, even if some building blocks exist to secure CCN, it is impossible to guarantee a 100% security even for this new paradigm. For example, nothing protects against a flooding attack as we highlight in this chapter or nothing prevents a user to download malicious or forbidden contents. In IP networks, firewalls are usually the first level of protection by enforcing the security policy defined by the administrator within the company's network. This paper leverage the usage of firewalls for the sake of content centric networking security and presents the following contributions:

1. identification of the security needs for a functional CCN architecture
2. design of a firewall dedicated to CNN, with semantic features
3. performance evaluation of the firewall

There are several architecture, like CCN, which aim to shift away of today Internet point-to-point primitives, move to a more data-oriented and content-centric paradigm, replace the end-to-end communication network model by publish/subscribe model of a distribution network and to used cached copy of content for faster retrieval. Main differences between the different research approaches are the content naming scheme and how inter-domain routing is handled.

TRIAD [61] was the first to propose such an architecture. Names in TRIAD are based on URLs and use DNS for their resolution. Furthermore directories are used to map content to a replica server close-by. Shortly afterwards Brent Baccala in [62] expressed a similar idea of moving a more content-centric approach.

In 2007, *Koponen et al.* renewed the idea of a content-oriented network at Berkeley. DONA[63] was the name of this project. They followed another idea which consists into replacing DNS with flat and self-certifying names avoiding PKI for key verification.

The PSIRP[64] project introduces an architecture based on rendezvous points. Content is publish at the source. Each pieces has two labels, a public label used for the subscription to the content a private label used to verify the publisher.

Another research project focusing on content-centric networking is the 4WARD NetInf[65] project. Content is published using information units called InformationObject (IO). As in general every IO needs a unique identifier by which it can be referenced, a multi-level DHT (Distributed Hash Table) handles the name resolution and location lookup for a given IO.

3.2 Content-Centric Networking background

3.2.1 Paradigm

The main idea behind CCN (also called Named Data Networking[66]) is a paradigm shift towards content oriented networking and routing. Today's Internet relies on the well established communication paradigm in which two end-points communicate over a network. However, regarding the behavior and habits of today's users, there is a strong shift towards content and not the location where this content is stored. Today, Internet is becoming more and more a content distribution network. Therefore some claim[67] that the best approach is to start from scratch building a new Internet architecture. However this involves long term investments for ISPs and CCN can thus be used over the already established architecture (for example CCN over IPv6).

The most popular architecture for research purposes is the Content Centric Networking proposed by Van Jacobson et al [68] from early 2007 and later introduced to the research community [58] in 2009. CCN current development is quite advanced thanks to the CCNx open source framework [69]. PARC pursues research efforts of their architecture, describing and implementing advanced features and functionalities as the capacity of CCN to transport voice [70] with the adapted architecture. Many issues are described in [59] and still need to be addressed to make CCN (or Named Data Networking) a viable solution, for example: the scalability of routing on names, the efficiency of key management, the management of contents or the security of CCN nodes are critical questions deserving research efforts. Also, the design of a complete model to better understand the working and the benefit of a CCN architecture according to the network configuration, as proposed by Carofiglio et al [71], is an important step forward. Privacy on the Internet is more than ever a critical topic. DiBenedetto et al proposed in [72] a application over CCN that enables privacy preserving communications while introducing less relative overhead than TOR running over IP.

3.2.2 Node model

CCN has two main types of packets, *Interest* and *Data* as seen in Figure 3.1. A user who wants to access a certain content sends out an *Interest* packet, specifying the name of the content (as defined by CCN nomenclature ContentName) to all its available faces. A Face can be anything which can serve as medium for transmitting and receiving data. A node which receives this packet and that can 'satisfy' the *Interest* sends out the corresponding *Data* packet onto the face from which it received the *Interest*. By definition, CCN nodes are stateful and only send *Data* if there was an *Interest* beforehand.

(A) Interest Packet (B) Data Packet

FIGURE 3.1: CCN packet structures

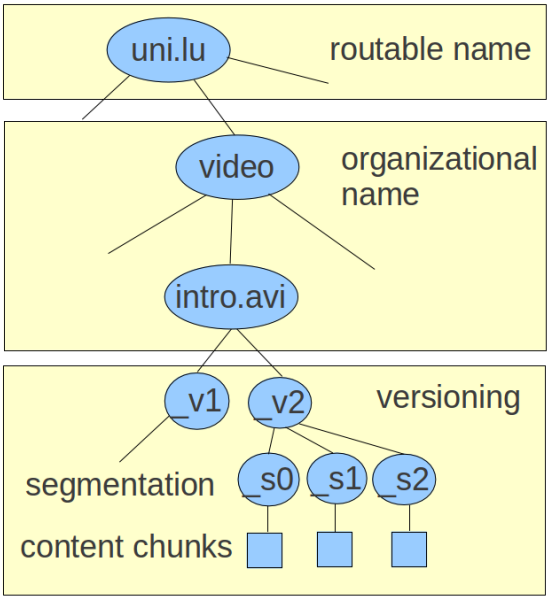


FIGURE 3.2: Hierarchical naming of a CCN content item

Data can only 'satisfy' a specific *Interest* if the ContentName of the *Interest* packet is a prefix of the *Data* packet. CCN names are defined in [58] as "opaque, binary objects composed of an (explicitly specified) number of components". This structure allows a fast and efficient prefix-based lookup similar to the IP lookup currently used. This new routing paradigm is based on a plain-text hierarchical naming instead of regular hosts' IP addresses so that names are directly intuitive and do not need an indirection mechanism between names and contents like DNS.An example of this hierarchical naming structure is presented in figure 3.2 for a content named "ccnx:/uni.lu/videos/intro.avi". It also allows names to be context dependent *i.e. /ThisRoom/Printer* references a printer in the current room. This context-naming could make possible efficient context-aware service discovery in the future Internet of Things.

CCN nodes are composed of three main table structures which handle the forwarding of packets, presented in figure 3.3. At the arrival of an *Interest* packet on any given

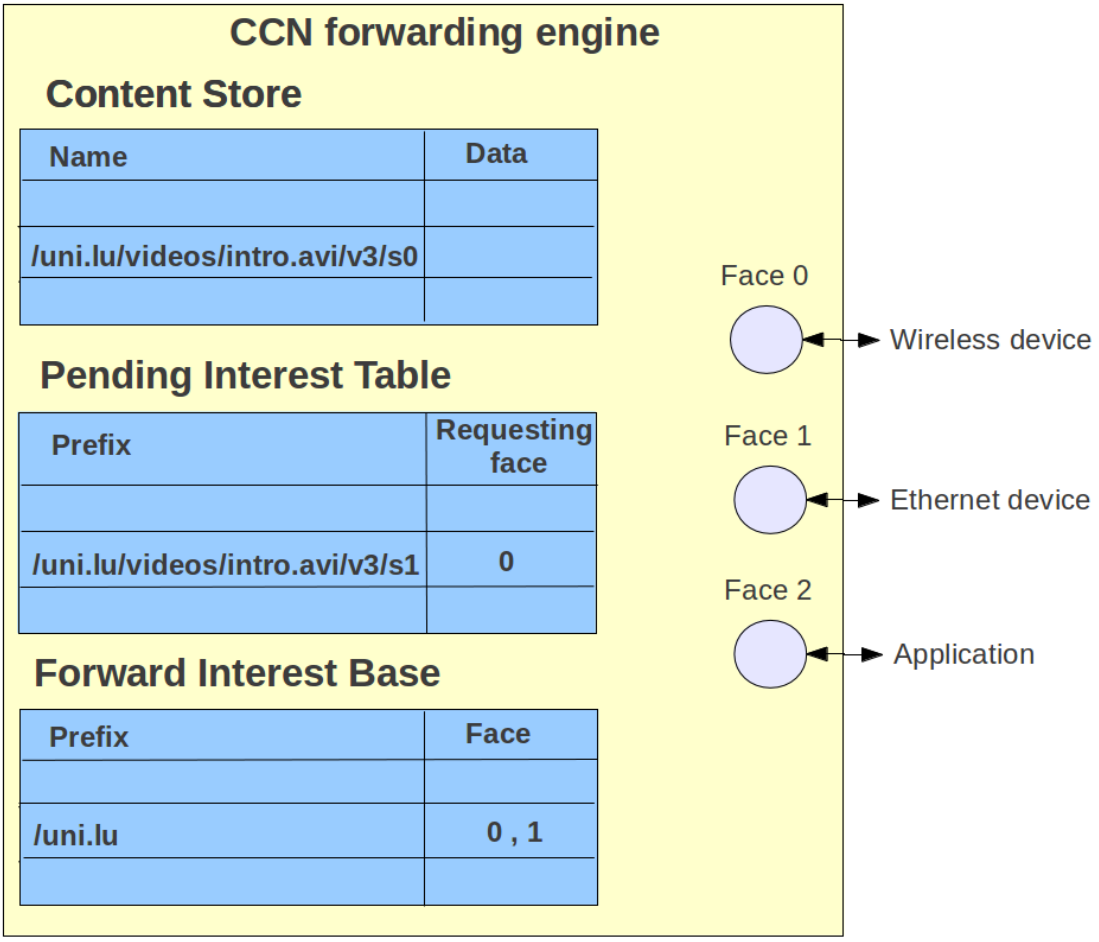


FIGURE 3.3: CCN forwarding engine

face, the engine performs a longest-match lookup on its structures and action is taken depending on the lookup result. The first structure to be searched is the Content Store. It can be seen as a buffer memory of past *Data* packets on the current router. IP routers also have such a buffer but it is purged once the packet is forwarded. The Content Store however preserves the *Data* packet based on LRU (Least-Recently-Used) scheme and enables therefore a fast retrieval of currently popular demands. If there is a match, the router forwards its local copy of the content to the face on which it received the *Interest* and updates its Content Store accordingly.

If there is no match in the Content Store, the lookup is launched on the next structure which is the PIT. The PIT stands for Pending Interest Table and keeps record of *Interests* waiting to be resolved upstream by other content source(s). If a received *Interest* matches an entry in the PIT, the engine compares the faces recorded for that entry. If there is already one existing, no update is made. Otherwise, the face from which the *Interest* was emitted is simply added to the list of already waiting faces.

If no match-up is found in the PIT then the engine searches in its last structure: the FIB. The Forward Information Base keeps record of potential content source(s) and works similarly to its IP counterpart except that it stores a list of possible providers for a given name rather than a single one only. If a match is found, the engine then creates a PIT entry for the given *Interest* and it is forwarded to all faces specified in the FIB entry. If no match could be made, it means that the current router has no information on the demanded content and discards the *Interest*.

CCN has also built-in strategy and security layers. The strategy layer is used to define policies to select which face is the best for given contents. In fact, due to its design, FIB entries contain multiple faces. CCN can send periodically *Interests* to all outgoing faces without fearing of loops and thereby testing which of the faces responds the fastest. This one will be used as preferred until another round of this experiment yields to a different result. Criteria for experimentation interval can be a threshold of packets sent, a time out, change of the SSID(Service Set Identifier), etc.

The security layer ensures that the content received by a previously announced *Interest* is authentic. As in CCN only the content matters but not the route it takes, the only thing which needs to be checked for authenticity, consistency and integrity is the content itself which reversely means that end-to-end encryption is not needed any more. Key management is another issue often discussed. In [58, 70], several solutions are discussed which range from a PKI to PGP like web-of-trust.

3.2.3 Security scheme

The key-stone of CCN security is the trust in the publisher. In fact, the paradigm shift of CCN makes every node capable to answer a Data request. To ensure the security of communications despite untrustworthy nodes, data is authenticated by its provider, not the connections it traverses (no end-to-end encryption). This security scheme should provide better results. For example, a secure connection to a mail server does not avoid SPAM mails to be received while with CCN, SPAM mails will fail the authentication because of their malicious source and should be discarded. So, CCN strongly relies on cryptography to authenticate the contents so that users can clearly know who emitted the content and can discard those from untrustworthy sources to avoid malware. Also, encryption is used to ensure privacy. CCN provides native security and privacy by encryption with lower overhead than current protocols [58]

To securely authenticate content, CCN has to bind the content name, the content itself and the content provider. To do so, the following information is embedded in each CCN data packet: $Signature(Name, Content, SignInfo)$. SignInfo includes: cryptographic

digest or fingerprint of publisher's key, key or key location. Key management is another issue often discussed. In 3.3, several solutions are discussed which range from a PKI to PGP like web-of-trust. To ease key management in CCN, Jacobson et al. propose to see an organization as a content name and public key as a Data. They propose to use the SDSI/SPKI where keys are mapped to identities via namespaces (CCN names) so that there is no single source of trust like the current certification authorities. This scheme opens evidence based security where data provenance is well known (traceability).

So, the CCN architecture, and particularly the security layer, provide means to secure a network. Based on IP use cases, we next define the needed features that a firewall should have to enforce an efficient security policy for CCN.

So far, there has been little investigation on the security of CCNs. We currently lack of knowledge of possible security issues related to CCN and particularly: 1) how could attackers benefit from CCN architecture to diffuse malware; and 2) how could CCN be attacked in order to intercept or disrupt the communications carried?

In particular, due to stateful routers (as the route between a content and the requester has to be memorized) network devices are exposed to new threats that can be exploited for attacks. Attackers may disturb the management of CCN nodes' tables (PIT, FIB, CS) from which result their performance and quality of service. This can lead to new possible DoS (Denial of Service) attacks or to malicious monitoring, that must be detected and mitigated. In fact, Tobias Lauinger [73] identifies several attacks related to caches, in particular denial-of-service attacks against CCN routers, but he only investigates one of these, "cache snooping" that enables attackers to efficiently monitor the content retrieved by their direct neighbours. Smetters et al. [74] of PARC propose authenticating the links between names and content, in addition to names and content themselves, in order to identify trustworthy content and avoid malware diffusion. However, the necessary PKI is hard to set up and if a provider becomes malicious, revocation remains a major problem that must be addressed in future work.

It seems that security by design may not be sufficient without proper tools that will enforce the security policies applied in a CCN network. Among others, CCN currently lack of: 1) a monitoring architecture that can detect suspicious traffic by analysing the activity of CCN nodes tables, 2) a firewall that can apply the security policy of a network (discard untrustworthy content and provider) based on name prefix or signature as proposed in this paper, and 3) a distributed key management system to enforce an efficient and safe authentication of CCN content and content providers as well as the revocation of malicious ones.

3.2.4 Alternative content oriented approaches

There are several architecture, like CCN, which aim to shift away of today Internet point-to-point primitives, move to a more data-oriented and content-centric paradigm, replace the end-to-end communication network model by publish/subscribe model of a distribution network and to used cached copy of content for faster retrieval. Main differences between the different research approaches are the content naming scheme and how inter-domain routing is handled.

TRIAD [61] was the first to propose such an architecture. Names in TRIAD are based on URLs and use DNS for their resolution. Furthermore directories are used to map content to a replica server close-by. Shortly afterwards Brent Baccala in [62] expressed a similar idea of moving a more content-centric approach.

In 2007, *Koponen et al.* renewed the idea of a content-oriented network at Berkeley. DONA[63] was the name of this project. They followed another idea which consists into replacing DNS with flat and self-certifying names avoiding PKI for key verification.

The PSIRP[64] project introduces an architecture based on rendezvous points. Content is publish at the source. Each pieces has two labels, a public label used for the subscription to the content a private label used to verify the publisher.

Another research project focusing on content-centric networking is the 4WARD NetInf[65] project. Content is published using information units called InformationObject (IO). As in general every IO needs a unique identifier by which it can be referenced, a multi-level DHT (Distributed Hash Table) handles the name resolution and location lookup for a given IO.

3.3 Threat descriptions

Content Centric Networking improves the security of Internet communications in many ways. First of all, CCN messages can not be sent toward a node without any prior *Interest* request from that node which makes the classical denial of service scheme inefficient as the attacker would need his target to generate a lot of *Interests* to enable the DoS attack. Also, CCN strongly relies on cryptography to authenticate the contents so that users can clearly know who emitted the content and can discard those from untrustworthy sources to avoid malware. If CCN improves security in some points, it also raises the possibility of new kind of attack. Unlike a terminal host which is less exposed to attacks, CCN routers are more vulnerable than IP routers because of their stateful nature and the

management of their inner tables from which result their performance and quality of service.

By focusing on CCN routers, new kinds of Denial of Service can be performed. We categorize these attacks regarding the tables they target.

3.3.1 Pending Interest Table attack

A first attack can be focused on the Pending Interest Table. This table is critical because of the stateful routing of CCN network. If an attacker can manage to fill the PIT with a lot of forged *Interests*, legitimate *Interests* might be dropped, resulting in the denial of service of the pending communications. The attack is easy to achieve from the technical aspects. The attacker only needs to generate a lot of *Interests* whatever is the requested content in order to create entries in its PIT. Such an attack would benefit from distributed attack sources that would make it more difficult to detect. Therefore, the monitoring of the PIT to avoid flooding is a critical point for a safe and efficient CCN infrastructure.

3.3.2 Forward Interest Base attack

Unlike the IP address space, the CCN address space is not clearly bounded as domains are defined through strings rather than a small IP prefix. Therefore, a possible attack consists in generating and advertising a lot of contents belonging to different domains in order to fill the FIB on a face of the router. In that way, new legitimate domains can not be routed through the CCN device which interface is full. This DoS is critical because one of the major interest of CCN is to allow end-users to directly diffuse their content in a peer-to-peer way instead of relying on big Internet content providers. This attack could reduce the diversity of routable domains and consequently the interest of CCN.

3.3.3 Content Store attack

DoS can also be launched against the Content Store in order to decrease the efficiency of the caching mechanism which is one of the main components that provides the incentive to deploy CCN infrastructures. According to the caching policy, an attacker could generate a lot of download requests for unpopular contents which would modify the distribution of the downloaded contents and update the cache in a inefficient way. From a technical point of view, this attack is hard to achieve for a single attacker as it would need a lot of bandwidth to have a significant impact on the distribution of contents passing through the router.

3.4 Monitoring architecture

3.4.1 Requirements

The monitoring task consists of collecting information about the functioning and the current status of the CCN nodes. The objective is to correctly select and process the necessary information to highlight important facts. In this paper, we focus on the detection of anomalies resulting from attacks. As CCN works in a distributed manner with independent nodes, monitoring the network from a global perspective is thus hard. A solution could counter this problem by leveraging a central service interacting with a large set of CCN devices to collect and analyse information. However, this raises serious issues about reliability and scalability and does not fit to the CCN paradigm where each node has to be involved in the functioning of the network including the security related aspects.

Therefore, to stick to the CCN paradigm, our architecture is implemented at each CCN device which has to monitor itself for detecting anomalies. As presented in the previous section, a device has three main components: the FIB, the PIT and the Content Store. Each of them plays a crucial role in the well operation of CCN. For example, an abnormal Content Store can provide faulty contents or make the caching inefficient; a badly populated FIB may entail erroneous forwarding and so, some content may not be accessible any more similar to a bogus PIT which leads to disrupt the data content transmission over the back path of a request. Therefore, all of these three tables have to be monitored.

The objective of our monitoring architecture is to detect attack patterns by monitoring the recent past activity over the three tables. Since they may contain many information which are related to many actions (lookup, updates, etc.), monitoring and keeping track of all individual entry or action requires many resources that may delay the process or even affect the entire functioning of a node up to a denial of service in the worst case. To guarantee the scalability and the timeliness, our architecture is designed to represent the three monitored tables by condensed metrics which values can be easily tracked along time.

Finally, devices can also share knowledge for detecting the attacks, in particular for highly distributed ones like DDoS, as well as for preventing future ones or recovering efficiently from anomalies. This may be provided as a content where devices can express their interest through the underlying CCN itself. However, as the individual monitoring is already required prior, our paper focuses on it.

3.4.2 Instrumentation

As explained in the previous section, there are different components of a CCN node that can be monitored for detecting malicious activities. All of them are impacted and/or impacts the network activity. For example, a node may receive an *Interest* (ingoing network activity) which has to be forwarded (outgoing network activity), this will update its PIT. Therefore, monitoring the network activity will track the global functioning of a node and its internal components without having a particular monitoring function for each of them.

For detecting attacks, network statistics are retrieved periodically from the CCNx implementation [69], every τ seconds. So, for each time window t, the following metrics are considered for all active faces of the CCN devices:

- $recv\_byt_t$: number of received bytes per second
- $sent\_byt_t$: number of sent bytes per second
- $recv\_data_t$: number of received *Data* packets per second
- $sent\_data_t$: number of sent *Data* packets per second
- $recv\_intr_t$: number of received *Interests* per second
- $sent\_intr_t$: number of sent *Interests* per second

We also consider more synthetic values on the router status that are also provided by the CCNx implementation:

- on Content statistics: number of $accessioned_t$, $stored_t$, $staled_t$, $sparse_t$, $duplicated_t$ and $sent_t$ contents
- on recent *Interest* statistics: number of $named_t$, $pending_t$, $propagating_t$ and $noted_t$ *Interests*
- on total *Interest* statistics: number of $accepted_t$, $dropped_t$, $sent_t$ and $stuffed_t$ *Interests*

3.4.3 Classification algorithm

The objective of the classification algorithm is to label each time window as anomalous or benign. A time window t_i is a tuple defined as:

$$< recv\_byt_{t_i}, sent\_byt_{t_i}, recv\_data_{t_i}, sent\_data_{t_i},$$
$$recv\_intr_{t_i}, sent\_intr_{t_i}, accessioned_{t_i}, stored_{t_i},$$
$$staled_{t_i}, sparse_{t_i}, duplicated_{t_i}, sent_{t_i}, named_{t_i},$$
$$pending_{t_i}, propagating_{t_i}, noted_{t_i}, accepted_{t_i},$$
$$dropped_{t_i}, sent_{t_i}, stuffed_{t_i} >$$

This chapter leverages Support Vector Machines (SVM) [75] which are able to efficiently classify data, even if the data points are not separable linearly, while the complexity remains low [76] allowing our solution to detect in real time attacks affecting the monitored CCN nodes. For sake of clarity, we have considered a single attack at a certain time which is handled by 2-class SVM. However, multiple attacks could be detected by building a unique multi-class classifier [77].

2-class SVM is a supervised method and so requires M training samples: $Train = \{(t_1, l_1), \ldots, (t_M, l_M)\}$ with $l_i = 1$ if the time window t_i contains an attack, else -1. For enhancing the data separability, these samples are mapped into a higher dimensional space. Defining an efficient mapping function, ϕ, is a difficult task because it corresponds to add an additional dimensional space over data not given by any features. This however may be avoided by using the kernel function defined as:

$$K(t_i, t_j) = \langle\ \phi(t_i).\phi(t_j)\ \rangle \tag{3.1}$$

Because our data points are vectors representing the different metrics of a time window, the Radial Basis Function (RBF) is adapted:

$$K(t_i, t_j) = e^{-q||t_i - t_j||} \tag{3.2}$$

where q is tunable. Different values for q have been tested to choose an optimal which provides the best result.

Once in the space, the points may be linearly separable by a hyperplane which divides the samples of two classes with the maximum margins regarding the hyperplane. This leads to the following optimisation problem.

$$max \sum_{t_i \in Train} \alpha_{t_i} - \frac{1}{2} \sum_{\substack{t_i \in Train \\ t_j \in train}} \alpha_i \alpha_j l_i l_j K(t_i, t_j) \tag{3.3}$$

subject to where $C = 1.0$ determined through initial experiments:

$$\begin{gathered} \sum_{t_i \in Train} \alpha_i l_i = 0 \\ \forall\ t_i \in Train, 0 \leq \alpha_i \leq C \end{gathered} \tag{3.4}$$

As highlighted by these equations, the problem solving leads also to determine α_i which is used afterwards for classifying a new time window. A major advantage of SVM is that it relies on a subset of initial samples for the decision function, *i.e.* the support vectors which represent the training points such that $\alpha_i \neq 0$. Assuming, t_x, a time window which requires a prediction about the attack, it will be labelled by the following function:

$$f(t_x) = sgn(\sum_{(t_i, l_i) \in Train, \alpha_i \neq 0} \alpha_{t_i} l_i K(t_i, t_x) + b) \tag{3.5}$$

3.5 Experiments

3.5.1 Attack description and test environment

The most critical threat among the three described in Section 3.3 is clearly the attack on the PIT table. Attacks on Content Store can just reduce the efficiency of the cache and therefore do not present critical security issues. Also, attacks on the FIB will become critical when large deployments of CCN will occur, with many providers announcing contents, while PIT attacks is already a threat for local deployments involving few CCN nodes. Therefore the main threat we want to address when monitoring CCN nodes is the Pending Interest Table DoS attack as described in section 3.3.1. To realize the detection, we first implemented different attack strategies against the PIT in a single attack tool based on the source code of the *ccndsmoketest*[78] program provided in the CCNx implementation. The PIT stores *Interests* according to the faces they belong. To fill the PIT table we have to consider two dimensions, the number of faces created and the number of *Interests* requested.

- **Burst attack**: sending multiple *Interests* to multiple faces. Our first strategy sends a given number of *Interests* on a given number of faces. In extreme scenarios, we can send a lot of *Interests* to a single face or send a single *Interest* to several faces. Both dimensions can be combined leading to the following

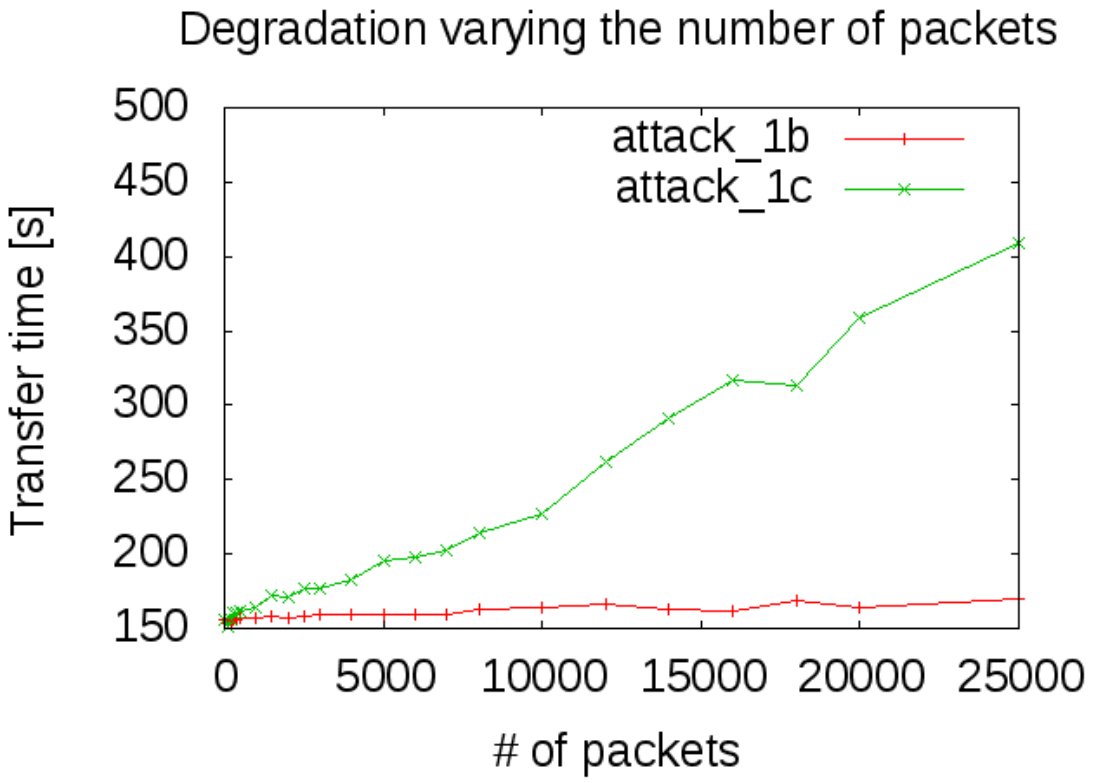


FIGURE 3.4: Impact of varying number of packets (attack 1b, 1c)

definition $Attack1(\#packets, \#faces)$ with the aforementioned remarkable values: $Attack1a(1, n)$, $Attack1b(n, 1)$, $Attack1c(n, 100)$, $Attack1d(100, n)$ where n defines the attack aggressiveness and may be tuned to study the impact of the attack.

- **Long duration attack**: keeping alive multiples faces with periodic *Interests*. Our second strategy consists into making the DoS more efficient by keeping alive a lot of faces with a small number of *Interests* that we send periodically. In this case, the attack aggressiveness, n, is the number of targeted faces. In this paper, keep alive *Interests* are sent every $t = 4$ seconds.

Our test-bed is composed of this attack tool and of two CCN devices running the routing daemon *ccnd* provided by CCNx. Both devices are on an restricted network used for this purpose. For each step in the experiment we transfer a 366MB video file from one device to the other.

We defined the impact factor of our attack as the time overhead introduced when transferring a content between our two CCN devices. Figure 3.4 shows the impact of the number of *Interest* packets we inject while targeting a constant number of faces. Firstly, we vary the *Interest* packet generation to inject them over one face (Attack 1b). The later we use the same principle but we inject them on 100 faces (Attack 1c). Figure

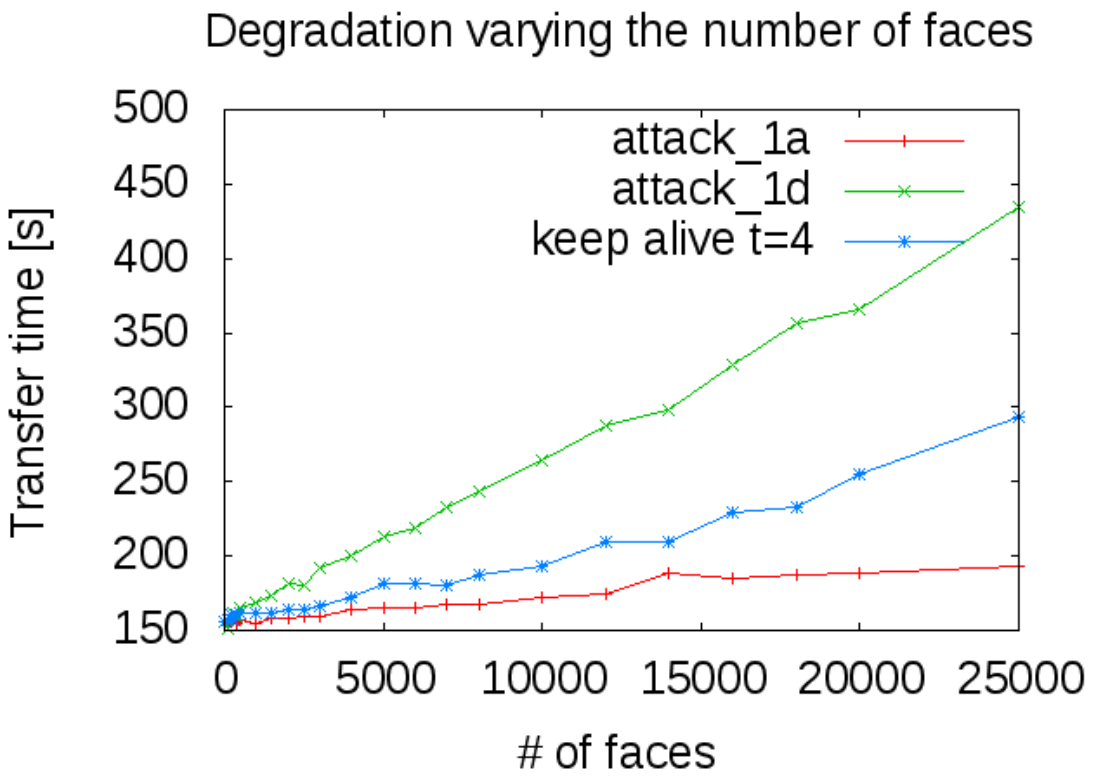


FIGURE 3.5: Impact of varying number of faces (attack 1a, 1d, long duration)

3.5 is similar but we vary the number of faces while maintaining a constant number of injected *Interest* packets (Attack 1a and 1d). Logically, performances are more degraded when the number of faces or *Interest* packets increase in particular if both are combined (attack 1c and 1d). Moreover, multiplying the number of faces used has a similar effect than sending multiple *Interest* packets. The second attack strategy, keeping alive many interfaces, can also significantly degrade the performance.

3.5.2 Attack detection

As previously described attack detection is based on SVM analysing metrics over time windows. In our evaluation, the size of a window is set to one second. To assess the detection, the following metrics are used:

- the True Positive Rate (TPR): proportion of correctly identified windows presenting an attack,
- the False Positive Rate (FPR): proportion of windows without attack classified as attacks.

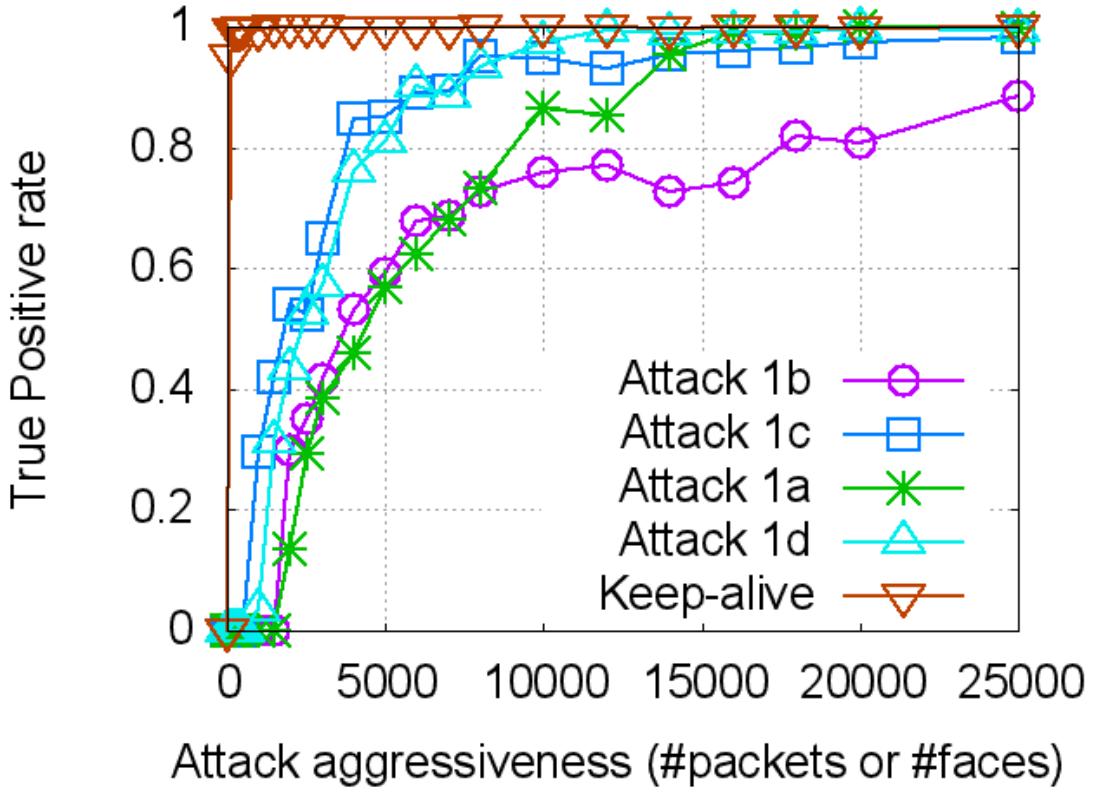


FIGURE 3.6: True Positive Rate

In order to strengthen our evaluation, only one third of the data is used for the training while the remaining is considered for testing and computing the previous metrics. Each experiment is run 10 times including a shuffle of windows for computing the average TPR and FPR. Initial experiments have been done to configure SVM for obtaining a good trade-off between TPR and FPR by using 5000 packets respectively faces as initial data.

In Figure 3.6 the true positive rate is plotted regarding the attack aggressiveness. This corresponds to the number of *Interest* packets for attacks 1b and 1c. The latter 1c is detected easier since the number of faces is multiplied meanwhile by 100 compared to 1b. Once the attack aggressiveness reaches 10,000 *Interest* packets, the TPR is higher than 95%. Similarly, the attack 1d is easier to monitor than 1a as the number of sent *Interest* packets is 100 times higher. Finally, the attack based on keep-alive *Interests* is well detected in any cases. In fact, such an attack last a longer time and is consequently much more visible.

Figure 3.7 shows that FPR remains low in most of cases. For the attacks 1b and 1c, generating a lot of *Interests*, they are never above 2%. The worst values are obtained for attacks involving a lot of faces (attack 1a, 1d and long duration attack) which seems contradictory as these attacks should be recognized easier when the number of solicited faces

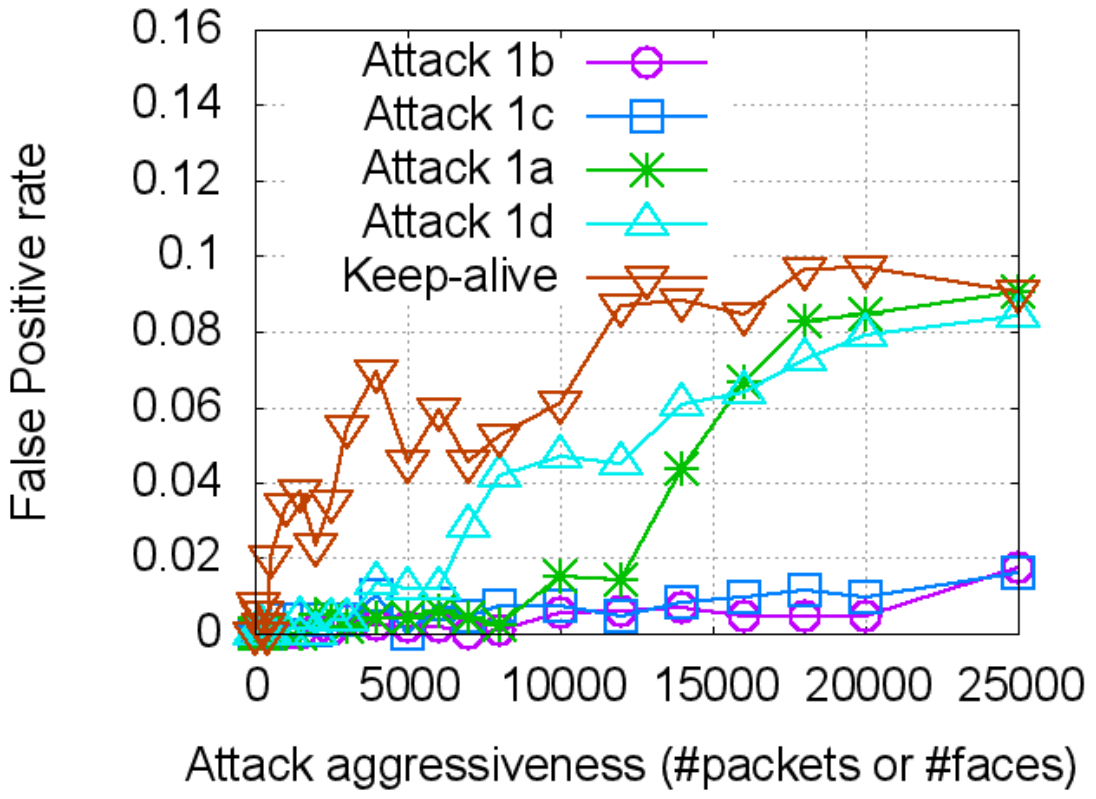


FIGURE 3.7: False Positive Rate

increases. In fact, this is due to two biases that we have investigated manually. First, the monitoring interface provided by CCNx gives metrics that are smoothed regarding the time. Hence, the impact of an attack is still visible on the monitoring interface in several time slots once it is finished (slow decrease of certain values over time) implying false positives. This all the more true with the keep alive which inject *Interest* packets periodically. This finding raises the need of a more accurate monitoring of inner values of CCN nodes for security purpose. Second, attacks involving many faces are longer to execute, which leads to have less windows without attacks. Thus, the training becomes less efficient for normal windows resulting in more false positives. The detection of attacks is only a first step in a security defensive mechanism. Protecting and mitigating attacks is the next step towards securing CCN. we therefore propose a first implementation of a CCN specific firewall in the following section

3.6 Firewall design

As networks expand, security concerns increase in parallel leading to create and deploy defensive solutions. Due to that, firewall already appears in eighties and their popularity have skyrocketed in the nineties. The historical evolution of firewalls during this period

can be found in [79]. Firewalls are generally characterized by the level they operate, like IP or application level, and the way they process. A usual differentiation is stateless and stateful [80]. While the stateless approach is quite simple by trying to match data (*e.g.* a packet) to a pattern, stateful approaches keep tracks of the current connections for applying different policies depending on the state of a connection. Stateless firewalls require de facto less resources and are so good candidates to be deployed at the frontline. That is why in this paper, we firstly investigate this option in the context of CCN by designing a dedicated layer 3 stateless firewall.

Having to deal with more threats and protocols, recent advances in firewall technologies aims to improve their efficiency by determining incoherency in policies [81, 82] and optimizing the rule matching order [83]. The authors in [84] give also a good overview about these topics whereas [85] proposes a dedicated approach for iptables [86]. In addition, recent trends also includes protocol or application specific firewall as for example for SIP [87] or for web services [88].

CCN scheme relocate network management functionalities into every nodes. This includes routing management as well as security as highlighted in this paper. Hence, perspectives includes also collaboration between nodes to facilitate this. The idea of distributed security was also explored in traditional networks in the past. In fact, a early concept of distributed firewall was introduced in [89] and a proof of concept proposed in [90]. In such an example, the main idea is just to exchange rules to apply conjointly. Similar concepts have been studied in [91] using a gateway as an interface to collect and apply filtering rule.

Intrusion Detection Systems (IDSs) are complementary of firewalls. Close to CCN, a P2P IDS is proposed in [92] and a topology aware collaborative approach is studied in [93].

3.6.1 Use case analysis

Regarding the basic functionalities of current firewalls like the open-source iptables [86], the main parameters taken in consideration when writing the filtering rules are the IP address of the hosts, the communication port (linked to the application) and the status of the connection (new, established, etc.). However, we explained in the previous section that such concepts are not available any more in the CCN world, making the design of a firewall challenging. Despite interesting security features, CCN still needs to provide a way for network administrators to properly enforce their security policy.

To drive the design of our firewall and the requirements of the language, we first present a list of use cases describing the different security considerations that an administrator of a Content Centric Network may have. We first remind some general use cases addressed by an IP firewall and analyse them with regard to their applicability to the CCN world. We then introduce CCN specific considerations before defining the needed features of our firewall.

3.6.1.1 IP firewall general use cases

One of the most important use case addressed by firewalls is the filtering of protocols according to the network policy (*IP_ UC1*). A CCN-firewall should be able to filter communications on the same principle, being able to differentiate the different kinds of traffic (web, mail, VoIP, P2P, etc.). For example, web traffic (http) may be authorized in the network while other protocols (ftp, p2p, etc.) are forbidden. A second crucial point of IP firewall is to filter traffic according to the status of the connection (*IP_ UC2*), especially to differentiate traffic issued from inside the network from unexpected traffic issued from the outside which is usually blocked to prevent attacks. The third point is to filter traffic from known malicious IP networks which are blacklisted in order to avoid malware infection (*IP_ UC3*). The network administrator can prevent communications to different blocks of IP address. Finally, serving both for quality of service and security purposes, a firewall can also filter unusual aggressive traffic toward the network (*IP_ UC4*). Typically, in order to mitigate a denial of service, a firewall can filter traffic from an host when the number of packets emitted per second exceeds a given threshold.

When considering the CCN paradigm, some of the aforementioned use cases derived from an IP firewall do not make sense and other must be adapted. In particular, the notion of connection is deprecated. CCN only offers a pull-based connectivity, *Data* can only follow the path previously set by the *Interests*. Any pushed traffic will be discarded by the CCN stack and does not need specific rules on the firewall. Some other use cases are still relevant but must be adapted. Indeed, filtering the traffic based on the protocol is mainly achieved based on the service port number which is not relevant in CCN. Instead, new alternative ways of filtering services based on content name should be defined. Also, filtering malicious network based on IP address blocks do not make sense in CCN. Alternatively, *Data* can be discarded according to the identity of their content provider.

3.6.1.2 CCN-specific use cases

Beyond the adaptation of usual IP firewall use cases to the CCN paradigm, a CCN firewall should also directly rely on CCN concepts to enforce the new security model proposed by CCN. In fact, a CCN firewall can filter content based on two new parameters:

- the content provider: each content should be authenticated with its provider's signature,
- the content name: it is the key parameter to route content and is always given as a plain text string which should be intelligible.

As defined in [58], every CCN *Data* must be signed by the content provider. A CCN firewall shall check the status of the content provider and filter those known as untrustworthy or banned according to the network policy (*CCN_ UC1*). Alternatively, some content may claim coming from a content provider but are not signed by it and should also be dismissed (*CCN_ UC2*). Beyond security considerations, some well-known content providers which content is not relevant within the company network can also be filtered, for example web site providing video-streaming can be blocked in the same way than malicious ones in *CCN_ UC1*.

The second interesting information of CCN is the fact that content name is mandatory and makes sense. This key field shall allow the firewall to perform new efficient filtering based on the content name (*CCN_ UC3*). For instance, specific media type can be excluded from the network (.avi, .mp3, etc.). Some content which name includes specific keywords can also be filtered. This approach can even be enhanced by a semantic analysis of the name. Both filtering (based on the content provider and on the content name) can be mixed (*CCN_ UC4*) to create more fine-grained rules which can, for example, only allow a node to download executable content from few specific (trustworthy) providers. When filtering content based on the content name, the direction of the traffic is important. To avoid the leak of intellectual property by users' mistake, the firewall can prevent some content, whose type is for instance a document (.doc, .pdf, etc.), to be shared externally, while local transfers are allowed (*CCN_ UC5*).

Finally, CCN nodes must also be protected against aggressive traffic rate to preserve the QoS (*CCN_ UC6*). This can be achieved based on the maintained statistics of faces.

A CCN firewall can also change the default routing and caching policies of a CCN node. Based on the content name or content provider, a CCN node can decide to overcome the general rules defined by the CCN stack to only store (*CCN_ UC7*) specific contents.

3.6.2 Firewall features

Based on the aforementioned use cases and of the CCN design, we propose in this section a first set of features for a CCN firewall. By comparison to iptables, the proposed features only concern the filtering capability of the firewall. Contrary to the IP world (iptables) we do not see significant use cases that could be related to features linked to the NAT table (obviously related to IP routing) or the mangle table (overwriting fields in headers).

We presented in section 3.2 the core of a CCN node which is composed of three tables that all have a specific purpose by opposition to the single IP forwarding table. Each one of the CCN inner tables, namely Content Store, PIT and FIB, may be restricted with a set of specific rules. Most of the following features concern the Pending Interest Table which is the key table through which every CCN packet must go through. Initially, only the matching between *Interest* and *Data* is done but the firewall introduces a lot of new features at this point. The Forward Interest Base table that stores the route to contents will not be affected by the firewall in the same way that iptables do not change directly IP routes.

Table 3.1 summarizes the next use cases, compares the IP and CCN world and presents the corresponding firewall features and the corresponding syntactic elements.

IP_UC	CCN_UC	Feature name	Syntactic element	Rule example
IP_UC1	CCN_UC3	Filtering on content name	`content_name`	interest * **@game*|**play**|**fun**\@ **15** pit drop
IP_UC2	-	-	-	-
IP_UC3	CCN_UC1	Filtering on content providers	`provider_sign`	data * * 0 **123456789ABCDEF;FFFF0000AAAA** pit drop
IP_UC4	CCN_UC6	Filtering on heavy traffic	`r_face`	**face 200**
-	CCN_UC2	Filtering on bad signature	`sign_check`	data * * **1** * pit drop
-	CCN_UC4	Composition of filters	`match_data`	data * **@game*|**fun**\@ **0** **0** **123456789ABCDEF;FFFF0000AAAA** pit drop
-	CCN_UC5	Filtering on content direction	`direction`	interest **in** \@\.doc$\@ 0 pit drop
-	CCN_UC7	Filtering of stored data	‘‘cs’’	data * * 1 * **cs** drop

TABLE 3.1: Relationship between use cases, firewall features and language.

3.6.2.1 Filtering on content providers

The signature of the content provider is mandatory for each CCN Data packet. More precisely, every content creates a mapping triple:

$$M_{(Name,Content,Provider)} = (Name, Content, Sign_P(N, C))$$

The signature includes a cryptographic digest encrypted by the publisher's private key, the public key itself or its location. To filter traffic based on content provider, the firewall must retrieve information on the provider's key from the signature and compare it with a blacklist of keys. Alternatively, to improve the security, a whitelist can also be created so that only content from authorized providers will pass through. Data packets that fail the check will not be transmitted to the face and the corresponding Pending Interest is deleted.

3.6.2.2 Filtering on bad signature

The validity of the signature can also be checked while processing the signature. If, being given the provider key, the content and its name, the firewall is not able to find the same fingerprint, the content can be discarded as its source can not be checked.

3.6.2.3 Filtering on content name

As illustrated by figure 3.1, the two types of CCN packets include the name of the content as a plain-text information. By analysing the different keywords composing the name thanks to regular expressions, the firewall can filter content based on a blacklist of the different pieces of information composing the content name, more precisely: the name prefix, keywords and the file extension. Interest packets that fail the check will not be written in the Pending Interest Table and Data packets that fail will not be transmitted to the requesting face. The corresponding Pending Interest is deleted.

It would be useful to filter all content with a given semantic, for example, to prohibit paedophilia-related content. Thus, our firewall can consider additional words that are semantically close to those defined by the user in the firewall rules. A semantic distance allowed can also be defined by the user.

3.6.2.4 Composition of filters

The two previous parameters (content provider and content name) must be considered together to allow the firewall to use more complex rules that can filter, or authorize, specific content type or content name according to the provider.

3.6.2.5 Filtering on content direction

The firewall should be able to differentiate local traffic (*Data* published locally or under the company's prefix) from external traffic.

3.6.2.6 Filtering on heavy traffic

The CCN stack includes a strategy layer which is in charge to maintain statistics for each active communication faces in order to select the best available communication channel. Based on these available statistics, the firewall can detect faces showing an abnormal activity in case of bottleneck and prevent DoS. This can be achieved by decreasing the rate of Interests written in the PIT by aggressive communications. Alternatively, as Interests are also used to control traffic stream, like a reception window, the firewall could overwrite the field of cumulative requested Interests to reduce the consumed bandwidth.

3.6.2.7 Filtering of stored data

The caching policy can easily be enforced by the firewall. To proceed, the same rules written with the firewall language can be applied to the Content Store in addition to the PIT. More precisely, if a CCN Data packet is eligible to be stored in the CS according the the caching algorithm (most recent, most frequent, etc.), the firewall can then check the provider signature and/or the content name according to the defined rules before caching the Data.

3.6.3 Rule definition language

For the ease of use and the readability of the rules, we deeply inspired our syntax from the one used in iptables. Hence, the firewall rules can be defined according to the following grammar expressed using the Augmented Backus Naur Form [94].

```
1 rule = r_interest | r_data | r_face
2 r_interest = "interest" SP direction SP match_interest SP "pit" SP action
```

```
3 r_data = "data" SP direction SP match_data SP ["cs"|"pit"] SP action
4 r_face = "face" SP number
5
6 direction =  "*"|"int"|"ext"
7 action = "forward"|"drop"
8 match_interest = content_name
9 match_data = content_name SP provider
10 content_name = "*"|reg_exp
11 provider  = sign_check SP provider_sign
12 sign_check = "0" | "1"
13 provider_sign = "*"|first_sign *next_signs
14 first_sign = hex_value
15 next_signs = ";" hex_value
16
17 reg_exp = "" re_posix "" number
18 re_posix = <a standard posix regexp>
19 hex_value = 1*hex
20 SP = 1*%d32 ; one or more space characters
21 hex = "A" | "B" | "C" | "D" | "E" | "F" | "a" | "b" | "c" | "d" | "e" | "f" | digit
22 number = 1*digit
23 digit = "0" | "1" | "2" | "3" | "4" | "5"| "6" | "7" | "8" | "9"
```

As highlighted, there are three possible rules: for filtering interests (`r_interest`), data (`r_data`) or faces (`r_face`). In fact, the last case allows the firewall to filter heavy traffic by limiting the global number of packets per second defined by `number` line 4. For other types of rules, we can first filter on the direction as mentioned in the previous section using `"int"` or `"ext"` to indicate respectively whether the content is or was asked by the node itself or another one. Also, the actions are the same for these rules, *i.e.* `"forward"` or `"drop"` on the PIT table which means that the firewall will forward or not specific *Interests* in line 2 of the language definition which is equivalent to add or not an *Interest* in the PIT table. In line 3, the firewall can decide to drop certain *Data* packets. At a first glance, it might be unfair to drop such *Data* since it means that the corresponding interest has been accepted previously. However, an *Interest* can return *Data* with a different name when using the dynamic generation of content upon request, as mentioned in section 3.2.In addition, a rule on *Data* can also control the caching policy by preceding the action by `"cs"`. An *Interest* can only be matched regarding the requested content (line 8). It is basically a standard POSIX regular expression delimited by "@" where `number` (line 17) precises the level to consider for extending semantically the regular expression.

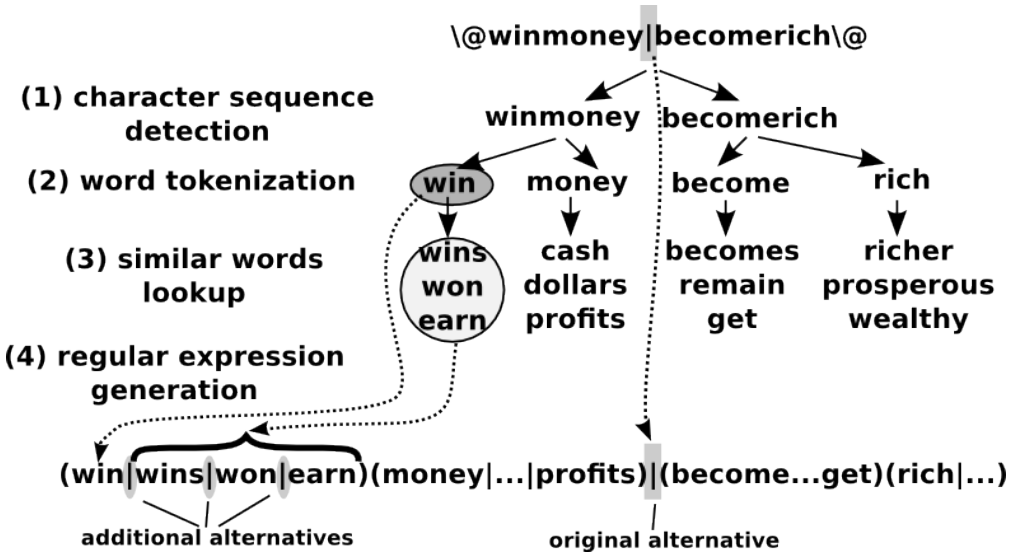


FIGURE 3.8: Semantic extensions – short example

For filtering content, it can also be done regarding the provider by checking if the signature is valid (`sign_check`) and/or on the provider itself identified by it signature , `provider_sign` which is a sequence of hexadecimal characters. As noticed in the grammar, we use the space characters (`SP`). There also several rules using the character `*` which is a shortcut for a regular expression matching everything avoiding to use the delimiter "@".

In table 3.1, some examples show how to construct rules related to the different use cases. For example for *CCN_UC1*, data signed by the keys *123456789ABCDEF* and *FFFF0000AAAA* are blacklisted. For helping in understanding the concept of direction, the example in the table means that the node where the firewall is instantiated cannot share files whose the extension is *.doc* as the firewall will drop corresponding interest when they are associated to internal content (*in*). However, the node can forward interests about doc files of other nodes since there is no restriction using *ext*.

3.7 Firewall architecture

To implement the features defined in the language, the firewall provides tools, modifies the CCNx library and relies on a deployment strategy described in this section.

3.7.1 Semantic preprocessing

As highlighted in previous sections, filtering on content name uses regular expressions which can also contain meaningful words like money, game, humour, etc. Assuming the goal is to filter names like winmoney or becomerich which, in the current Internet, could be used in a URL to attract people in malicious websites hosting malware, the semantic extension corresponds to also filter related content names like for example: wineuros, windollars, earnmoney, bericher, etc. This is essential in CCN as it provides a great freedom in naming and announcing content. To accomplish that, there are four steps as highlighted in figure 3.8:

1. sequences of 3 or more characters are extracted. This allows to only extract potential meaning full words by discarding digits or symbols of the regular expression like {, } or ^ as well as irrelevant very short sequences of characters.
2. sequences of characters are segmented as real human readable words
3. for each extracted word, a list of similar ones is created
4. for each list of similar word, the regular expression with alternatives is created integrating the original word as well and replaces the original word itself in the original regular expression. This is illustrated in 3.8 where the original pipe character (surrounded by a grey square) is kept.

For the word segmentation, the technique proposed in [95] is employed. It finds the optimal word segmentation by successively dividing the sequence of characters in two parts while multiplying the probability of each word corresponding to their frequency in text samples.

Searching for similar word is done through DISCO [96] relying also on text samples like Wikipedia[1]. It computes the relatedness between two words $w1$ and $w2$ by finding the number of co-occurrence of these words with a third one $w3$. Assuming all possible $w3$ which are close to $w1$ and $w2$ (2 intermediate words maximum), the relatedness is computed using the mutual information metric. Using the same method, DISCO is able to get the n most similar words to another.

Therefore, when a strictly positive number n is specified after a regular expression (`number` at line 3.8), this leads to generate the n most similar words of each word extracted at step 2 in the previously described process. It is important to note that some irrelevant word might not have been filtered by the first step like "abcde" but in this case

[1] http://www.wikipedia.org, accessed on 08/08/12

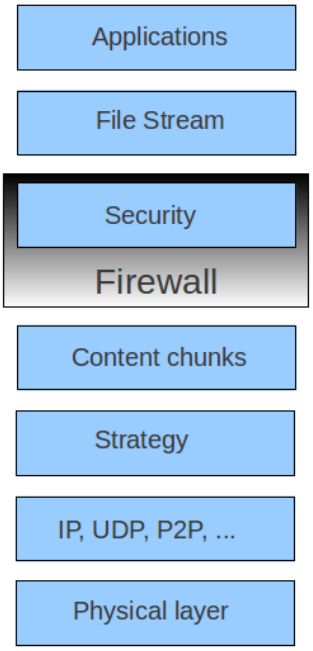


FIGURE 3.9: Firewall implementation within CCN stack

DISCO will not find any similar word and so alternatives will be generated in the final regular expression.

As such semantic algorithms are resources consuming, they cannot be executed when the firewall is running. Therefore, it has to be done as a preprocessing, *i.e.* the original rules are read and extended.

3.7.2 Integration within the CCNx library

3.7.2.1 Placement of the firewall within the CCN stack

Given the CCN stack scheme from [58] our firewall works within the Security layer and process directly content chunks as illustrated in Fig. 3.9. The Security layer as defined in [58] will handle authenticity of the incoming chunks and our firewall can request signature verification which it will handle and give the result to our implementation. In order to provide some flexibility and modularity, we integrated the code of our firewall within the CCN demon so that it is triggered when the CCN inner tables (PIT and CS) should be modified. This induces only minor changes to the CCN standard implementation as well as provides flexibility for further updates. Our firewall captures a packet once it arrived from a face and then applies any given rule on it before letting the standard process performing updates on the tables.

3.7.2.2 Implementation

In order to integrate the rules preprocessed by DISCO we create a simple linked list which holds the rules in the memory. These are loaded depending on what type of rule they correspond to i.e. a face rule will be stored differently than an interest rule. Furthermore there is a matching algorithm which checks if one or more rules matches with a given CCN Data or Interest packet. These functions are provided in the *ccnd_firewall.c* and *ccnd_firewall.h* files.

In order to load the rules at runtime we modified the *ccnd.c* source file such that when a new instance of ccnd is created, by the method *ccnd_ create*, it will load the rules stored in a text file whose path is given as parameter. We also modified the *ccndstart* script launching the demon such that it allowed us to indicate the path to the rules.

We finally integrated a piece of code in *ccnd.c* which is called each time a CCN packet is processed. For this we included our matching algorithm call in the *process_ incoming_ interest* and it will check if the interest just being processed matched any rule provided beforehand. We apply the same method but with focus on data packets in the *process_ incoming_ data* method.

3.7.3 Management

3.7.3.1 Default security policy

One of the first things to do when configuring an IP firewall is to set the default security policy that will be applied in the absence of matching rules. The settings range from the most restrictive rule: *deny all, accept on match*, to the most permissive rule: *accept all, deny on match*. For our CCN firewall, we also provide the administrator with such configurations. The recommended default security policy is to deny all, except packets signed with the public key of the network administrator. This single exception is to give a way for a node to always have its firewall rules up to date.

3.7.3.2 Deployment and administration

To tackle security issues created because of nodes mobility in companies (laptop computers, smart-phones, etc.), we recommend a deployment of the firewall on each node rather than only on nodes providing the Internet connectivity. Upon installation, the firewall software is given the public key of the network administrator to be able to authenticate its content. Then, the rules implementing the security policies of the company can be

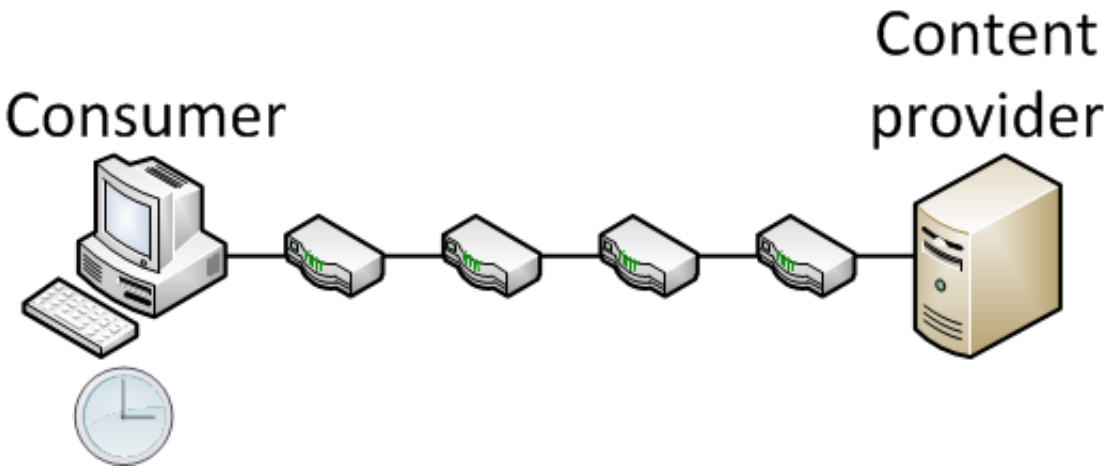


FIGURE 3.10: CCN firewall evaluation architecture

updated on a regular basis. Firewall rules are defined as a content that is requested by the firewall periodically. The rules are applied if properly signed by the network administrator.

3.8 Experiments and performance evaluation

To test our implementation we used the following setup as illustrated in Fig. 3.10. The architecture consisted of six CCN nodes arranged and configured such that the packets need to go through all nodes to be delivered. Each node is a standard workstation with an Intel Pentium 4 CPU running at 3 GHz and 2 GB of RAM. For the evaluation, we used ccnx 0.6.0 release as reference.

One node named "Content provider" provides the content used for our evaluation, some simple raw files. The intermediate nodes served merely as routers without caching repository. Their function was only to allow the "Consumer" node to reach the content provider. Finally the last node on the path was the node requesting our testing content. This is also our measuring point for our evaluation. Our firewall implementation was deployed on all the nodes, to follow the recommended deployment strategy. We performed two evaluations based on this setup, each time starting CCN on all nodes with the same route configuration and requesting the same content.

In a first attempt to evaluate the performance of our implementation, we measured the end-to-end transfer time for two different binary files with a size of 500 MB and 1 GB. We requested the files several times but for each time we increase the number of rules to see if there is an impact on the transmission time. The rules are designed so that the

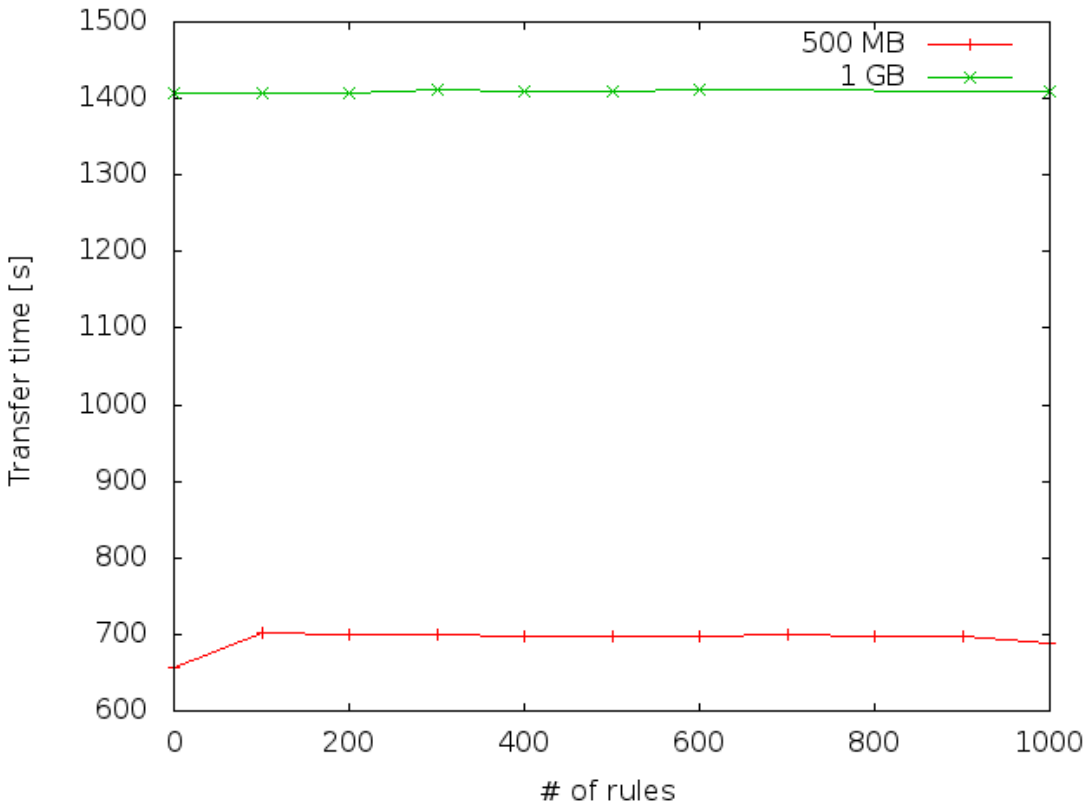


FIGURE 3.11: Impact of the number of rules on transfer time

Data packet is accepted when matching the very last rule so that we are sure that they are all processed. The results obtained from this evaluation are given in Fig. 3.11. It shows that there is no visible impact when we apply our firewall onto the standard CCN implementation, even on the most demanding configuration (1000 rules on each node, 1GB file).

A second evaluation was performed in order to compare the performance of a standard CCN implementation and another including our firewall. This experiment was performed several times to obtain significant values for a later comparison with our implementation. For each observation, we transfer the same binary file of 500 MB to obtain a mean transfer time. Later on, we performed several experiments, each with a fixed number of 1000 rules, with our implementation to have comparative values to the standard CCN implementation.

Our obtained results are shown in Fig. 3.12. We measured the end-to-end transfer time from the content provider to the consumer node. We checked if the obtained values represent expected results by applying a normal distribution. For this purpose we calculate the mean transfer time and the standard deviation for both setups. The results are visible in Table 3.2.

	Standard CCN	Firewall modification
mean μ	684.9718	697.1672
std. deviation σ	1.3463	1.5437

TABLE 3.2: Observation results

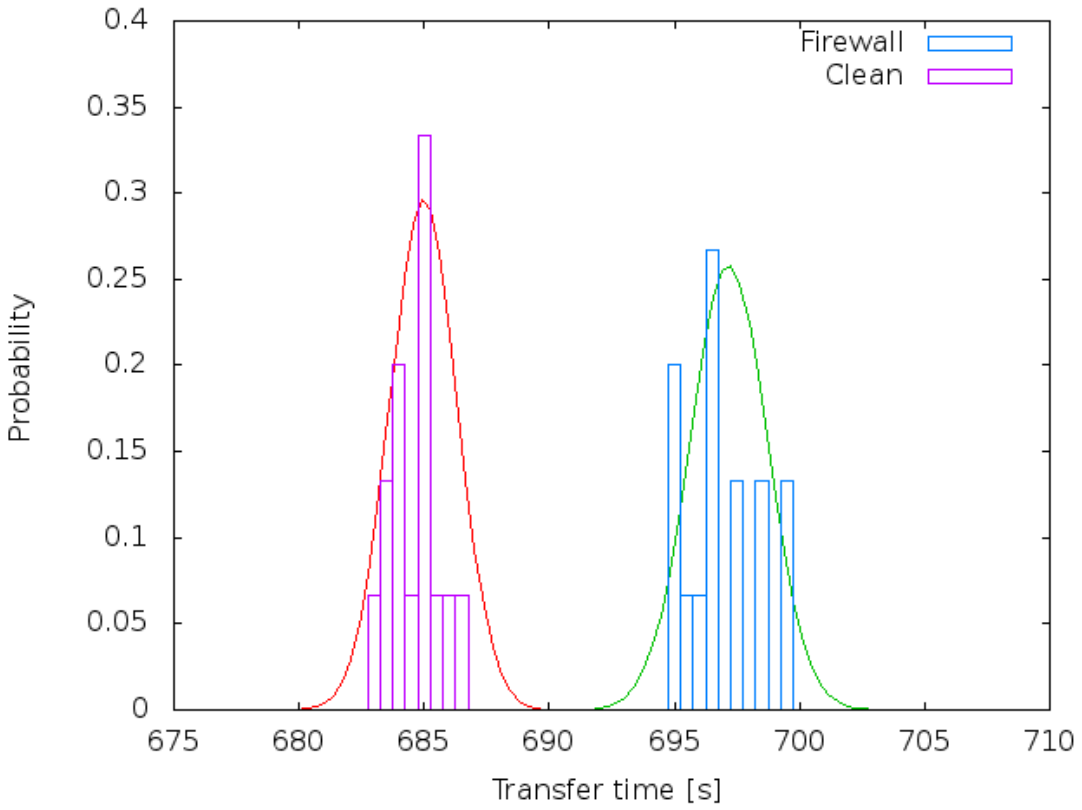


FIGURE 3.12: Impact a the firewall with 1,000 rules on transfer time

As visible from the graph both the clean CCN implementation and our additional modification to it follow a normal distribution and confirmed by applying Chi-square and KS-test on both measured sets obtaining a confidence level of 91% for our implementation and 85% for the standard CCN implementation. From our measured observation we can deduce that the average overhead on the transfer time induced by our implementation is the difference between both averages: $697.17 - 684.97 = 12.20$ s, which corresponds to 1.75%.

3.9 Conclusion

We presented in this Chapter a first monitoring architecture for CCN. While this new paradigm worth being investigated for the sake of future Internet, it also raises new

management challenges we presented in this paper. Among those, we investigated one of the most important problem affecting CCN devices: the possible denial of service through the flooding of the PIT table. To address this issue, we used the monitoring features of the reference implementation coupled with a classification algorithm based on SVM and which can efficiently detect such attacks with a small computational cost. In fact, we implemented and experimented different attack strategies to perform DoS and all of them can be detected with very low error rates, which could be even lower with a more accurate report of operating values.

Our research opens directions for a lot of future works. First of all, our detection mechanism will be implemented within the CCNx libraries in order to enable real-time detection and the usage of associated countermeasure to mitigate attacks. We will then extend our monitoring architecture to monitor the other tables (FIB and Content Store tables) to detect other types of attacks. Finally, we want to extend our test-bed and generate more realistic traces including traffic from different applications. Attack detection was the focus point of our approach and future work will also focus on attack prevention.

We also proposed a first firewall dedicated to the CCN paradigm, from the use case study of the security needs of a CCN administrator, to the grammar definition and the implementation within CCNx. Since this paradigm relies on content based routing, where the name of the content and its authentication are key aspects, we used these opportunities to provides innovative features compared to a regular IP firewall. In particular, we propose to leverage semantic tools. Performances evaluation are very promising since the message throughput degradation is lower than 2%. In future work, we plan to investigate potential optimization, for example by being inspired by rule reordering research for standard firewalls or using bloom filters for accelerating rule lookups.

Chapter 4

Identifying Abnormal Patterns in Cellular Communication Flows

Analysing communication flows on the network can help to improve the overall quality it provides to its users and allow the operators to detect abnormal patterns and react accordingly.

The chapter considers the analysis of large volumes of cellular communications records. We propose a method that detects abnormal communications events covering call data record volumes, comprising a country-level data set. We detect patterns by calculating a weighted average using a sliding window with a fixed period and correlate the results with actual events happening at that time. We are able to successfully detect several events using a data set provided by a mobile phone operator, and suggest examples of future usage of the outcome such as real time pattern detection and possible visualisation for mobile phone operators.

Furthermore we analyse the same data set of cellular communications records but using a different approach. By using the collected call and message exchanges we present a method based on the PageRank algorithm that also detects abnormal communications events. Taking the number of calls and the total call duration as parameters we use a weighted version of the PageRank algorithm to further investigate the influence of these parameters on the connected network graph. We proceed by correlating the results obtained with events happening in the respective region and at that time.

4.1 Introduction

Call Data Records or Call Detail Records (CDR) represent the information a mobile phone operator typically gathers to ensure correct billing for its customers. A CDR is the collected information of a communication transaction such as caller and callee id, duration of the transaction, connected base station and other variables and processing of these variables can be used to analyse the current communication infrastructure and extract information to provide better service or offer more customer- specific applications, and to detect shortcomings of its current network layout.

In this Chapter we describe a model based on CDRs. This model is based on the communication flows between base stations and describes both the communication between users and the communication between base stations. We define a metric which allows us to detect abnormal communication patterns in the overall CDR corpus which, when correlated to real events, allows the creation of a detection mechanism that can highlight such patterns in near real time. Additionally we support our claim by evaluating a country-level data set containing CDRs for a period of 150 days and, after applying several data transformations, successfully detect events which correlate with the communication pattern in the CDR corpus. This gives us a local situational awareness of the ongoing communication of a base station.

In order to grasp the global impact of communication flows we apply the PageRank algorithm that to analyse this impact the whole network and also enables us to detect detect abnormal communication patterns in the overall CDR corpus. When correlated to real events as with the previous metric, it enables the creation of a detection mechanism that can be utilized to monitor the traffic in near real time. To support our claims we test our the Page rank method against the same country-level data set and detect communication pattern in the CDR corpus which we correlate events with.

4.2 Related work

In [97] the authors describe a system that allows its users to explore and analyse of huge quantities of CDRs. They use several visualization techniques to provide an overview of the whole network without losing the details. Furthermore they correlate findings of their tool with news and events that happened during the evaluation period. Associating events of all kinds (e.g., parades, public concerts, soccer matches, traffic congestion, riots, protests etc.) with exceptional spatio-temporal patterns from CDRs is illustrated in [98]. CDRs and weather data can be used to predict climatic changes, as shown in [99]. The authors are able to infer a model which predicts precipitation using exploratory

factor analysis to reveal latent variables from the weather data and spectral analysis of the recurring mobile and weather data. Clustering is another common approach to extracting information from large amounts of data. The authors in [100] use such a method, symbolic clustering, to assign profiles to users and base stations based on their regular behaviour (weekly and daily activities). These clusters of similar activities can be used to offer better service to customers with similar behaviours, and to detect base stations with heavy load over certain periods of time. In [101] and [97], the authors develop tools which help to explore and analyse large volumes of CDRs using network graphs. They present a web-based prototype implementation which allows its users to gather statistical information on the current network topology. Another important research topic has always been mobility models. Creating a Thiessen polygon[1] based on the base station location and aggregated information gathered from the road network and population density, the authors of [102] infer movement patterns based on CDRs which may indicate critical locations in times of crisis or emergency. The authors of [103] create a social network graph based on CDRs correlated with the billing addresses of customers. They show that communication between cities can be defined by a gravity model where the communication intensity between two cities follows a similar distribution to the product of their sizes divided by the square of their distance. In [104] the authors successfully create a model of unusual social events by analysing CDRs using a Bayesian location inference framework and are able to determine which users are attending a particular event. They support their claim by showing concrete examples for different locations.

All these works show interesting approaches either to visualizing huge amounts of data or to extracting pattern from communication flows. Most of them focus only on one parameter when extracting information or analyse the network situation based on events that have already happened . In our approach we consider several parameters for extracting communication patterns and directly detect events resulting from abnormal behaviour. We analyse the number of calls as well as their duration and the users making the calls connected to the base station. Additionally we also analyse the location of these base stations and detect correlation effects among them.

4.3 Data modelling and anomaly detection

Our model is based on CDRs, using call and SMS communication to analyse the behaviour of the population during a given time period in order to differentiate between

[1] Thiessen polygon: http://en.wikipedia.org/wiki/Thiessen_polygon

normal "everyday traffic" and abnormal, out of the norm traffic. Our model uses aggregated communications flows between base stations, regrouping calls and messages sent over a pair-wise link between two base stations. Additionally we correlate these abnormal patterns with possible candidate events happening at the same time, leading to the definition of a metric which is able to distinguish between these possible states. Once these abnormal patterns have been characterized we can use them to detect similar patterns in later real-time monitoring of the network communication.

4.3.1 Sliding window metric

The CDR corpus comprises information of base station-to-base station communication including calls and their duration as well as the users connected to each station on an hourly basis. We evaluate the average number of calls per user on each base station and the corresponding average duration over the whole data set to obtain preliminary results for each time slot such as mean values for each time slot. These values, if increasing or decreasing drastically from those for previous slots, can indicate if the general behaviour of the population is changing or staying constant and represent the common daily condition. However to detect a spontaneous increase in the communication pattern, we need to have some prior knowledge of how communication looks in general. In order to obtain those values which we consider out of the norm, we calculate a weighted average using a sliding window with a fixed period to extract the outlying records, which are interesting as they do not reflect the normal behaviour of users during the day. This sliding window is applied to the number of calls X and the total duration of the calls Y where X_i and Y_i represent the value at time instance i. The weighted expected average for time instance i is calculated using a window of 10 time instances representing the knowledge over the past 10 hours, which is compared to the actual average for that time instance.

$$E_i\left[X_{t+1}\right] = \alpha \cdot E_i\left[X_t\right] + (1-\alpha) \cdot X_{i,t+1} \tag{4.1}$$

Where $E_i\left[X_t\right]$ is the expected value calculated over the previous 10 time slots. If the actual value $X_{i,t+1}$ is above or below the expected value by three or more standard deviations ($E_i\left[X_{t+1}\right]+3\cdot\sigma$ and $E_i\left[X_{t+1}\right]-3\cdot\sigma$ respectively) then the outlier is interesting for evaluation, as it is clearly out of the norm, which in our case covers 99.7% of the general behaviour given $E_i\left[X_{t+1}\right]$ and $3 \cdot \sigma$. The same method is applied to Y.

4.3.2 PageRank and Weighted PageRank algorithm

To detect possible patterns we analyse the CDR corpus using the PageRank algorithm introduced in [105, 106]. Using PageRank enables to detect the base stations that have the highest communication traffic flows during a given time slot. If the increase in communication is too important if may be a sign for some abnormal behaviour. Furthermore we correlate these abnormal patterns with possible candidate events happening the same time that can explain these patterns. Once we have categorized certain patterns, we can use them to later detect similar patterns while monitoring of the network communication flows in real-time.

The CDR corpus is used to calculate a PageRank score $PR()$ for each base station during each time slot. It allows to detect base stations with a heavy load of communication flows. If there is a spontaneous increase of the PageRank score on a given base station, it indicates that the activity on this base station is also higher than normal. This can be then flagged as unusual behaviour when compared to the behaviour of previous time frames. For a first evaluation we consider only if the base stations have ongoing connections. In the next step we evaluate the influence of the number of call and their duration on the PageRank score calculus

Let N be the total number of base stations in the network. We denote a base station bs_i a single base station i and $PR(bs_i;t)$ the PageRank score of base station i at time t. We compute an initial score $PR(bs_i;0)$ for all base stations using (4.2).

$$PR(bs_i;0) = \frac{1}{N} \tag{4.2}$$

In each subsequent time slot we proceed by selecting for each base station i a subset $M(bs_i)$ that contains all base stations j which are connected to base station i. Furthermore we need the number of outbound connection of each base station j that we denote with $L(bs_j)$. d is a damping factor introduced in [107]. The initial purpose of this damping factor is to model a user's intention to continue randomly clicking on links as PageRank states that he will eventually stop. The PageRank calculation for each subsequent time slot uses the formula in (4.3)

$$PR(bs_i;t+1) = \frac{1-d}{N} + d \cdot \left(\sum_{bs_j \in M(bs_i)} \frac{PR(bs_j;t)}{L(bs_j)} \right) \tag{4.3}$$

We include the number of calls and their total duration to analyse their impact on the score calculation and thus the impact on the complete network. In order to take these

values into account we modify the original PageRank to include a weight on the inbound and outbound connections where $W_{in}(i,j)$ and $W_{out}(i,j)$ express the weight for the inbound and outbound connections between base station i and j. They are calculated using (4.4) and (4.5) where w_{in} and w_{out} simply express the numeric value of the numbers of calls respectively the total duration of calls that are measured on a given base station.

$$W_{in}(i,j) = \frac{\sum_{bs_k \in M(bs_i)} w_{in}(bs_k;t)}{\sum_{bs_l \in M(bs_j)} w_{in}(bs_l;t)} \tag{4.4}$$

$$W_{out}(i,j) = \frac{\sum_{bs_k \in M(bs_i)} w_{out}(bs_k;t)}{\sum_{bs_l \in M(bs_j)} w_{out}(bs_l;t)} \tag{4.5}$$

The formula (4.6) below is how we finally calculate the weighed PageRank score. Taking into account $W_{in}(i,j)$ and $W_{out}(i,j)$ allows us to analyse the impact of inbound and outbound calls respectively the total duration of calls on a given base station pair.

$$PR(bs_i;t+1) = \frac{1-d}{N} + d \cdot \left(\sum_{bsj \in M(bs_i)} \frac{PR(bs_j;t)}{L(bs_j)} + W_{in}(i,j) + W_{out}(i,j) \right) \tag{4.6}$$

4.4 Experiments and results

4.4.1 Sliding window method

For our analysis, we focused on the first and the second sets to test our metric of the D4D data set introduced in Section 2.3.2. *SET1* consists of 6 GB of textual data, corresponding to around $175 \cdot 10^6$ records while *SET2* contains 2 GB of textual data corresponding to around $55 \cdot 10^6$ records. In a preliminary exercise we summed the number of users connected to a given base station over *SET2* in order to join the results obtained with the records in *SET1*. However analysing data for 150 days can be quite difficult due to the huge quantity of records. In order to make concrete analysis we needed to find a procedure which would allow us to quickly process the data and apply our previously-defined metric without losing the granularity of the data.

We tested several techniques to perform this operation but due to the large amount of data, some methods proved to be impractical and too time consuming. The map-reduce paradigm, introduced in [108, 109] and implemented in the *Hadoop* platform proved to be the best solution for balancing execution time and implementation effort in our case and was therefore used to apply several transformations to the original data to make it suit

our needs. For this purpose we use our local Hadoop cluster, running Hadoop version 2.0.0-cdh 4.30. which implements the map-reduce paradigm, introduced in [108, 109]. The cluster itself is composed of of 4 nodes, each running hexacore Xeon CPU with 2.4 GHz and a total memory of 120 GB. The distributed file system has 27.54 TB of free space. Using the cluster allows to reduce the huge computational task into several smaller tasks that perform the same operation as the larger task but more efficiently.Figure 4.1 illustrates how the map-reduce paradigm operates within the Hadoop infrastructure.

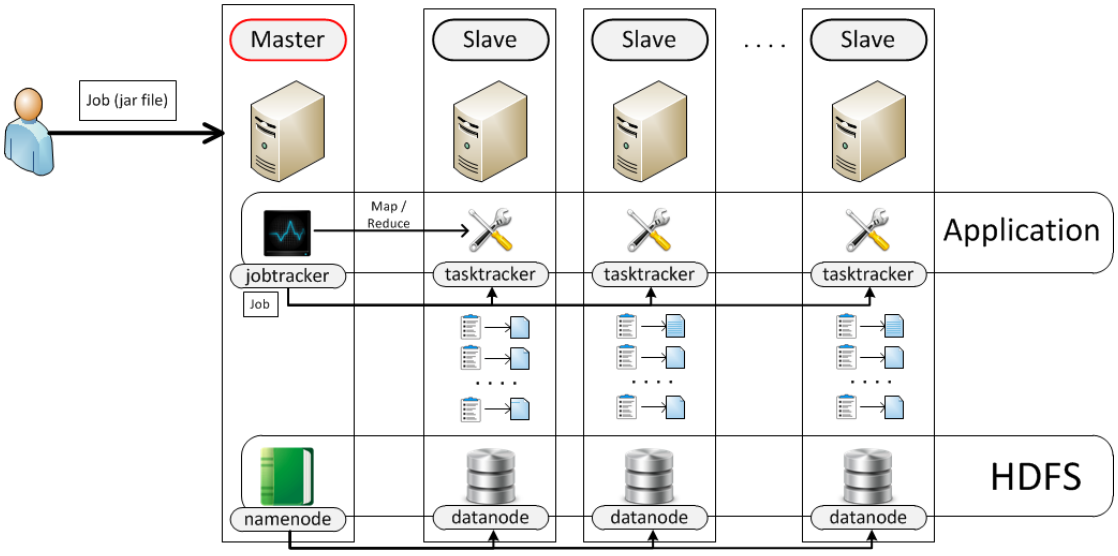


FIGURE 4.1: Hadoop architecture

The resulting dataset was obtained by mapping each time stamp as an individual key and summing over the records in the reduce phase.

The next step was to join the data we had from *SET1* with the data we obtained through pre-processing. Our goal was to obtain records composed of a given time stamp, source and destination base stations and, for each to these, the number of connected users, the number of ongoing calls and their total durations. For this purpose we proceeded in the same way as for pre-processing by creating a mapping for each individual time stamp. In the resulting reduce phase we distinguished between two different types of records. We created for each source and destination base station pair a network flow composed of the number of calls and their duration. We then associated the number of connected users with each given source and destination.

For the resulting dataset we created a new dataset which consisted of the average number of calls per user and the average duration of the calls per user for a given time slot. We applied our metric to this resulting set in order to obtain results. We calculated the weighted average for the number of calls and the duration, then applied the formula in (4.1) to retain only those with a value above or below 3σ. The graphs of Figures 4.2 and 4.3 represent those results. We then summed the number of base stations for

any given time slot and produced the graphs. Certain outliers can be linked to events happened during particular time instances. The points with low values occur because of power outage during that period, and are highlighted with circles. The point, emphasized with a rectangle, corresponding to the highest traffic peak, can be correlated with the appearance of the former Ivorian president, Mr. Laurent Gbagbo, before the International Criminal Court in The Hague[2]. Similarly the triangular highlighted point can be correlated with the beginning of the Malian uprising[3] close to the border.

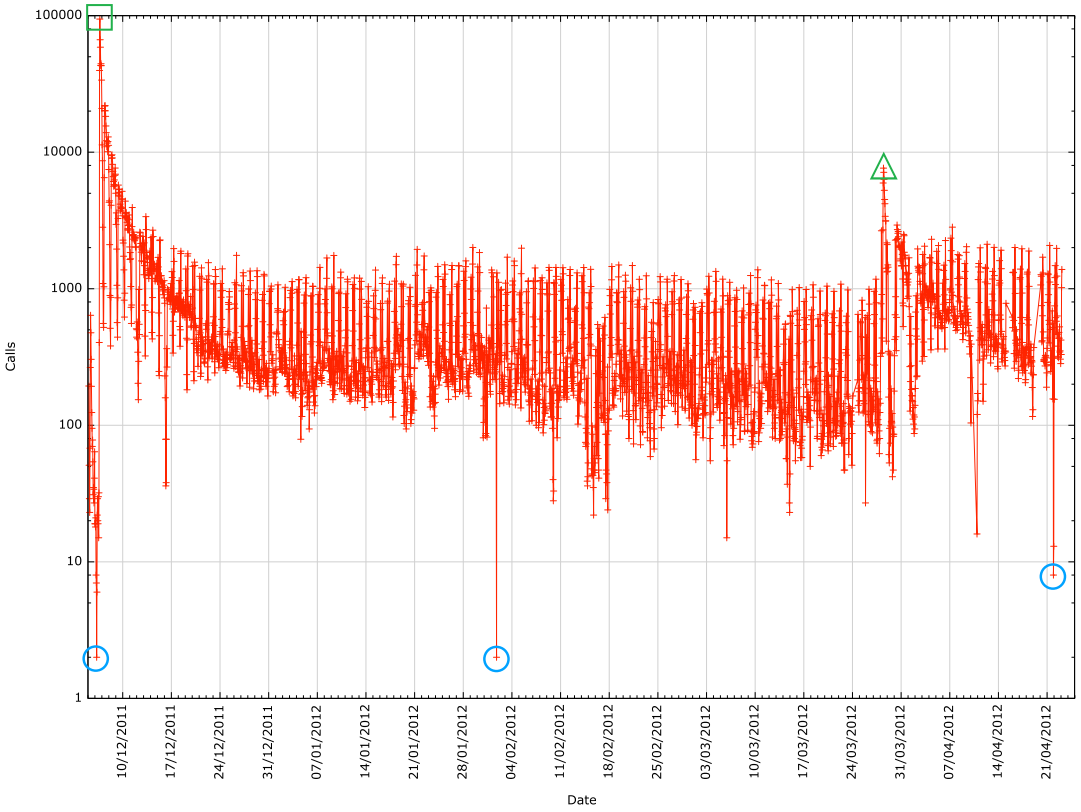


FIGURE 4.2: Number of base station with abnormal ongoing calls during a time slot

We decided further to investigate the period of the Malian uprising. Therefore we selected the nine base stations closest to the border, as illustrated in Figure 4.4. The red highlighted base stations are those with no communication flows and the green highlighted one is behaving quite anomalously among the others. We analysed the average number of calls and their duration on each base station for the period of March to attempt to detect some preliminary signs of the Malian uprising. Figure 4.5 and 4.6 shows the results we obtained.

[2] http://www.guardian.co.uk/world/2011/dec/05/laurent-gbagbo-international-criminal-court1
[3] http://www.breakingnews.com/topic/mali-unrest

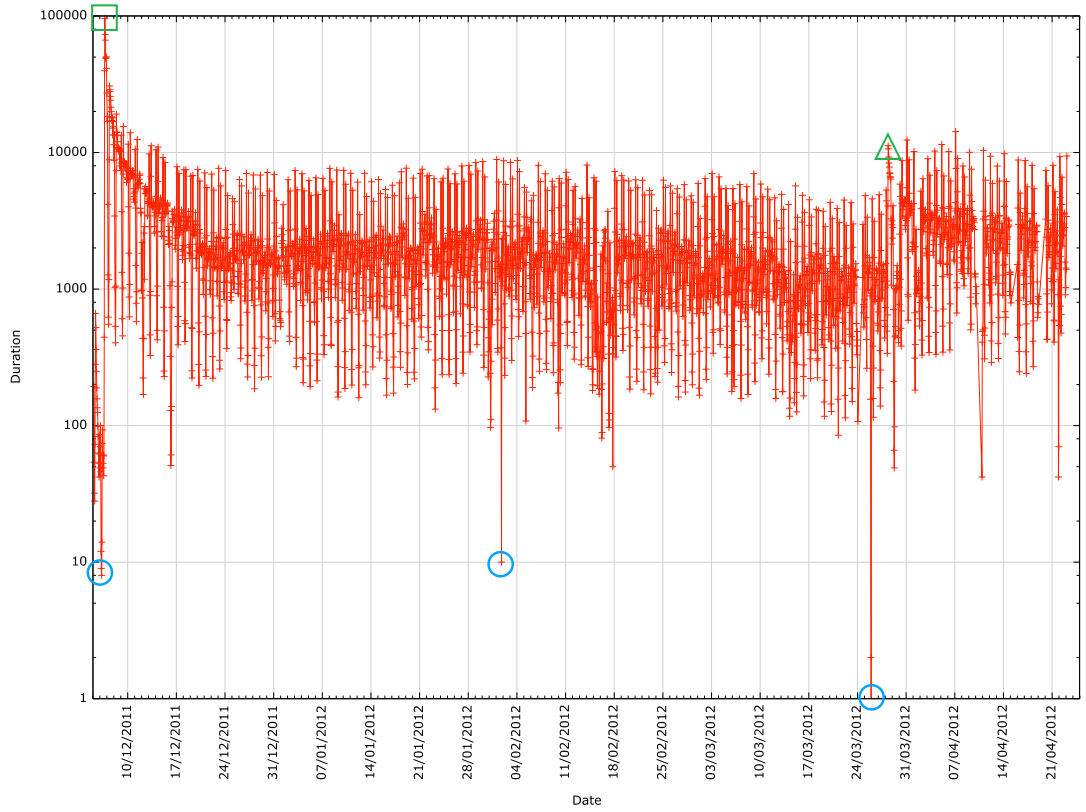


FIGURE 4.3: Number of base station with abnormal call durations during a time slot

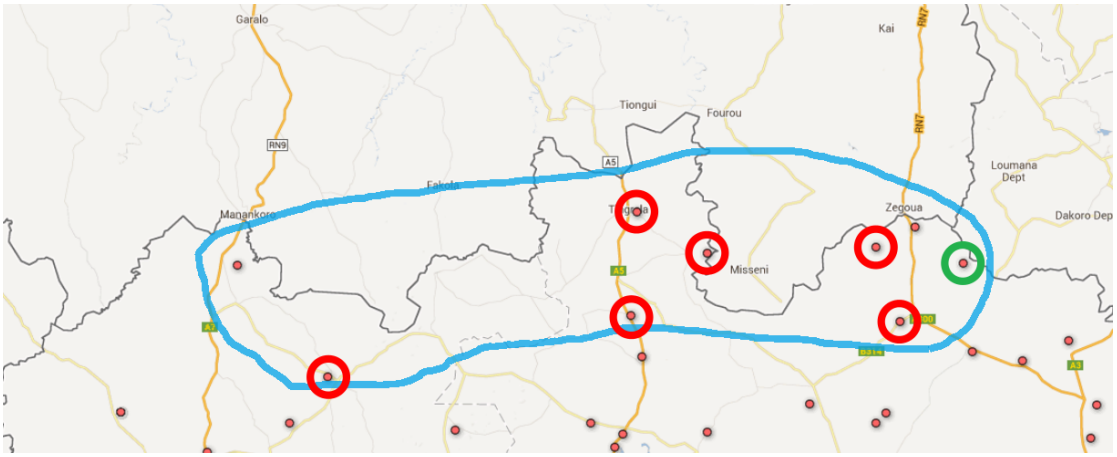


FIGURE 4.4: Base station near the Malian border

An interesting observation is that on the same date that the Malian uprising started, seven out of the nine base stations had no communication flows with any others, and this situation persisted for almost a week. A closer look at that time interval is presented in Figures 4.7 and 4.8. The reason for this outage remains uncertain; however one possibility could be power failures in this area, although certain base stations remained operational. Another possible explanation could be preliminary sabotage acts by the Malian rebels

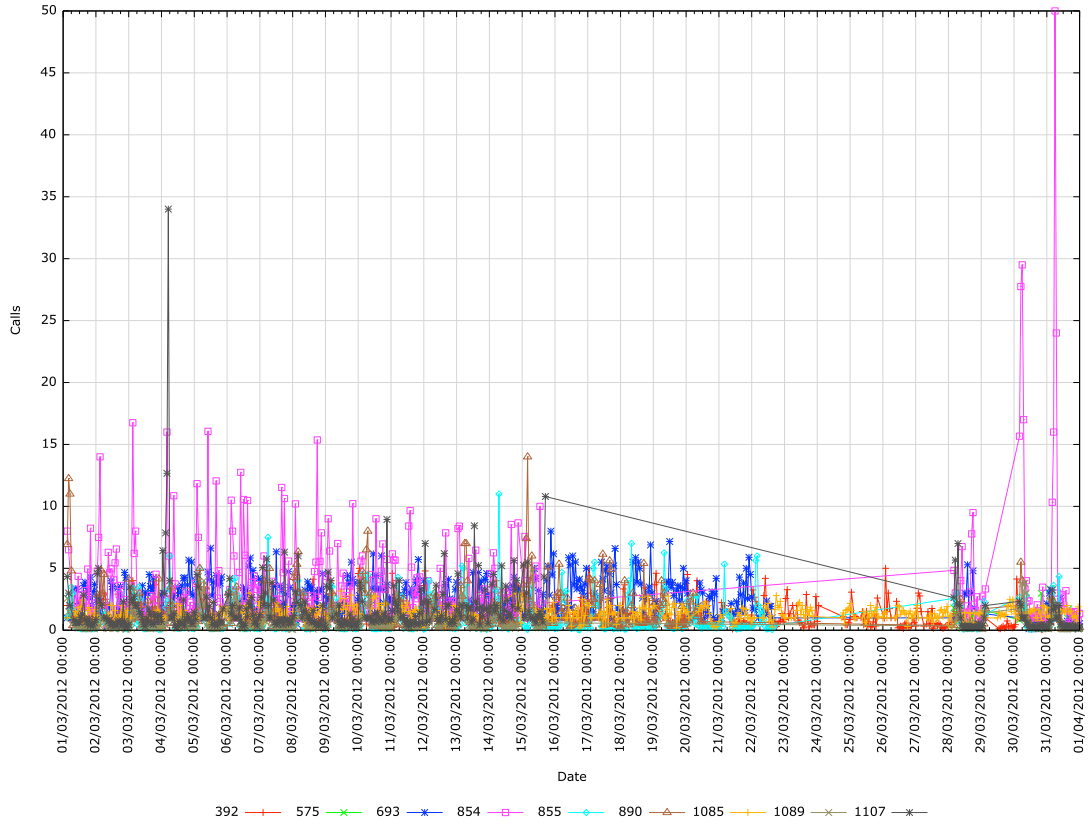


FIGURE 4.5: Number of ongoing calls during March near the Malian border

who were involved in the subsequent Malian uprising. Also one can see that one of the previously shut-down base stations, no. 854 for example, experienced a huge increase in communications. This could be an after-effect of the shut-down period.

We then focused again on the whole dataset for our next evaluation. We extracted the mean values and standard deviation in each time slot for the number of calls and their total duration. This evaluation was performed in order to detect clusters having similar communication profiles. Each month was treated separately. Figure 4.9 illustrates the result. There are two main clusters grouping either a small number of calls with a short duration, group A, or a large number of calls with a long duration, group B. These represent the usual diurnal behaviour of night and day communication. Interestingly the outliers highlighted in group C represent quite a similar behaviour: the duration is quite long and the number of calls is around the global average over the period of time. The interesting factor (not visible in the figure) is that all these calls were made during the same time slot during the night at 2 a.m. but on different dates. We examined other periods for similar patterns and discovered the same recurring pattern during those periods.

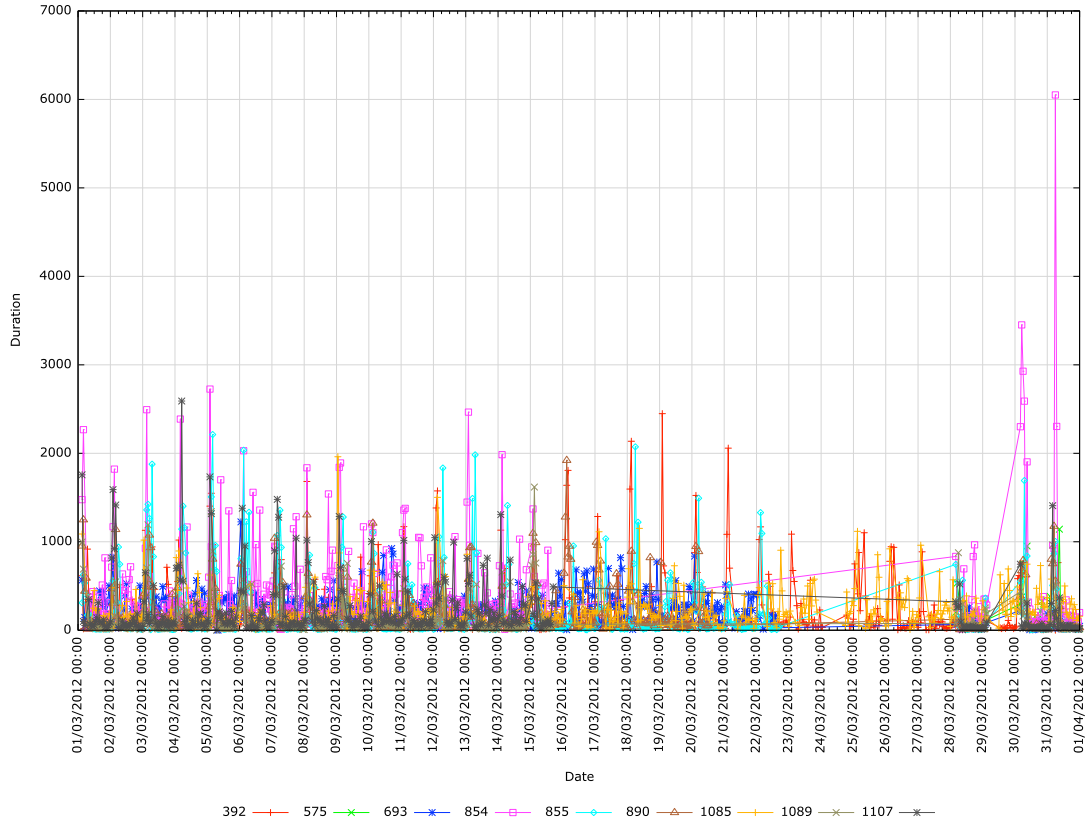


FIGURE 4.6: Duration of on-going calls during March near the Malian border

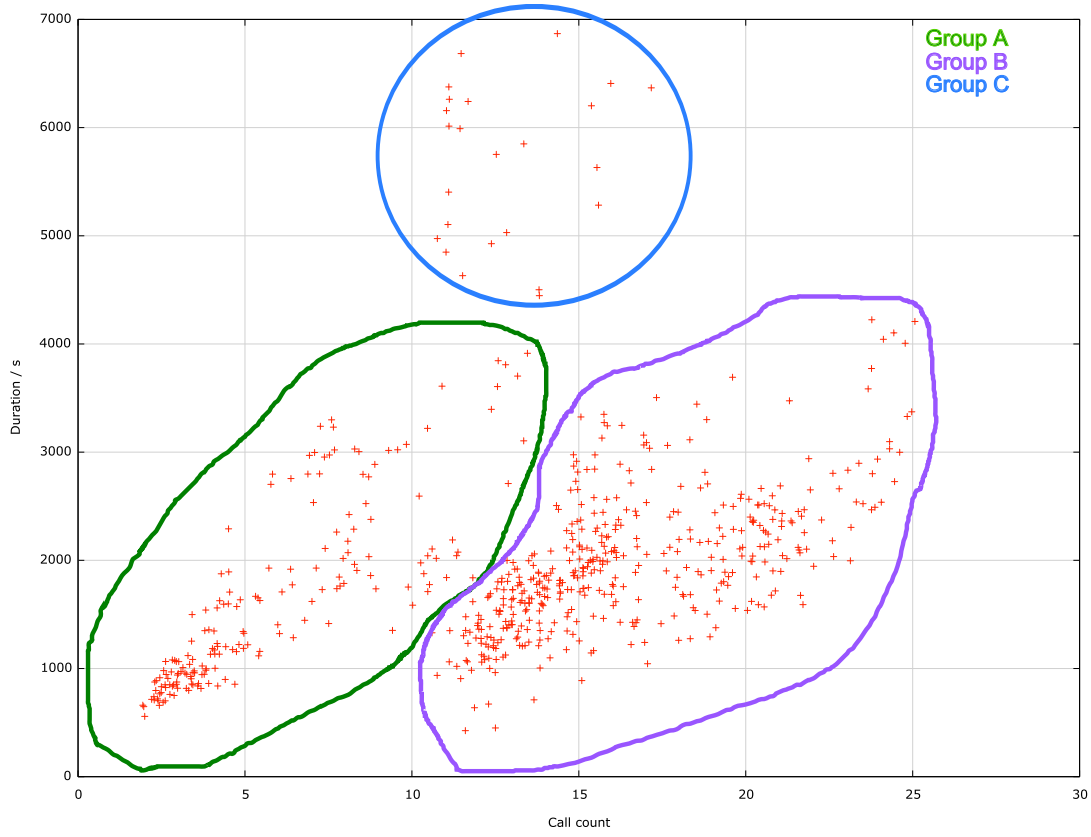


FIGURE 4.9: Mean on-going call count vs. mean call duration for December 2011

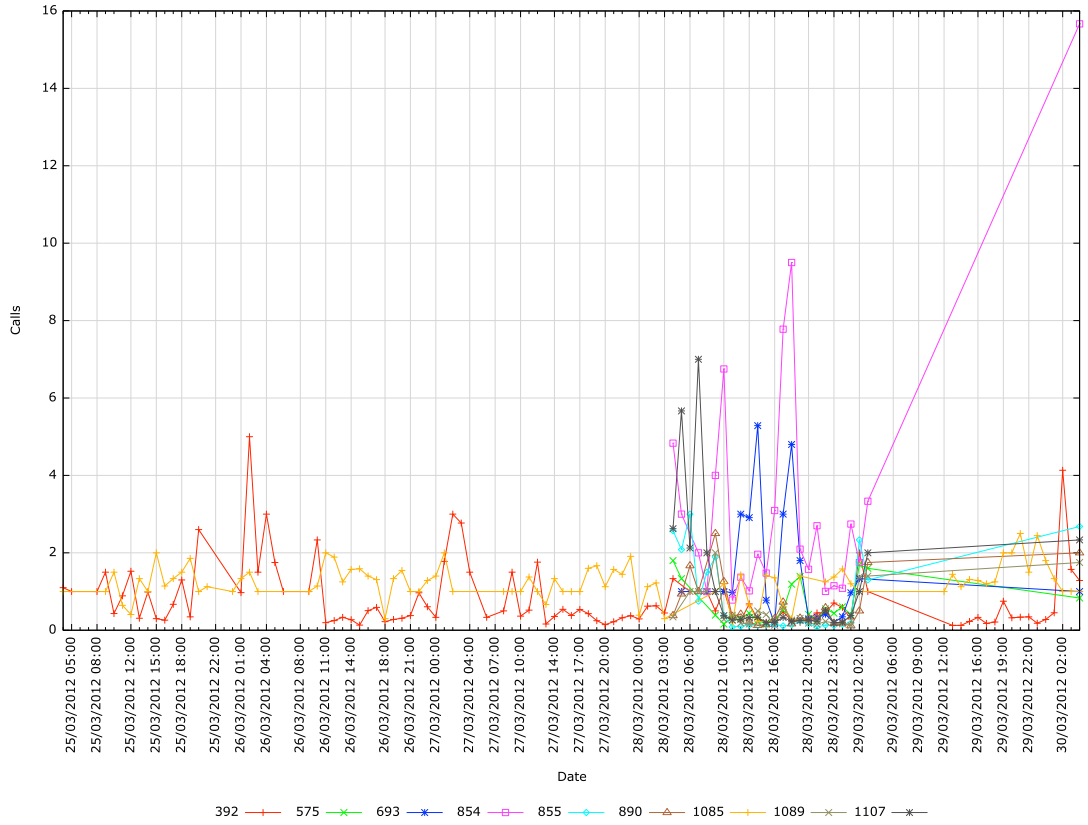


FIGURE 4.7: In-depth examination over the calls for the last week of March

To detect dependency effects among base stations and determine if they have an influence on each other, we performed a Principal Component Analysis to extract those components having the greatest impact on the base stations. PCA[4] allows us to identify the components with the largest variability and we are able to identify base stations that share a common behaviour. We used R software environment for statistical computing and graphics version 2.15.1, to apply PCA to both the number of calls and their duration obtaining the graphs (Figure 4.10 and 4.11) below. The x axis represents the different components detected by the PCA starting from PC1. Figure 4.10 shows that the first principal component (PC1) has the highest impact on all the base stations. At least 12% of all the available base stations are influenced by PC1. Figure 4.11 shows a similar behaviour: again the first principal component (PC1) has the highest impact on all the base stations, with as many as 26% being influenced by this component. We subsequently cross-referenced the results of both analysis limiting us to the twenty most influenced base stations for each PC1 and extracted the top 10 from these . Figure 4.12 shows the top 10 base stations occurring in each PCA, mapped on their coordinates, which, as one might expect, are all located in the former capital, Abidjan.

[4]PCA: http://en.wikipedia.org/wiki/Principal_component_analysis

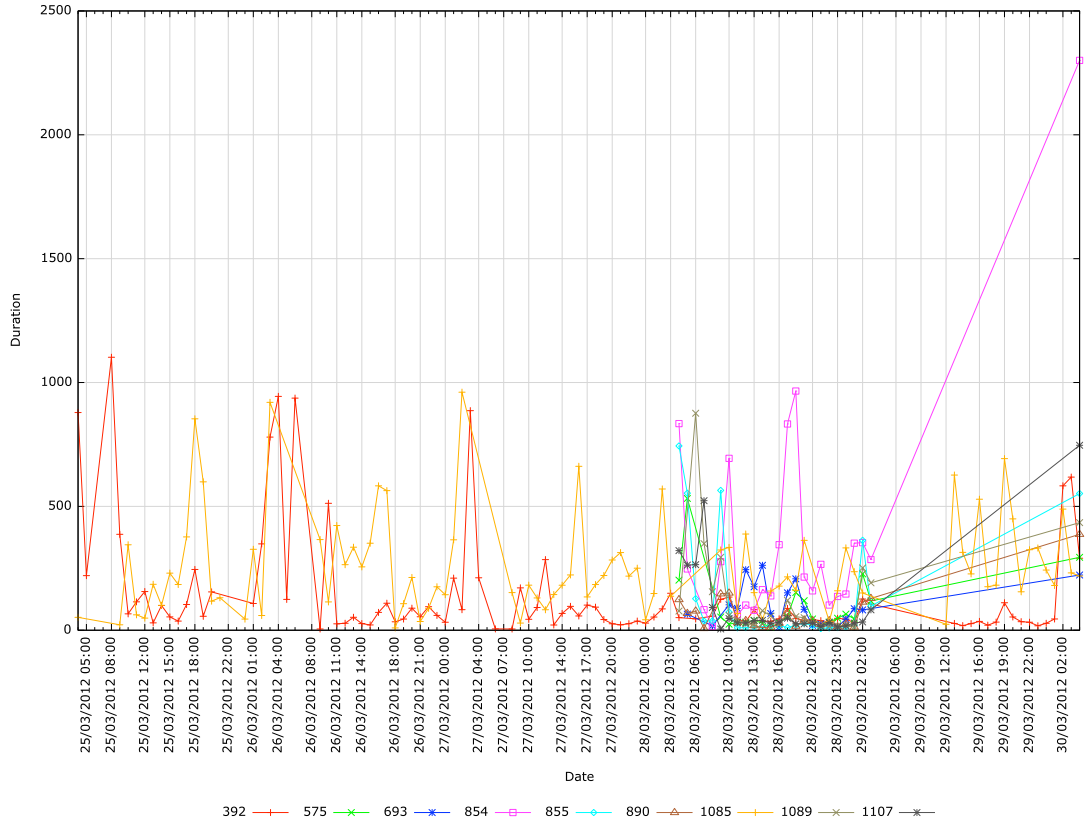


FIGURE 4.8: In-depth examination over the duration for the last week of March

FIGURE 4.12: Top 10 base stations ranked by PC1

4.4.2 Pagerank for anomaly detection

The first task is to count all the outbound connections on each base station and associate these values to each source base station and their connected destinations in order to calculate the PageRank score. Furthermore we average the number of ongoing calls and the total duration with the number of connected users on each base station.

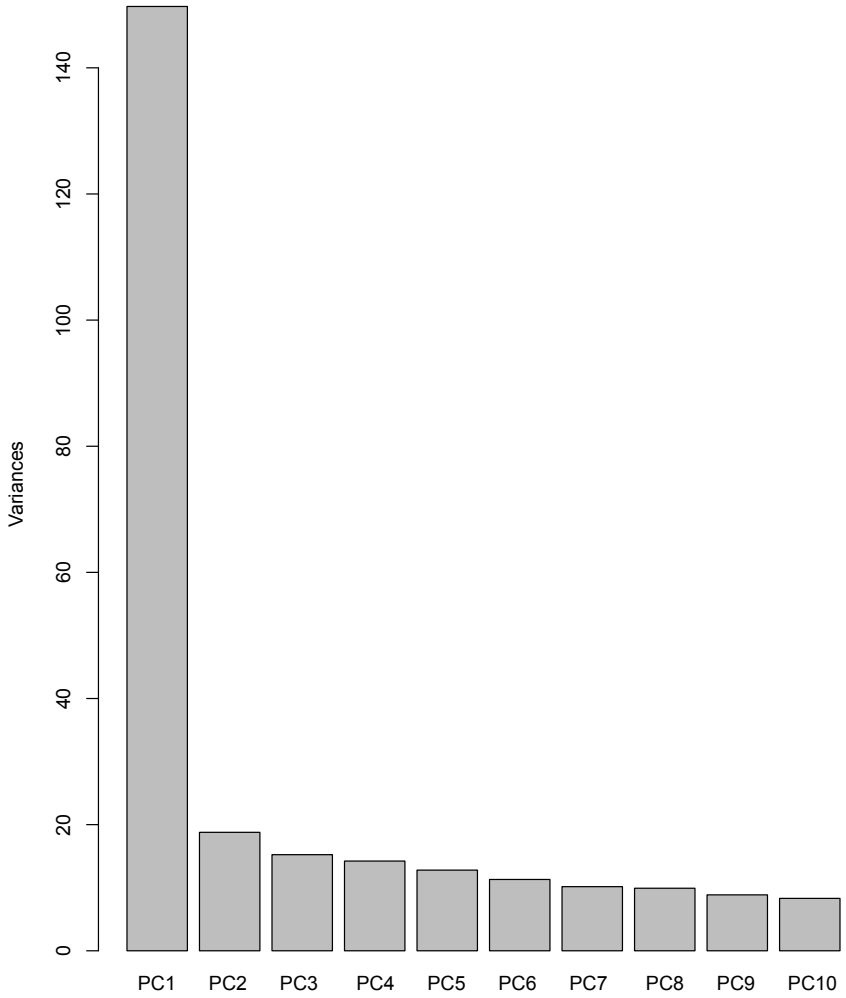


FIGURE 4.10: Principal Component Analysis based on the number of calls

We then proceeded to obtain the mean value for each time slot over all the base stations and a mean value for each base station over all time slots. Furthermore the maximum, minimum and standard deviation were calculated in the same manner. All the figures show the average (red plot), minimum (light blue plot) and maximum (purple plot) values as well as the standard deviation added (green plot) and subtracted (dark blue plot) from the average value. Figures 4.13 and 4.14 illustrates the results employing the standard PageRank score calculation. Figure 4.13 shows the values depending on the time slot and Figure 4.14 depending on the base station. There is a high increase in the overall traffic in the network at the beginning of Figure 4.13 as highlighted with a red circle in the upper left corner of the figure. The highlighted increase, depicted by red circled data points, corresponds to a certain number of base stations located in major cities at specific moment in time. Figure 4.14 highlights in a similar fashion these involved base station. Interestingly after further investigation on possible causes for this increase we concluded that it corresponds to major political event happening in this country[5].

[5] http://www.guardian.co.uk/world/2011/dec/05/laurent-gbagbo-international-criminal-court1

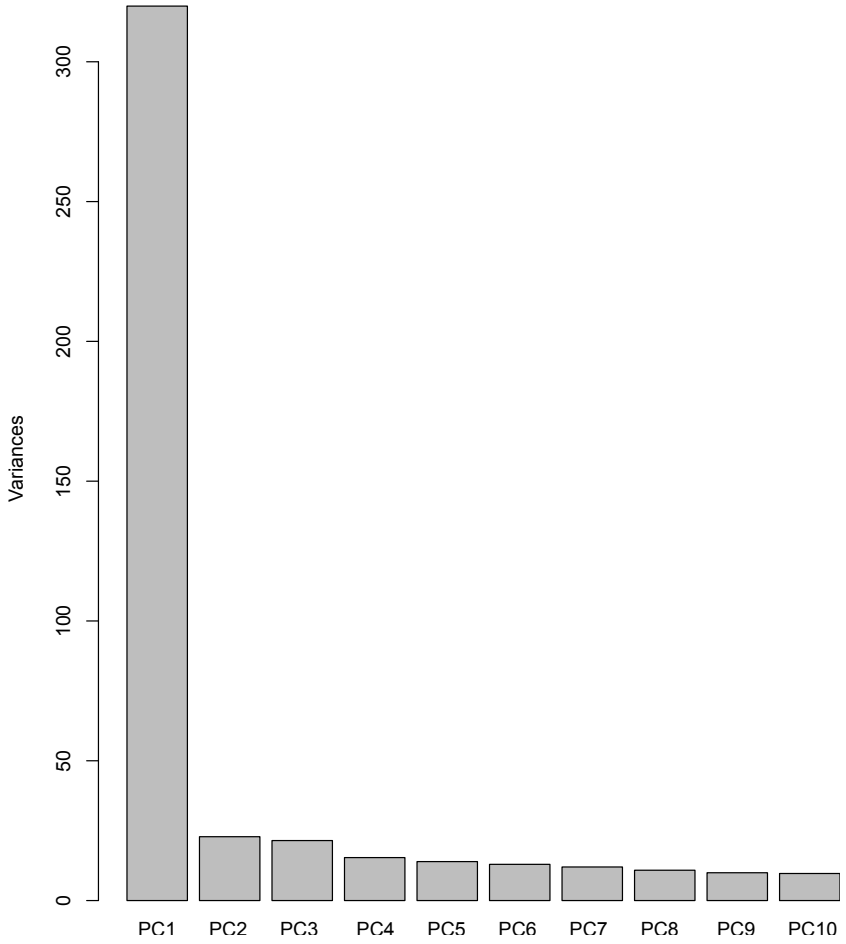


FIGURE 4.11: Principal Component Analysis based on the duration of calls

Another recurring pattern over all the evaluations is the drop of overall communication flows happening at the end of the evaluation period. This can be attributed to political unrest with a neighbouring country[6].

The next step consisted in applying our weighted PageRank algorithm over the data set. For this purpose we proceeded in a similar fashion as for the PageRank calculation. We first processed the dataset in such a manner that we obtained records that provided us the weight values for each inbound and outbound connection. These weights were either the total number of calls on a given edge or respectively the total duration of these calls on a given edge. Figures 4.15 and 4.16 shows the results we obtained for the total number of calls again in relation to the time slots and base stations whereas Figures 4.17 and 4.18 represent the results taking the duration into account.

From previous experiment performed some abnormal base station are already known. The map in Figure 4.7 showed these abnormal base stations. We now compare the PageRank score distribution of some of these base stations with other randomly chosen

[6] http://www.breakingnews.com/topic/mali-unrest

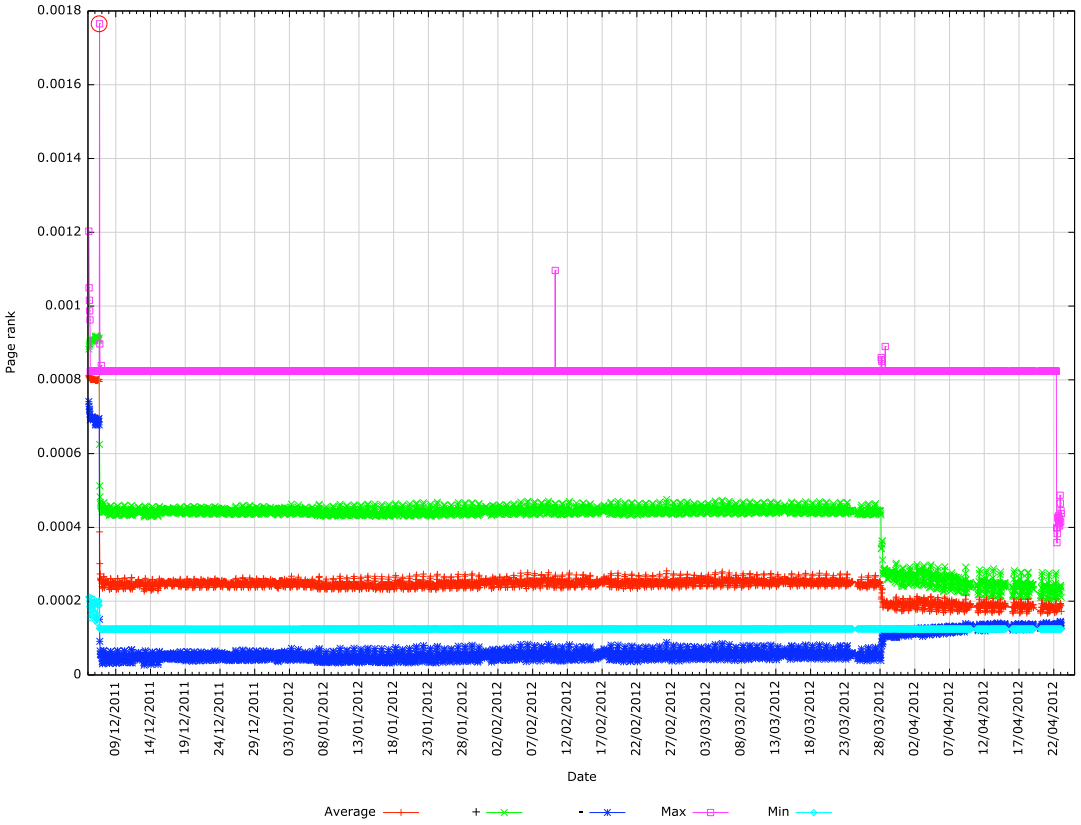


FIGURE 4.13: PageRank score over all base stations per time slot

base stations during a distinct period. For this purpose we plot the probability density functions of four candidate base station as illustrated in Figure 4.19

This lead us to the principle component analysis that provides us with transformed set of variables that have most influences on the base stations using the PageRank score values. The first component has the highest influence on all the base stations over the period that we previously used for the probability density functions. This component influences 85 % of the base stations during this period. Our goal was to analyse the different influential coefficient of this component and discover some of our base stations that were discovered in the previous experiment. Figure 4.20 shows the result of this analysis

4.5 Future work and conclusions

To summarize, we have presented a model based on Call Detail Records which describes the communication flows of a given network. We applied a metric on a country-level data set that allows us to distinguish normal communication flows form abnormal ones, enabling us further investigate these patterns after applying several data transformation.

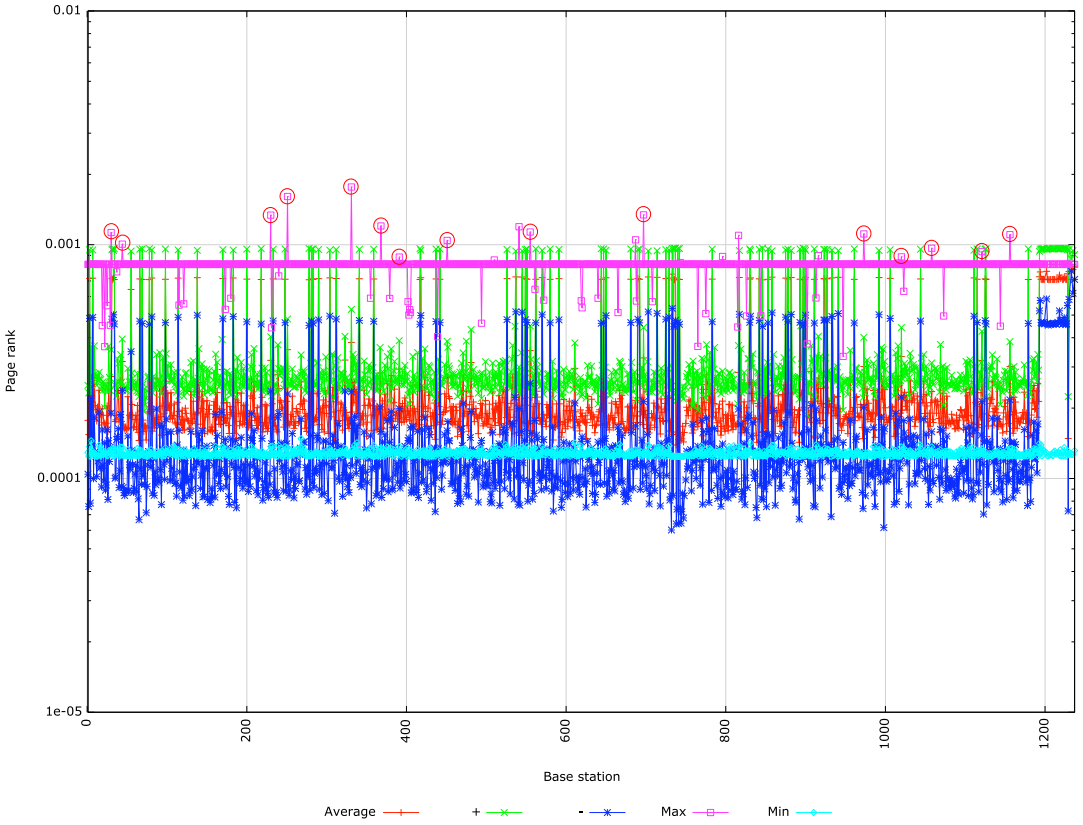


FIGURE 4.14: PageRank score over the evaluation period for each base stations

We successfully correlated several abnormal communication patterns to real events happening at the same time and determined which components have the highest impact on the overall network.

We also used PageRank algorithm to analyse the activity on each base stations and determine the global impact on the communication infrastructure. This allowed to view the network in a whole. We applied our model on the same country-level data set and were able to detect certain interesting results and we were able to correlate these with possible real events happening at the same time.

A topic not adequately covered but nevertheless important is the granularity of the data. The window size represents a common diurnal working behaviour. Increasing or decreasing this window has a huge impact on the granularity of the result, as large windows will produce a smaller result set which can be quickly processed but is not as detailed as is the case for smaller window. The latter results in a detailed set close to the original raw dataset, but entails a higher processing time.

The real time execution and evaluation is mentioned in this chapter but not thoroughly discussed. We are going to extend our method to evaluate the feasibility of our method by using online streaming data and platform to evaluate this such as Storm [110, 111].

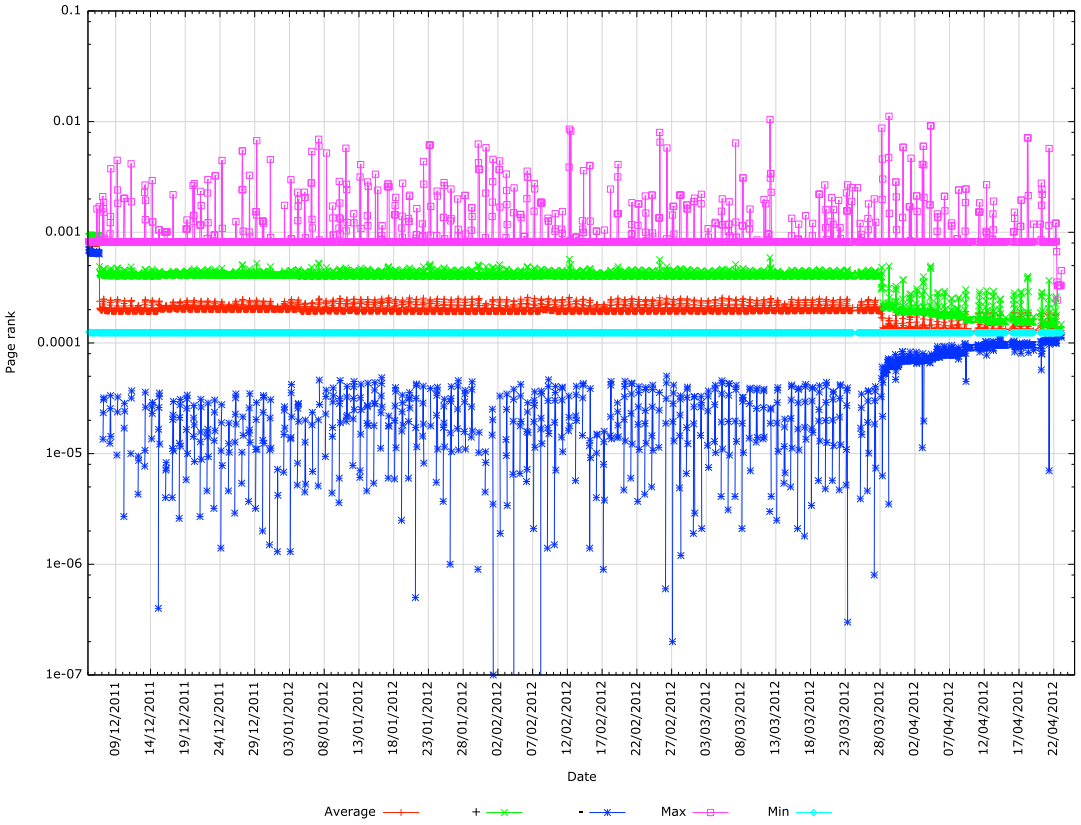


FIGURE 4.15: Weighted PageRank score using total number of calls over all base stations per time slot

Furthermore we will investigate distributed data mining approaches to further enhance our method and combine these with Storm. An interesting topic is also the mobility of the users. In a further experiment will we evaluate the impact of the users mobility on the network.

4.6 Acknowledgements

We would like to express our gratitude to the D4D Challenge committee and France Telecom / Orange Côte d'Ivoire for allowing us access to their collected data sets within the framework of the D4D Challenge.

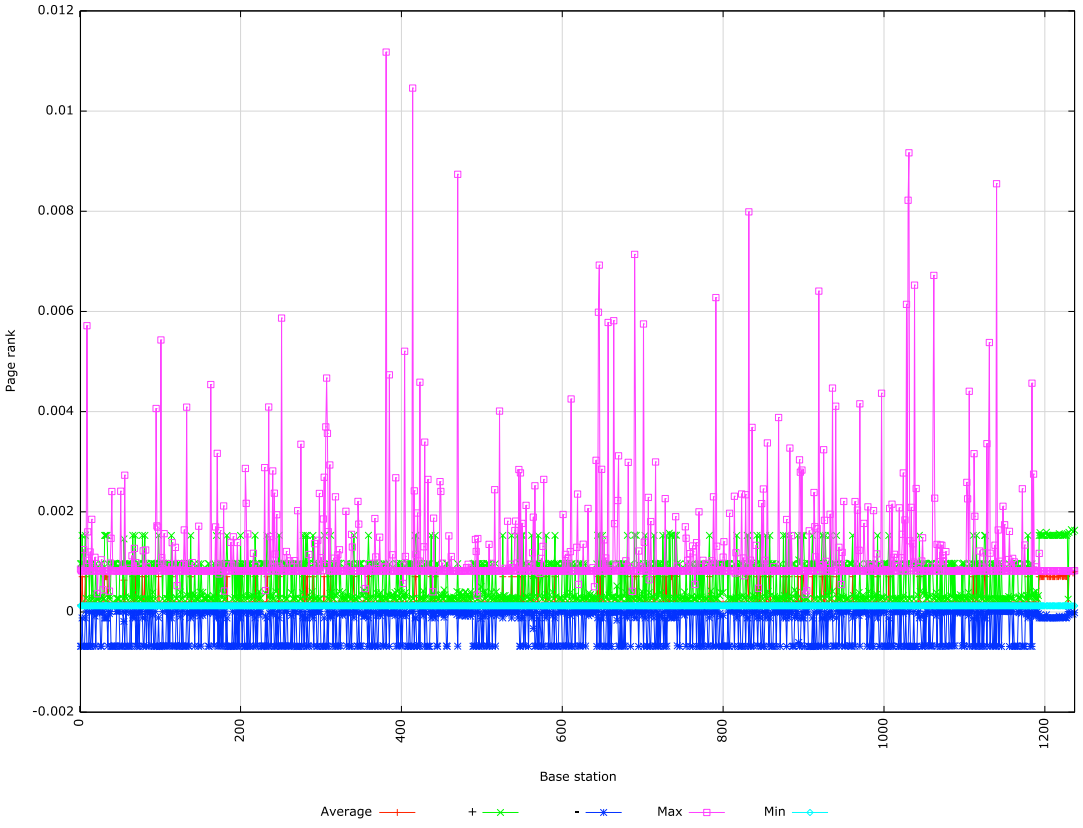


FIGURE 4.16: Weighted PageRank score using total number of calls over the evaluation period for each base stations

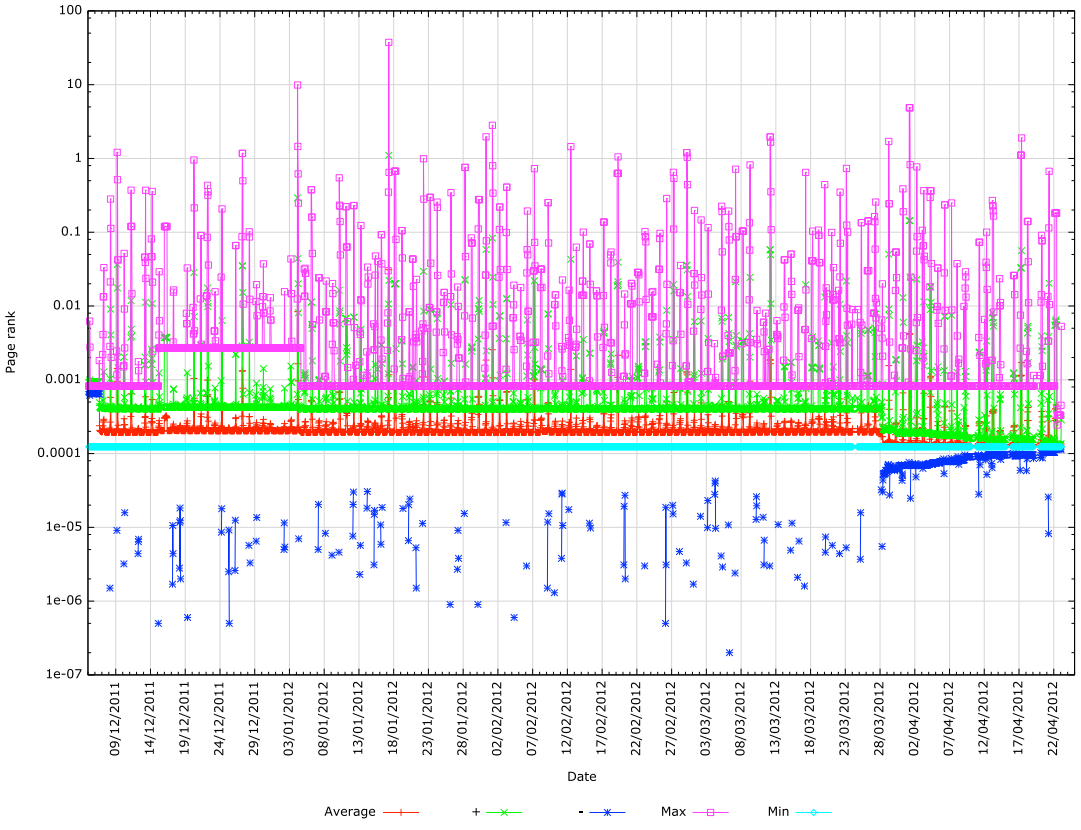


FIGURE 4.17: Weighted PageRank score considering the total duration of calls over all base stations per time slot

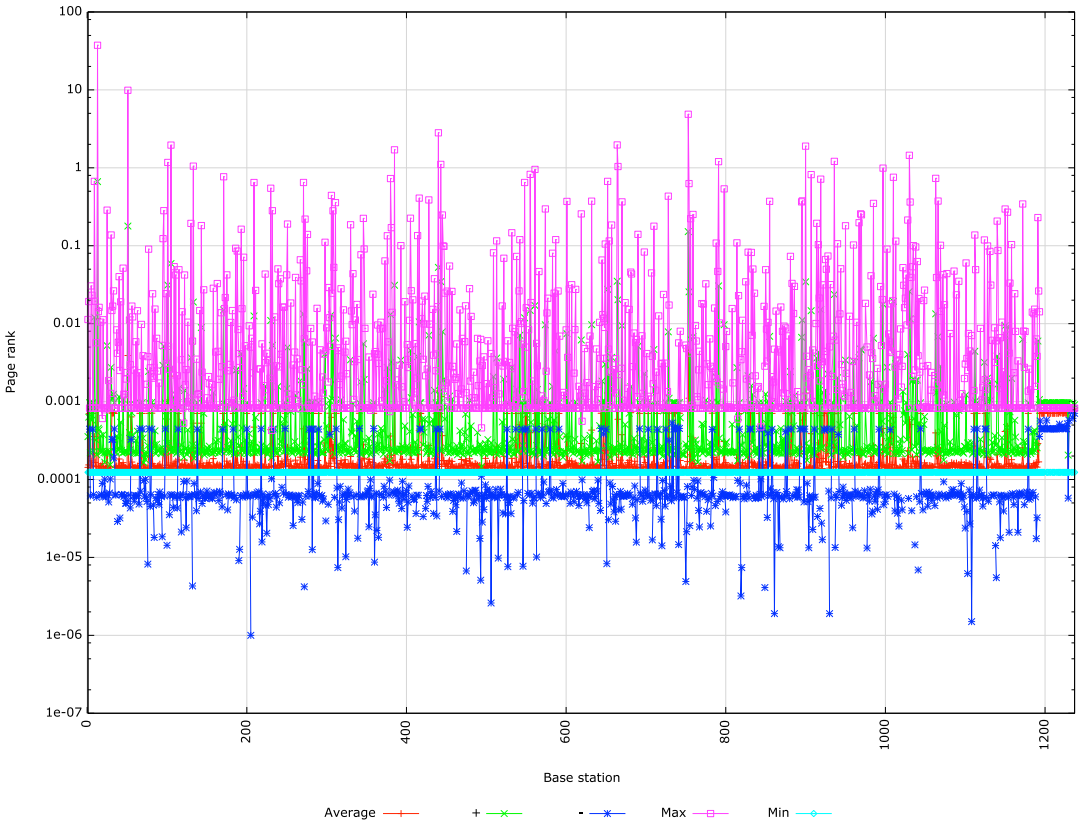


FIGURE 4.18: Weighted PageRank score considering the total duration of calls over the evaluation period for each base stations

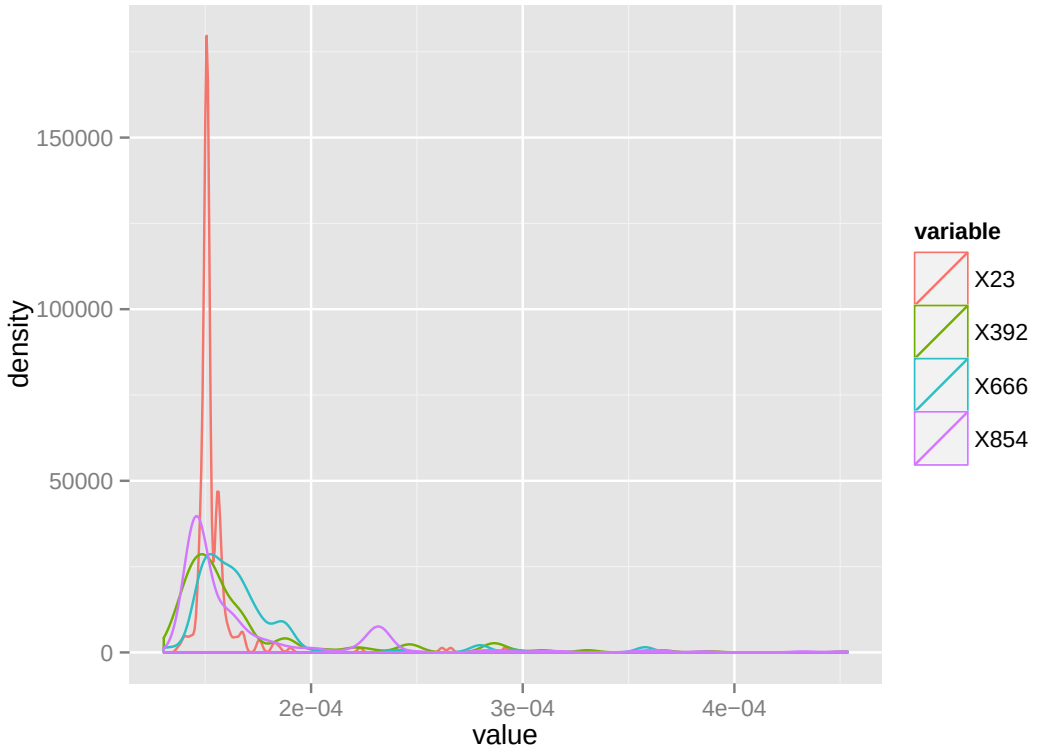


FIGURE 4.19: Probability density plot of abnormal and normal behaving base station

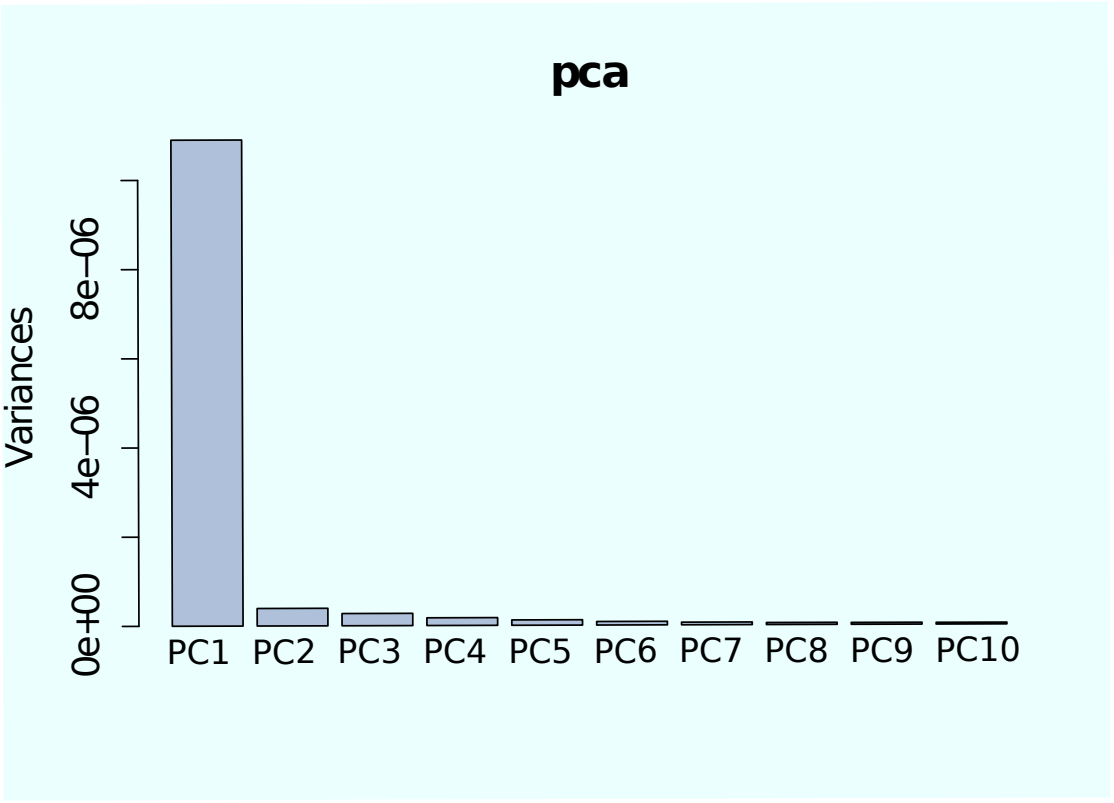


FIGURE 4.20: Principle Component Analysis using calculated PageRank scores

Chapter 5

Measuring Broadband Network Capabilities

The Application Traffic Layer Optimization (ALTO) protocol allows network service providers (NSP) to make available a pair of maps to applications such that the applications can intelligently (compared to randomly) connect to a desired resource. The network map aggregates the service provider network into provider defined identifiers (PID) and the cost map provides a pair-wise link cost between each PID. Clearly, a NSP has an authoritative view of its network and is able to provide an ALTO server that distributes such maps. However, ALTO also envisions third-parties as being able to provide such maps. In this paper, we demonstrate how a third-party ALTO server can provide maps by mining public information. Specifically, we build our maps from the United States Federal Communications Commission public broadband data set, which contains a rich (multi-tier wireline broadband measurements) and extensive (measurements for specific application uses) data set. We initially view the network as consisting of disconnected components that we organize by Internet service providers (ISP); each component corresponds to a PID. This essentially constitutes the network map. We then analyse multiple attributes associated with hosts in the component to determine a measure of homophily that can be leveraged to connect different components. Based on the strength of the homophily, we assign costs to the edges we create between previously disconnected components, thereby creating the cost maps. In all, we examined over 1 billion records spread over 90 GBytes as part of our analysis.

5.1 Introduction and problem statement

The Application Layer Traffic Optimization Protocol [38] is a (soon-to-be) IETF-standardized protocol [39] that constitutes a service for disseminating network information (e.g., basic network location structure and preference of network paths) to applications to improve performance. The basic information in ALTO is provided in the form of two maps: a network topology map and a cost map. These maps allows the network provider to provide a succinct, yet abstract view, of the network to allow applications express preferences when connecting to the desired resource. For example, a node in a peer-to-peer network can choose other peers that belong to the same Internet service provider (ISP) as the querying peer to minimize latency; similarly, a Content Distribution Network (CDN) router can use the maps to rendezvous a client with the nearest surrogate. Figure 5.1 shows an example of the map service. In the example, the network map partitions all endpoints into three provider-defined identifiers (PID); the *Default* PID is essentially a catch-all for all hosts not belonging to the ISP. The cost map provides directed costs between each pair of PIDs. Using these maps, a peer in PID-2 will contact peers in PID-1 first before reaching out to other peers on the Internet, fore-armed with the knowledge that it costs less (2 cost units) to access peers in PID-1 than it does to access peers in the default

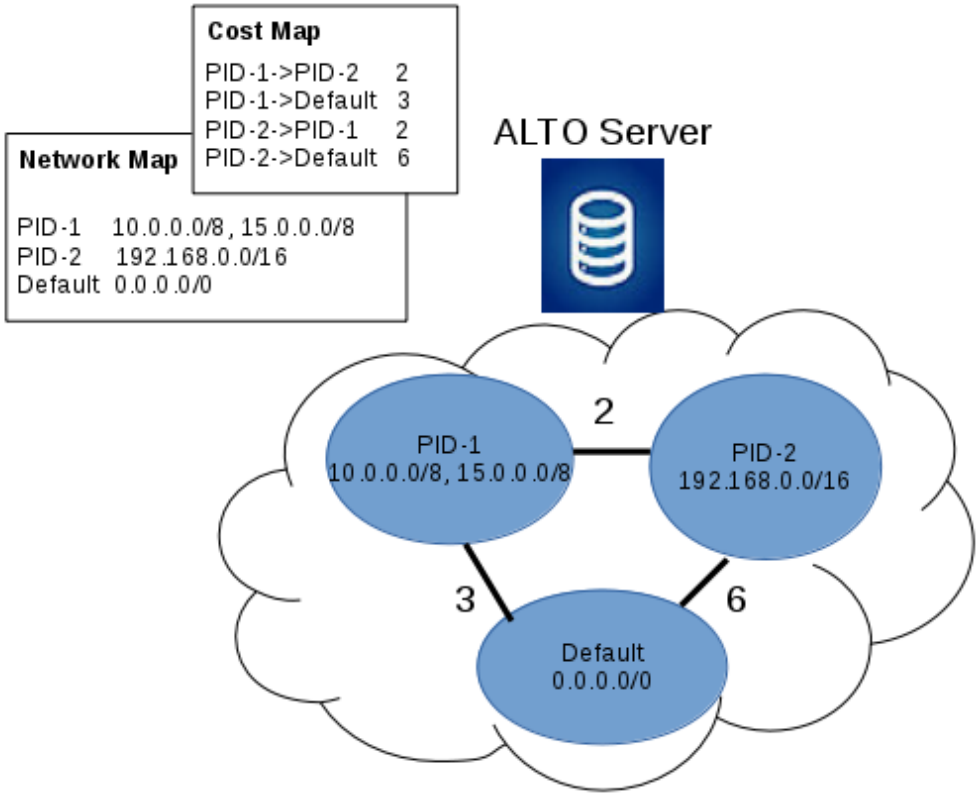


FIGURE 5.1: An ALTO server

PID (6 cost units). The costs on the link represents generic costs and are computed by

the ISP according to its preferences and policies. These two abstractions — the topology map and the cost map — provided by the ISP allow peers (or clients) on the network to affect the choice of peers (or servers) they will connect with. Published literature indicates active research interest in ALTO for peer-to-peer (P2P) applications [112, 113], content distribution networks (CDN) [114], and virtual private networks (VPN) [115].

In principle, the ALTO protocol allows for parties not associated with the ISP (third parties) to create topology and cost maps and distribute these through an ALTO server. However, in ALTO-like deployments prevalent today, it is the ISP that creates these maps and make them available on an ALTO server. After all, the ISP is the authoritative entity that knows the dynamics and reach of its network as well as the settlement pricing associated with peering and transit links. To the best of our knowledge, there isn't any published literature that allows third parties to create network topology and cost maps tailored to a specific application running in an ISP network.

Our primary contribution is to demonstrate how third parties can create network and topology maps for ALTO from public sources of information. More specifically, we use the United States Federal Communications Commission (FCC) public database from the Measuring Broadband America (MBA) program. This is an ongoing, rigorous, nation-wide study of residential broadband performance in the United States. We mine the public data to assign each subscriber to an ISP (this relationship is not captured in the data distributed by the FCC). The ISP clusters serve as the PIDs in our work. In fact, unlike an ISP that possesses an ISP-wide view, our work operates at a layer above, essentially considering relationships between a cluster of ISPs. Once the PIDs are identified, we use social network analysis techniques to determine a measure of homophily of the nodes that will be used to connected the disconnected components. Based on the strength of the homophily, we assign costs to the edges created between the network components; these costs essentially consists of a cost map. We base our results from the July 2012 data [31]. In all, we examined over a billion records spread out in 90 GBytes of files as part of our analysis.

5.2 Describing the data: FCC Measuring Broadband America and data set characteristics

To model or reason on network dynamics, one needs to have measures derived from the actual installation over a reasonable period of time. Obtaining such data is not easy because the network operators typically do not make such data public due to privacy and business reasons. Furthermore distributing sensors evenly to measure performance

and capabilities of a system need careful planning as these should not interfere with the daily traffic of the system biasing the measures. In an ongoing study that started in 2010, the U.S. Federal Communication Commission (FCC) has been measuring the nationwide performance of broadband service in the United States under a program called Measuring Broadband America (MBA). The data collected is available publicly after anonymising all subscriber identifying information. The subscriber is represented by an invariant opaque unit identification number, a *unit_id*. All other identification information — name, street address, IP addresses, customer tier — are removed from the data before making it publicly available. The measurement methodology of MBA has been developed in collaboration with SamKnows, an international broadband measurement firm.

The MBA program collects data with the support of national broadband service providers to measure the actual capabilities of their national wired network. MBA is an opt-in program where volunteers agree to host a SamKnows whitebox in their homes. The whitebox plugs into the home network (Figure 5.2[1]) and runs a series of network performance tests hourly to gather the data. The data is then collected, analysed and published as an annual report by the FCC. More information about the MBA program is available in the FCC MBA Technical Appendix [31].

There are 14 tests run once every hour. The whitebox senses if the subscriber's network is in use, and if so it delays the test to avoid interference with non-test traffic. Data from each of the tests is sent to SamKnows servers where it is tabulated, anonymised and then made available for public consumption. For our work, we used the data corpus from 2012 [2]. The data set consists of 12 tables, each table corresponding to the data drawn from a certain performance test. For our work, we use three tables — curr_dns, curr_netusage, and curr_videostream (see Figure 5.3):

1. *curr_dns* uses 9 predefined and well-known websites to measure the DNS resolution time for each of them. The data tabulated in this table includes the nameserver used for lookup, the name of the target host, the IP address of the target, among other fields. (In Table 5.1 below, data associated with *curr_dns* was the largest in terms of both file size and records analysed)

2. *curr_netusage* captures the total amount of data received and transmitted form the subscriber as well as the amount of bytes transmitted and received as a result of active performance measurements (the *sk_{rx,tx}_bytes* in Figure 5.3).

3. *curr_videostream* simulates the subscriber viewing an online streaming video. A three-second playout buffer was configured in the client, and the client attempted

[1]Figure source: https://www.samknows.com/broadband/how-it-works

[2]The anonymised raw data for each month of 2012 is available at http://www.fcc.gov/measuring-broadband-america/2012/raw-data-2012/

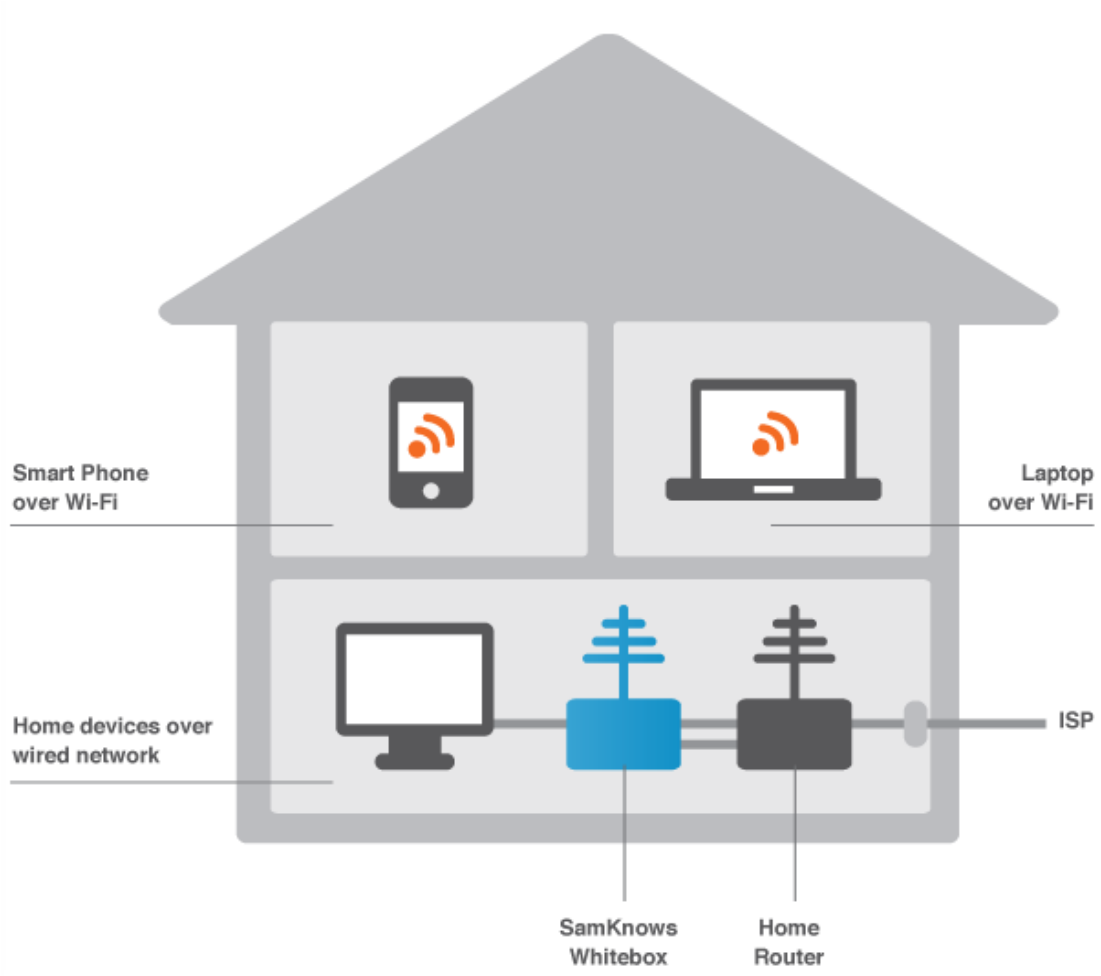


FIGURE 5.2: Deployment architecture for collecting measurements

to download data at the maximum rate necessary to ensure that this buffer never drained. The test captures features of the buffer (buffer size, the number of buffer underruns), as well as the connection itself i.e. download throughput, jitter and latency.

Each of the tables contain *unit_id* as an invariant, allowing us to cross-index the same subscriber across tables for the whole year. Note that each table has a *location_id* field as well, but the contents of this field are opaque (random digits) and cannot be used to derive the physical location of the subscribers.

Table 5.1 shows the total volume of data we analysed from the 2012 dataset.

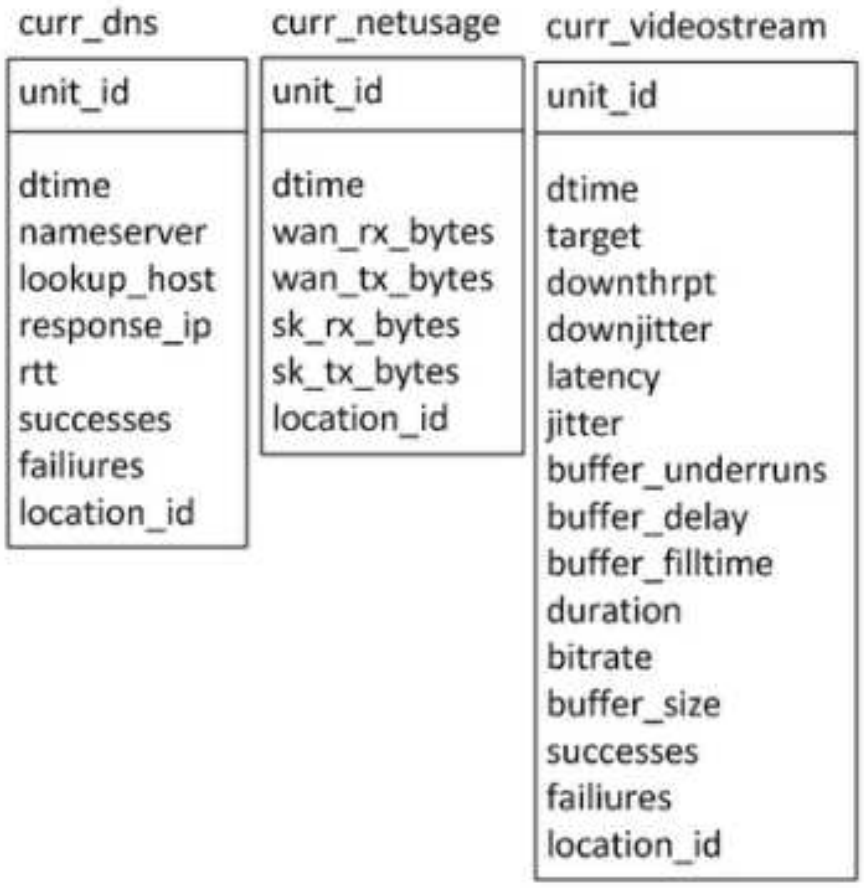


FIGURE 5.3: FCC MBA tables used in our work

	Data analysed (GBytes)	Records analysed (Millions)
curr_dns	75.9	984.3
curr_netusage	2.9	48
curr_videostream	11.8	96
Total	90.6	1128.3

TABLE 5.1: Total volume of data analysed for 2012

5.3 Stripping some anonymity: geo-locating unit_ids to create a network map

Given the amount of data and records to be analysed, the first challenge, was to arrive at computing resources comparable in scale with respect to the dataset consisting of over 1 billion records spread across gigabyte-sized files. To analyse the volume of data we used a canonical Map-Reduce computational paradigm [108, 109] on a Hadoop cluster using Apache Hadoop open-source implementation. The Hadoop cluster is composed of 4 nodes, each running hexacore Xeon CPU with 2.4 GHz and a total memory of 120

GB. The files are stored on a distributed files systems having a total of 27.54 TB of free space. We use Hadoop version 2.0.0-cdh 4.30 for our computations. The advantage of this paradigm is to reduce the huge computational task into several smaller tasks that perform the same operation as the larger task but more efficiently. The time needed for extracting DNS resolver's IP addresses associated to individual *unit_ id*s from the raw data was around 30 min. for the whole data set. To identify the geographic location of the *unit_ ids* (see next paragraph), we used a geo-location service, although we were limited by the number of queries. It took us two days to analyse the entire dataset.

A second, more pressing challenge, was to identify the geographic location of the *unit_ id*s generating the data. In order to derive a topological map and impose costs on the links, it is important to know the physical locations of the *unit_ id*s that contributed the measurements. However, in the MBA dataset, the population is anonymised and the individual subscriber reporting the measurement data is simply referred to by an opaque integral number. Therefore, an important task was to use the information in the public dataset to reveal a coarse location of the subscriber and assign that subscriber to a specific ISP for further analysis.

To geo-locate the units, we simply note that broadband subscriber devices are likely to be configured using DHCP by their ISP. Besides imparting an IP address to the subscriber device, DHCP also populates the DNS name servers the subscriber devices uses for DNS queries. In most installations, these DNS name servers are located in close physical proximity of the subscriber device. The FCC technical appendix states that the DNS resolution tests were targeted directly at the ISP's recursive resolvers to circumvent caching and users configuring the subscriber device to circumvent the ISP's DNS resolvers. Therefore, a reasonable approximation of a subscribers geo-location could be the geographic location of the DNS name server serving the subscriber. We use this very heuristic to geo-locate a subscriber. We stress that this methodology does not identify the specific location of a subscriber, who still remains anonymous. Instead, it simply locates the subscriber in a larger metropolitan region and associates an ISP with the specific subscriber. This level of granularity suffices for our work.

Our first, and very simple filter consisted of obtaining a mapping from a *unit_ id* (representing a subscriber) to one or more DNS name servers that the *unit_ id* is sending DNS requests to. It turned out that while this was a necessary condition for advancing, it was not a sufficient one. The raw data would need to be further processed to reduce inconsistencies and remove outliers. A number of interesting artefacts were uncovered during further processing of the data. These artefacts informed the selection of the *unit_ id*s for further analysis.

The artefacts that lead to outliers are documented next.

1. A handful of *unit_ id*s were geo-located in areas outside the contiguous United States, such as Ukraine, Poland or the United Kingdom. We theorize that the subscribers corresponding to the *unit_ id*s geo-located outside the contiguous United States had simply configured their devices to use alternate DNS servers, probably located outside the United States. We removed these records before conducting our analysis.

2. We also observed a reasonable number of non-ISP DNS resolvers, especially Google's 8.8.8.8 and 8.8.4.4 and OpenDNS 208.67.222.222 and 208.67.220.220. These 4 public DNS servers are geo-located in California. We removed these records to ensure that the specific location that these resolvers represented was not oversampled.

3. We noticed that a large number of *unit_ id*s were being geo-located in Potwin, Kansas ($37°N97°W$). Intrigued as to why there appeared to be a large population of Internet users being located in a small rural community in Kansas (population 441), we investigated further. It appears that Potwin, Kansas is the geographical centre of the United States. If the IP geo-location service is unable to pin point the location of an IP address, it returns the coordinates corresponding to the geographic centre of the country. This accounted for the popularity of Potwin, Kansas as an Internet destination. Other researchers have dubbed this the "Potwin Effect" [116]. We excised all records that showed the impact of such natural aggregation points that, if not accounted for, will skew our results in an unwarranted direction.

Subsequent filters extracted the stable *unit_ id*s from our dataset. In order to determine which *unit_ id*s are stable, i.e., remain constant with respect to their geographic location over the observation period from January to December 2012, we extracted for each *unit_ id* the IP address of each DNS name server it consulted. This was repeated for each month of the observation period. The resulting sets were cleaned up of private IP addresses and other artefacts discussed above. The cleaned set consisted of about 8000 distinct *unit_ id*s.

In order to determine the stability of each *unit_ id* we proceeded to sum up the occurrences of IP addresses over the whole observation period separated in monthly files. If the IP address of a DNS server occurred 12 times this meant that the *unit_ id* always accessed the same DNS server and therefore remained stable over the observation period. The obtained stable *unit_ id*s, 2,089 of them, were used for further analysis. We note that about 2,000 *unit_ id*s constitute a good sample from a population of 7,968 subscribers in the July 2012 MBA report [31]. Assuming a 99% confidence level and ± 3 point margin of error, we will require a sample of 1494 *unit_ id*s. With our more than adequate stable set of 2,089 *unit_ id*s , we were poised to perform further analysis on the

dataset to create the full topology and cost maps. The stable *unit_ids* superimposed on the map of the United States are illustrated in Figure 5.4. The radius of the circle represents the concentration of *unit_ids* in the given geographical location.

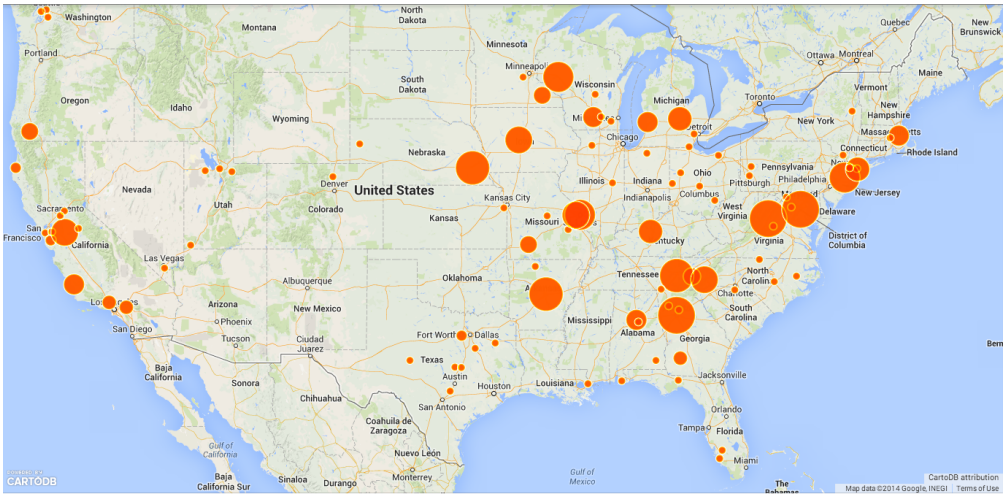


FIGURE 5.4: Distribution of stable measurement points (unit_ids)

A side effect of geo-locating subscribers using the DNS resolver is that we were also able to associate each of the stable *unit_ids* with an ISP. Because the DNS resolution test was targeted directly against the ISP's recursive resolvers, it was trivial, therefore, to figure out which ISP served a particular *unit_id*. Table 5.2 shows those ISPs having at least 19 stable *unit_ids*. The remaining 53 *unit_ids* are distributed over 32 ISPs, but because each ISP has a small number of *unit_ids*, they are not listed in the table. For the analysis in the rest of this paper, we only consider the top 9 national ISPs listed in Table 5.2.

ISP	Number of *unit_ids*
Comcast	726
Charter Communications	447
Cox Communications	386
Verizon DSL (MCI)	243
Windstream Communications	151
Mediacom Communications	49
Cablevision	37
Time Warner	31
Century Link (Embarq)	19
Total	**2,089**

TABLE 5.2: *unit_id* to ISP distribution

The collection of ISPs can serve as our network map. In ALTO, a network map is provides a full set of network location groupings and endpoints contained within each grouping. The ISPs serve as our location grouping, with each ISP corresponding to a PID. The

stable hosts in an ISP serve as endpoints contained within each grouping. Figure 5.5 shows the top 9 ISPs as isolated clusters (or disconnected components) in a network. (Only 30% of the hosts in each ISP are shown in the figure to aid in visualization of the graph with minimum clutter and overlap.) Note that at this point, no links exists between the components; i.e., we have only created a network map, a cost map that connects the components still needs to be created.

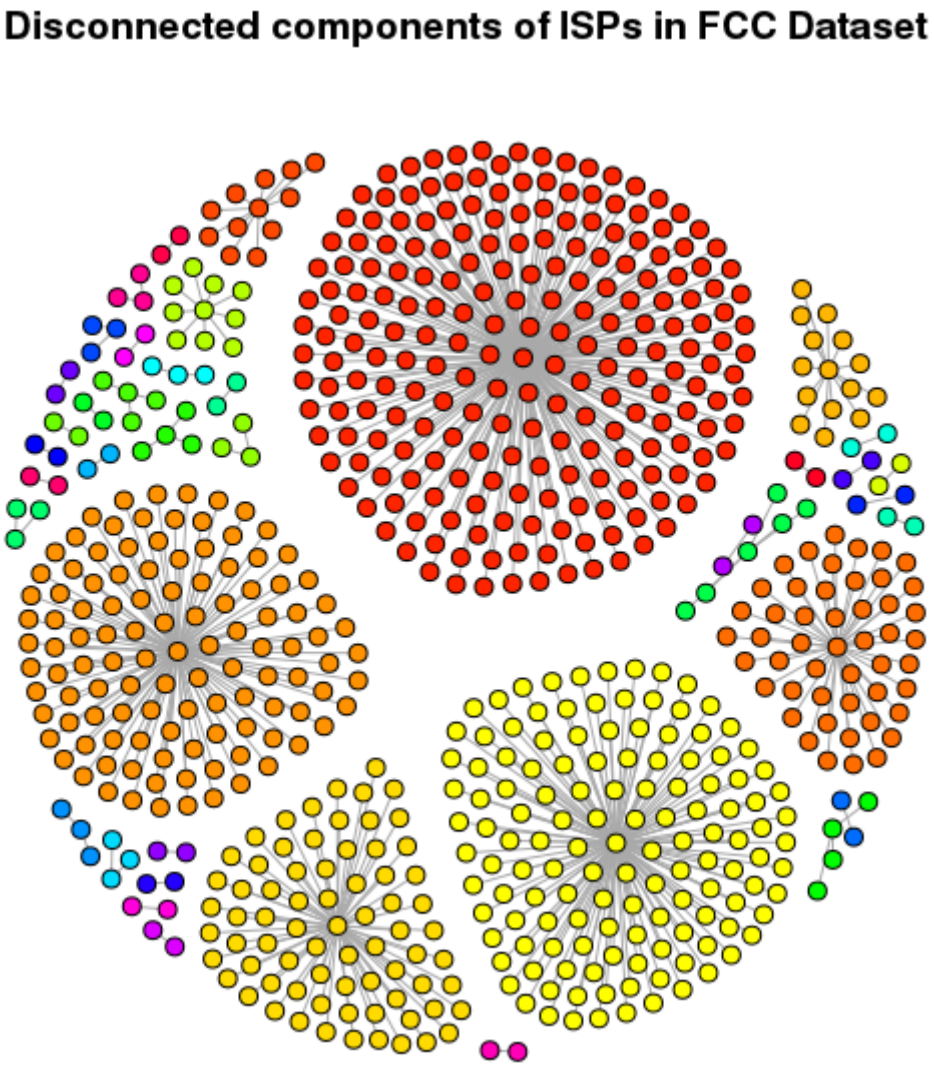


FIGURE 5.5: ISPs in a network map depicted as disconnected components

5.4 Building cost maps from FCC data

To build the cost maps, we use the *curr_videostream* table from 5.3. We build cost maps for each ISP using two important features, which are relevant for end user experience. These are the download throughput rate and latency. The datasets contain these measures for each *unit_id* over a 12 months period, measured on a hourly basis. We

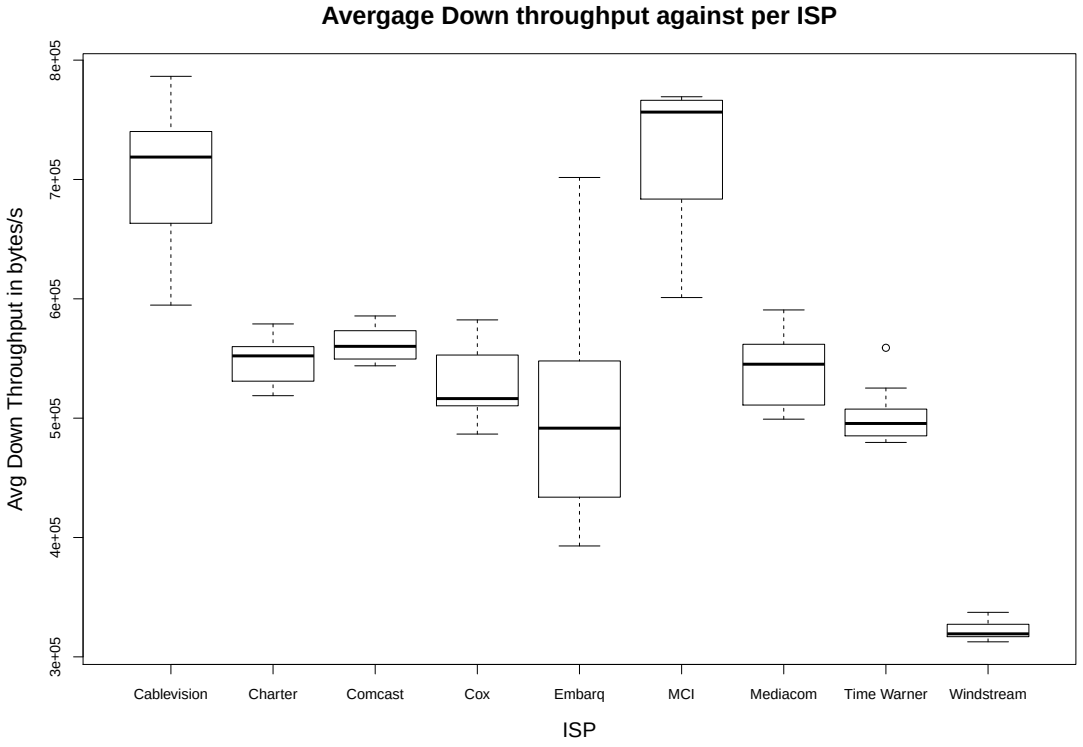


FIGURE 5.6: Box and whisker comparing down throughput plot for major US-based ISPs

compute a monthly average of these rates for each ISP and define the cost matrix in terms of Sharpe ratio [117], a common metric used in financial engineering. This ratio measures to which extent a given asset is a adequate trade off between the risk undertaken by an investor compared to the expected return. The Sharpe ratio is a straightforward and natural extension for calculating a cost matrix for our topology: We model the investor as a node in the network and a portfolio is modelled as the specific service of interest — seeking PIDs where nodes have a high download throughput or PIDs where nodes have low latency.

The Sharpe ratio for a portfolio p, is computed by subtracting the risk-free rate of return (R_f) from the rate of the portfolio return itself ($\bar{r_p}$), and dividing by the standard deviation of the portfolio returns (σ_p), as shown in (5.1):

$$S_p = \frac{\bar{r_p} - R_f}{\sigma_p} \tag{5.1}$$

Historical averages alone might not be appropriate if the associated standard deviations are high, because in such cases the effective metric of interest is much lower than the historical average. However, the Sharpe ratio takes the associated standard deviation in account, so it is a better candidate for approximating a cost function. Higher Sharpe

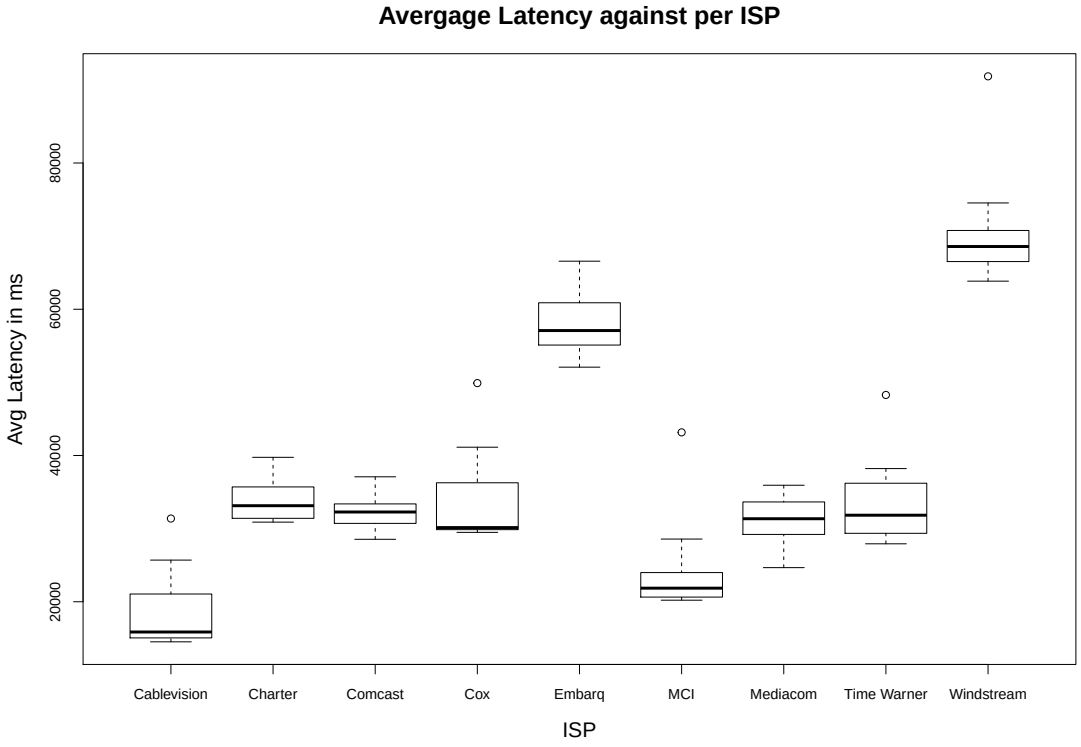


FIGURE 5.7: Box and whisker comparing latency plot for major US-based ISPs

ratios are equivalent to high averages and small standard deviations. Smaller values for the Sharpe ratios are due to either high standard deviations, or to small averages. A negative Sharpe ratio implies that the risk-free rate of return would perform better than the portfolio being analysed

We create two cost maps, each cost map is specific to a class of applications. Cost map, C_d is created for applications that seek peers with high download bandwidth, and the second cost map, C_l is used by applications that want to minimize latency during communications. The cost maps C_d and C_l are computed by analysing the Sharpe ratio using the Equation (5.1) where $\bar{r_p}$ represents the actual average download throughput (or average latency) for the given month, R_f the global yearly average download throughput (or yearly average latency) and σ_p the standard deviation of the download throughput (or latency). R_f represents the risk-free rate of return. Figure 5.9 shows the Sharpe ratios for the download bandwidth (Figure 5.9a) and latency (Figure 5.9b). The figure shows the the monthly evolution (x-axis) of the Sharpe ratio (y-axis) for our 9 ISPs. The horizontal line in the graphs (at $y = 0$) represents the thresholds where risk-free investment is the better option. These ratios can be

compared in terms of absolute values, but also on the temporal evolution for individual ISPs. There are some interesting trends to explore here, which we discuss below before showing how to create the cost maps.

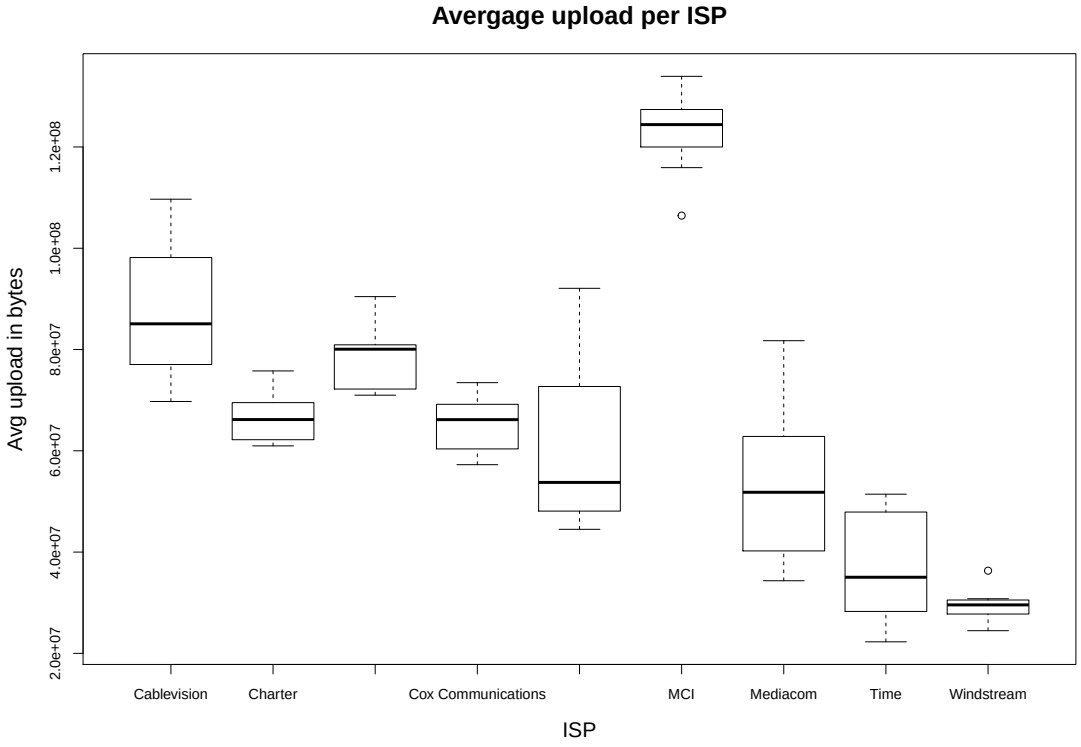
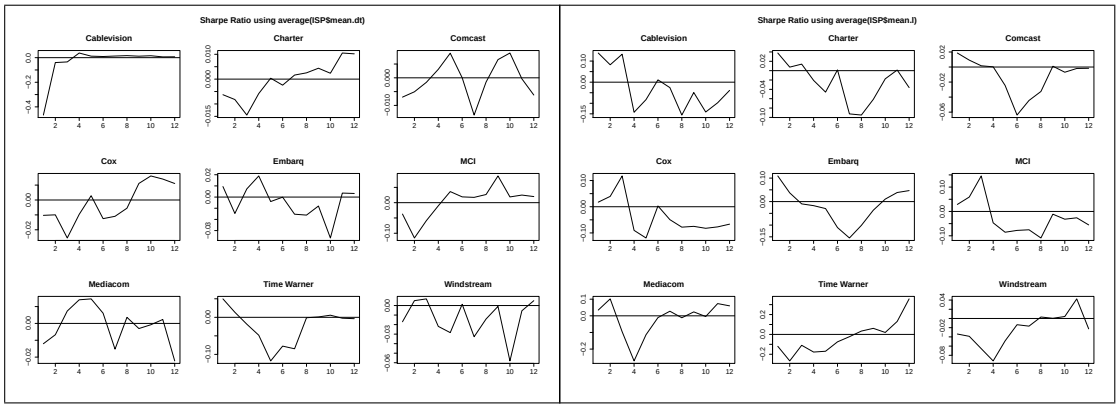


FIGURE 5.8: Box and whisker comparing upload plot for major US-based ISPs



(A) Download Throughput (B) Latency

FIGURE 5.9: Sharpe ratios for download throughput and latency

From Figure 5.9a is clear that Time Warner achieves high Sharpe ratios over significant periods (month 2 - month 12), with values ≥ 0.0, which means that most monthly averages were higher then the yearly average. Conversely, the Sharpe ratio is ≤ 0.0 for January and February and increase strongly afterwards. It appears that the reason for the stability in bandwidth after February 2012 is the merger of Time Warner and Insight Communications, which served the US states of Kentucky, Indiana and Ohio. This

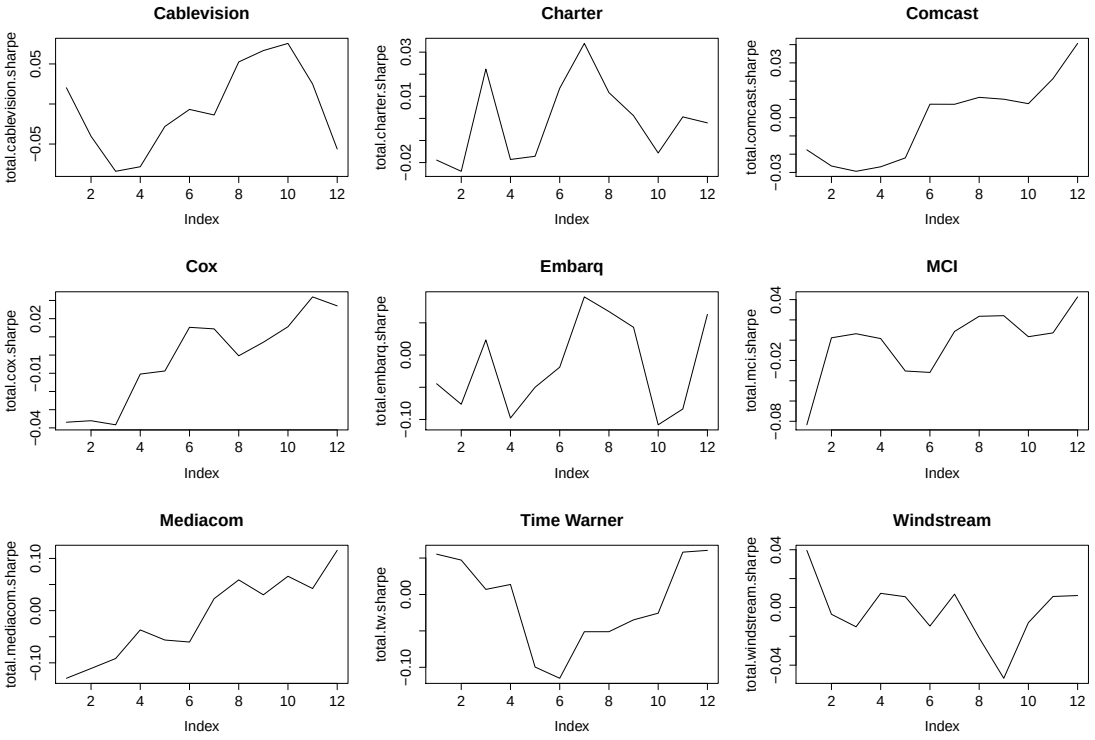


FIGURE 5.10: Sharpe ratio for major US based ISP

merger was completed on February 29, 2012 [3] Insight Communications is the ninth-largest cable operator in the US, and therefore, an increase in the number of subscribers is reflected as an increase in the download throughput. Cox Communications shows a steady increase in its performance: over one year, the Sharpe ratio doubles with a monthly permanent increase. This may be a result of cumulative engineering and provisioning (we did not research the cause for this). Cablevision shows an interesting pattern: the Sharpe ratio is overall constant, but a high increase between the third and sixth month is observed. There are other clear winners: Charter, Mediacom and Windstream appear to have Sharpe ratios ≥ 0 for most of the year, although Windstream and Embarq both show sizeable dips in the ratio between August and October.

The cost map, C_d is calculated from the Sharpe ratio using Equations 5.2 and 5.3.

$$\forall i \in ISP, C_d^i = \sum_{m=1}^{12} |S_m^i| * W_i \tag{5.2}$$

Here, C_d^i is the download throughput cost for each ISP i, which is calculated over summation of the absolute value of each month's Sharpe ratio for the ISP i (S_m^i) and multiplying

[3] http://www.kentucky.com/2012/02/29/2089006/time-warner-cable-takes-over-insight.html, site visited April 25, 2014.

by a weight associated with the ISP. We give preferences to ISPs whose Sharpe ratio is above the risk-free investment line, thus, W_i is the number of months that the ISP i stayed above the risk-free investment in the last 12 months.

In ALTO, a lower value for a cost indicates a huger preference for traffic to be sent from a source to a destination. Therefore, the actual cost for each ISP is further calculated as:

$$\begin{aligned} \forall i \in ISP, Max\_cost = \lceil C_d^i \rceil \\ \forall i \in ISP, C_d^i = Max\_cost - C_d^i \end{aligned} \tag{5.3}$$

That is, we subtract the maximum cost over all ISPs from the cost calculated for each individual ISP. The latency cost for each ISP, C_l^i, is calculated in a similar manner. The computations of Equations 5.2 and 5.3 result in the cost matrix shown in Table 5.3 for C_d and C_l.

	Cablevision		Charter		Comcast		Cox		Embarq		MCI		Mediacom		TimeWarner		Windstream		Default	
	C_d	C_l	C_d	C_l	C_d	C_l	C_d	C_l	C_d	C_l	C_d	C_l	C_d	C_l	C_d	C_l	C_d	C_l	C_d	C_l
Cablevision	0.00	0.00	6.52	5.84	6.69	6.97	6.30	4.74	6.31	4.49	3.23	5.74	6.35	3.03	5.30	0.28	6.21	6.47	∞	∞
Charter	0.97	3.61	0.00	0.00	6.69	6.97	6.30	4.74	6.31	4.49	3.23	5.74	6.35	3.03	5.30	0.28	6.21	6.47	∞	∞
Comcast	0.97	3.61	6.52	5.84	0.00	0.00	6.30	4.74	6.31	4.49	3.23	5.74	6.35	3.03	5.30	0.28	6.21	6.47	∞	∞
Cox	0.97	3.61	6.52	5.84	6.69	6.97	0.00	0.00	6.31	4.49	3.23	5.74	6.35	3.03	5.30	0.28	6.21	6.47	∞	∞
Embarq	0.97	3.61	6.52	5.84	6.69	6.97	6.30	4.74	0.00	0.00	3.23	5.74	6.35	3.03	5.30	0.28	6.21	6.47	∞	∞
MCI	0.97	3.61	6.52	5.84	6.69	6.97	6.30	4.74	6.31	4.49	0.00	0.00	6.35	3.03	5.30	0.28	6.21	6.47	∞	∞
Mediacom	0.97	3.61	6.52	5.84	6.69	6.97	6.30	4.74	6.31	4.49	3.22	5.74	0.00	0.00	5.30	0.28	6.21	6.47	∞	∞
TimeWarner	0.97	3.61	6.52	5.84	6.69	6.97	6.30	4.74	6.31	4.49	3.23	5.74	6.35	3.03	0.00	0.00	6.21	6.47	∞	∞
Windstream	0.97	3.61	6.52	5.84	6.69	6.97	6.30	4.74	6.31	4.49	3.23	5.74	6.35	3.03	5.30	0.28	0.00	0.00	∞	∞

TABLE 5.3: Cost map for download throughput and latency

Let's take an example of a querying peer that is located in Cablevision ISP. The cost map downloaded by this peer from the ALTO server informs it that if the querying peer wants to optimize download throughput, it should make connection to peers in the local ISP first (i.e., Cablevision), followed by making connection to peers in MCI (cost: 3.23), Time Warner (5.30), Windstream (6.21), Cox (6.30), Embarq (6.31), Mediacom (6.35), Charter (6.52), Comcast (6.69), respectively. If there are not enough peers in the set of 9 ISPs for which costs are known, then and only then would the querying peer consider making connection to peers in other ISPs (represented by the *Default* ISP). On the other hand, if the querying peer located in Cablevision ISP wanted to optimize for latency, then it would choose peers from the local ISP first (i.e., Cablevision), followed by peers from Time Warner (0.28), Mediacom (3.03), Cox (4.74), Embarq (4.49), MCI (5.74), Charter (5.84), Windstream (6.47), Comcast (6.97), respectively. As before, if there aren't enough peers in the ISPs for which latency costs are known, then and only then would the querying peer consider peers located in other ISPs.

5.5 Social network analysis

Within a given ISP domain, there is a strong incentive to identify regions nodes that show similar behaviour over a given time interval. This can help in identifying potential problems, like for instance misconfigured or under-provisioned network resources, as well as to correlate observed problems over several nodes. In order to unveil groups of nodes that exhibit common temporal characteristics, we compute the covariance matrix for the vector having as entries the random variables that correspond to a feature observed for a unit_id. For illustration, the feature can be the the measured down throughput measured periodically at all unit_ids of an ISP. Links between two unit_ids i and j can be established if the computed entry in the i^{th} row and J^{th} column is significantly large. Such large values will indicate common similar behaviour observed over a given time interval and the induced connected components represent groups of nodes having common behaviour over a given tome frame. Differences between connected components for a same ISP but for different time frames are good markers for temporal variations - for instance if two unit_ids (one located in Chicago and the other in Boston) have been linked for a time interval (January-April) but are no linked any more in May (of the same year) then very probably a significant change in the network capacities, or in the traffic management policies have occurred.

5.6 Network analysis within an ISP

In order to evaluate possible links between individual *unit_ids* of a same ISP we computed several metrics on all the nodes of a single ISP. We selected Comcast as it offers the majority of nodes. In order to determine which nodes have a similar behaviour we computed a Pearson correlation matrix using p-value = 0.003 and correlation value > 0.9. Based on this matrix we constructed for every month an adjacency matrix used to create a graph of the components and how they are connected. Only the *unit_ids* which have a similar behaviour are of interest to us therefore we omitted any *unit_id* with a degree equal to 1. Figure 5.11 illustrates the resulting graph for the month o January. Based on the resulting graph for each month we computed several graph specific metrics as shown in table 5.4. This table shows the results for the numbers of vertexes and edges per month, the density, the number of cluster of connected components and the average number of components per cluster. The density of a graph allows us to determine how populate the graph is compared to the number of available edges. The cluster of connected components gives us the number of subgraphs that have at least two vertices that are connected to each other by paths and the average number of components per cluster gives us the average number of nodes per over all the subgraphs. We also explored several centrality measures of the graphs to determine the relative importance of individual nodes. Graph centrality has four main measures *degree*, *betweenness*, *closeness* and *eigenvector*. All centrality calculation of a vertex v, for a given graph $G := (V, E)$ with $|V|$ vertices and $|E|$ edges. Degree centrality gives a higher score to a node that has a high in/out-degree (5.4). Closeness centrality gives a higher score to a node that has short path distance to every other nodes (5.5). Betweenness centrality gives a higher score to a node that sits on many shortest path of other node pairs (5.6). Eigenvector centrality gives a higher score to a node if it connects to many high score nodes. The computed metrics are illustrated in Fig5.12, 5.13 and 5.14

$$C_D(G) = \frac{\sum_{i=1}^{|V|}[C_D(v*) - C_D(v_i)]}{\sum_{j=1}^{|Y|}[C_D(y*) - C_D(y_i)]} | C_D(v) = deg(v) \tag{5.4}$$

$$C_C(v) = \sum_{t \in V \setminus v} 2^{d_G(v,t)} \tag{5.5}$$

$$C_B(v) = \sum_{s \neq t \neq v \in V} \frac{\sigma_{st}(v)}{\sigma_{st}} \tag{5.6}$$

Based on the metrics and the correlated graph an ISP can construct list of connected components having a similar behaviour with regards to down throughput and latency.

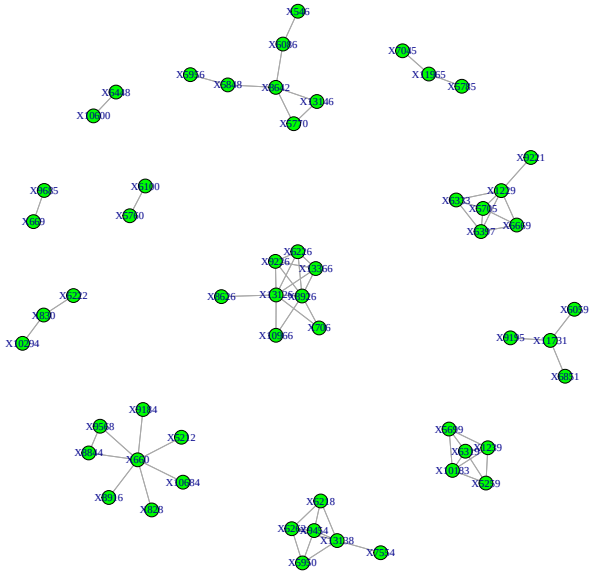


FIGURE 5.11: Correlated components for Comcast in January

month	vertexes	edges	density	clusters	average comp
January	56	138	0.045	12	4.67
February	51	138	0.054	9	5.67
March	40	100	0.064	7	5.72
April	42	112	0.065	6	7
May	55	264	0.089	7	7.86
June	75	372	0.067	9	8.33
July	68	334	0.073	6	11.33
August	36	66	0.052	13	2.77
September	62	338	0.089	7	8.86
October	45	108	0.055	10	4.5
November	47	152	0.070	9	5.22
December	50	114	0.047	12	4.17

TABLE 5.4: Metrics for Comcast ISP

Consider an ISP offering an P2P service over Bit-torrent protocol. If a new peer join the network, it could trigger an ISP maintain ALTO server and get a list of pre-selected peer that offer good down throughput and latency. Similar graphs can be constructed on other metrics depending on the requirements of an application

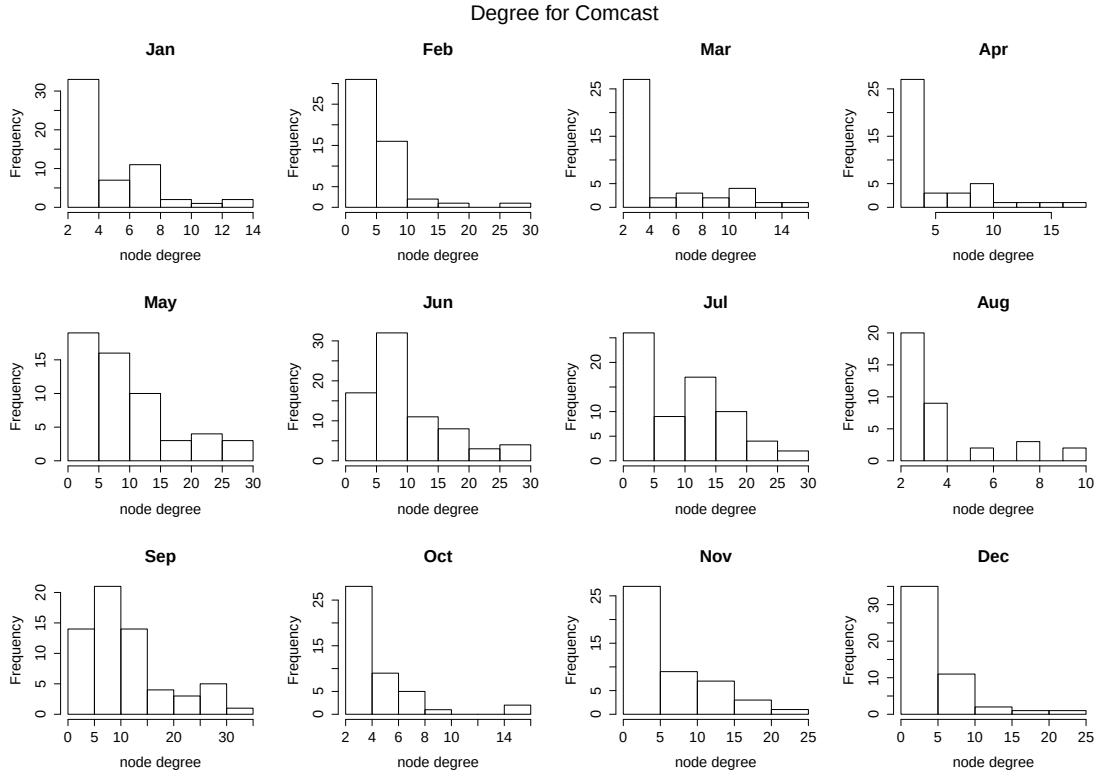


FIGURE 5.12: Degree of the component graph

5.7 Future work

Future research activity will investigate how to extend this approach to wireless networks, where applications requirements can drive the can drive advanced vertical handover schemes. We will research how Application Level Traffic Optimization (ALTO) like servers can exposing such maps towards applications or service providers, by identifying the relevant map topologies and features to be used for seamless media handover schemes. .This is expected to build **location-traffic application maps**. Static network related information can be augmented with location-based information, where GPS coordinates for one or both endpoints can be included. This should lead to discovering correlations between individual locations and specific applications. The challenging part will consist in dealing with moving GPS coordinates during the lifetime of an application, since the velocity of some characteristics (GPS coordinates) will require some "Big Data" like components to analyse in real time the aggregate data of millions of mobile users to derive correlations and to do predictive analytics.

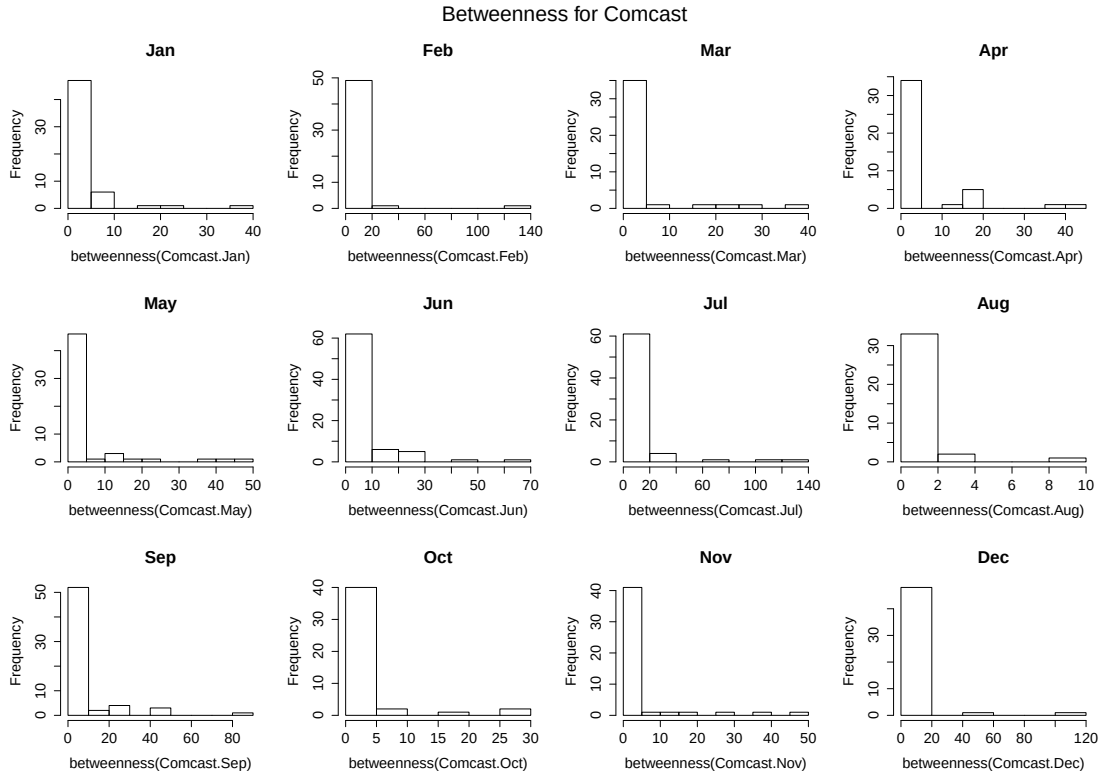


FIGURE 5.13: Betweenness of the component graph

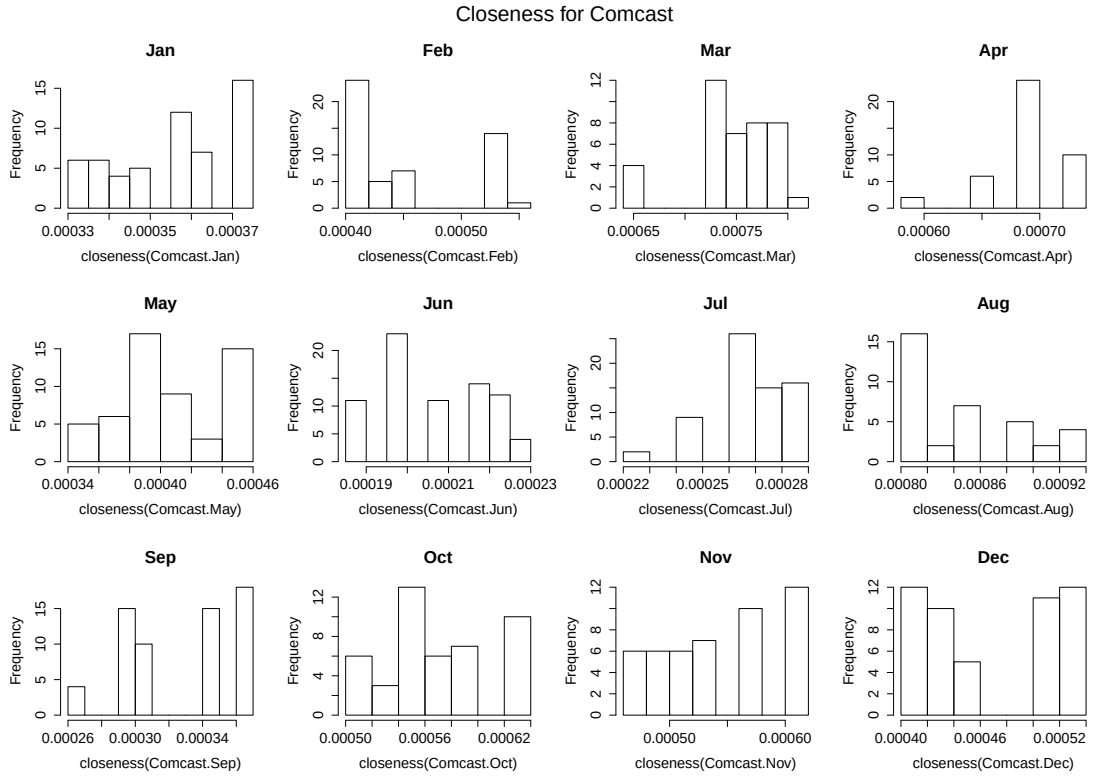


FIGURE 5.14: Closeness of the component graph

Chapter 6

Conclusion and Final Remarks

This thesis has analysed the use and modelling of a heterogeneous communication infrastructure in public safety communication. This chapter summarise the contributions and proposes several future developments.

6.1 Summary of contributions

The initial ideal of this research was to effectively simulate and model heterogeneous communication infrastructures related to public safety networks in order to illustrate the benefits of such an infrastructure in time of crisis. To do this we first had to create realistic model of existing communication architectures. When modelling a communication infrastructure, one needs to have knowledge of how its network is constructed and its capabilities. We also need to know how the traffic looks in time of crisis, and, in general, to be able to construct accurate models for later simulations of possible crisis scenarios. However such data is often not easy to obtain from network operators, who, both to ensure privacy and for business reasons, are unwilling to divulge information about their architecture. The US FCC has been measuring its nationwide performance broadband service with its Measuring Broadband America programme, and offers public access the resulting anonymised data sets. The data does not necessarily reflect the situation in other countries, but the method applied can certainly be used to create accurate models from other data sets.

Issues regarding privacy and business confidentiality also arise when creating models of how a network is utilised. Especially cellular network offer a wide range of different services. Understanding how a population uses a communication infrastructure influences future decisions on expanding and changing this infrastructure. We use the Data for

Development Challenge data set, a country-level data set of mobile phone calls and short message exchanges, to create methodologies to detect abnormal pattern in the call data. These patterns, if correlated with possible candidate events, provide a basis for realistic assumptions and models on how a population reacts to different kinds of event. Our research on the D4D data set and relevant scientific outcomes are published as scientific contributions in [118], while those using the FCC data set are partially presented in [119] and [120]

We then analysed an alternative way to disseminate public safety information among emergency response organisations by using an overlay infrastructure on the Internet, facilitating access to content. As the analysed approach is novel we tested its robustness to attacks by simulating similar attacks performed on the Internet architecture. We prototyped a monitoring architecture for this novel approach that mitigates possible attacks and integrates seamlessly into the Content-Centric Networking architecture. We also implemented a preliminary CCN-compliant firewall allowing the semantic filtering of content. This firewall implementation allows enforcement of policies regarding transmitted content within a network, and also regulates the forwarding of content to other networks, thereby enforcing policies on access rights to the content. The results and scientific contributions obtained are published in [121] and [122].

6.2 Future developments

Twitter offers a complete communication architecture and services from which public safety network communication could benefit. In future work, we intend to analyse how the flow of tweets can be used as an information source, thereby enhancing the overall situational awareness of the incident at hand. If the information is carefully analysed, Twitter streams and their associated metadata can help to provide additional information to public safety dispatch centres even before any of the first responders are on site. Twitter also offers another means of communication between public safety agencies and the affected population.

We also intend to explore ways of quickly and efficiently integrating the results obtained from the real-world data set into existing simulation engine in order to improve the overall quality of simulation. This will allow us to create probabilistic models to integrate into the simulators , enabling validation of our model based on the captured data.

Further work is required to explore the use and feasibility of simulators in modelling large heterogeneous communication infrastructures. Currently simulators tend to focus on one particular communication infrastructure. If we were able to simulate the use

of multiple different communication architectures, we could show how interoperability, reliability and redundancy improve the situation at hand by regulating the traffic in a optimal way such that the different communication architectures are optimally balanced. Investigate the use of large-scale emulation platforms and the possible local deployment of such a platform will also be left for future development.

The collection and capturing of broadband data in the US and the Data for Development Challenge have shown the importance of real-world data to validate existing models and create more accurate models for future use. Expanding the measurements done by the FCC to Luxembourg and not only focussing on wired broadband data offers a unique opportunity to create a country-level data set collecting information from multiple actors in order to analyse the data and discover possible dependencies among the multiple actors involved.

Appendix A

Entity-Relation Database model

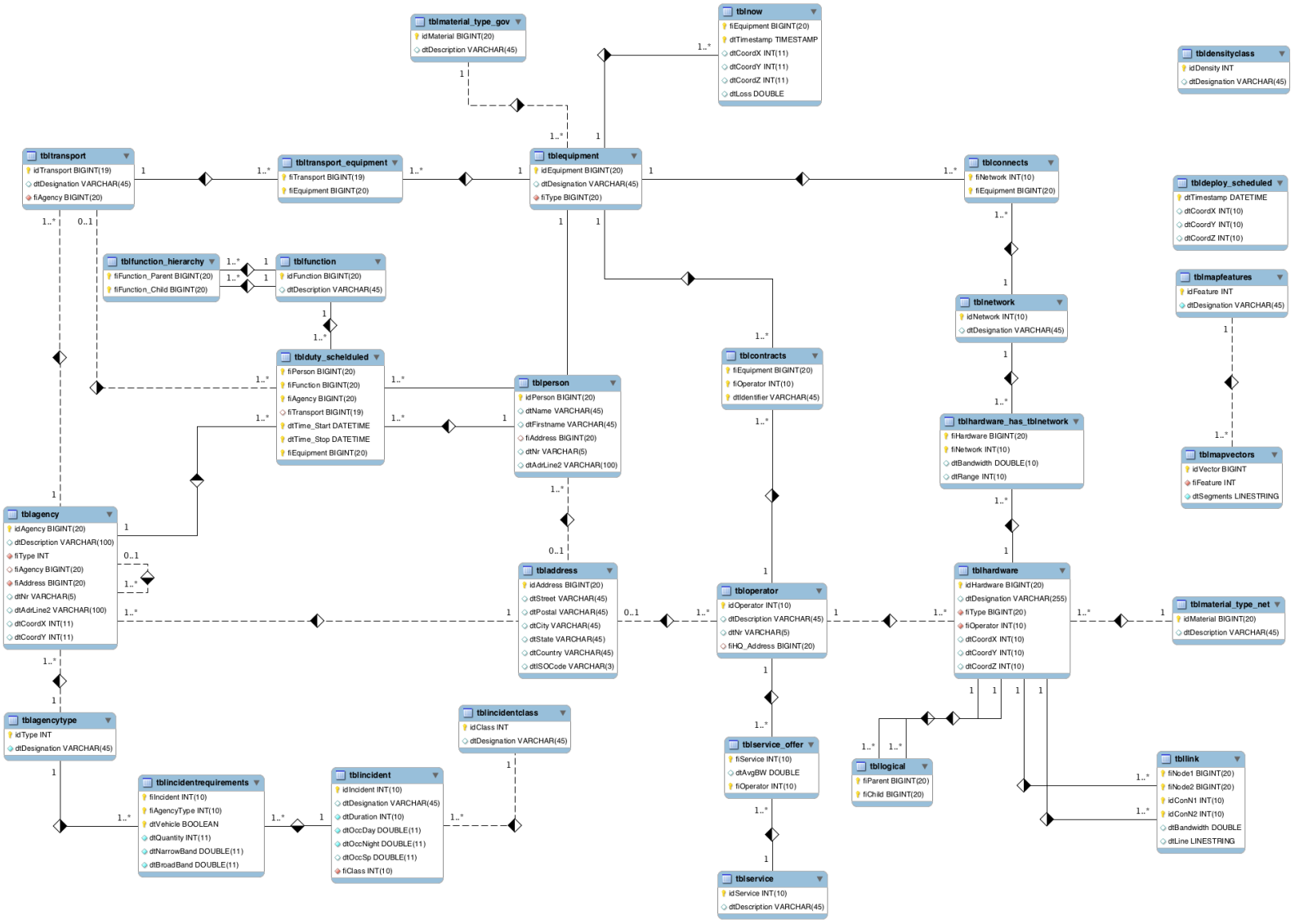


FIGURE A.1: Entity-relation database model

Appendix B

FCC Data dictionary for Measuring Broadband America programme

unit_id	Unique identifier for an individual unit
dtime	Time test finished in UTC
target	Target hostname or IP address
rtt_avg	Average RTT in microseconds
rtt_min	Minimum RTT in microseconds
rtt_max	Maximum RTT in microseconds
rtt_std	Standard Deviation in Measured RTT in microseconds
successes	Number of successes
failiures	Number of failures
location_id	Please ignore (this is an internal key mapping to unit profile data)

Table B.1: curr_dlping

unit_id	Unique identifier for an individual unit
dtime	Time test finished in UTC
nameserver	Nameserver used to handle the DNS request
lookup_host	Hostname to be resolved
response_ip	Field unused at present
rtt	DNS resolution time in microseconds
successes	Number of successes (always 1 or 0 for this test)
failures	Number of failures (always 1 or 0 for this test)
location_id	Please ignore (this is an internal key mapping to unit profile data)

Table B.2: curr_dns

unit_id	Unique identifier for an individual unit
dtime	Time test finished in UTC
target	Target hostname or IP address
address	The IP address of the server (resolved by the client's DNS)
fetch_time	Time the test ran for in microseconds
bytes_total	Total bytes downloaded across all connections
bytes_sec	Running total of throughput, which is sum of speeds measured for each stream (in bytes/sec), from the start of the test to the current interval
bytes_sec_interval	Throughput at this specific interval (e.g. Throughput between 25-30 seconds)
warmup_time	Time consumed for all the TCP streams to arrive at optimal window size (Units: microseconds)
warmup_bytes	Bytes transferred for all the TCP streams during the warm-up phase.
sequence	The interval that this row refers to (e.g. in the US, sequence=0 implies result is for 0-5 seconds of the test)
threads	The number of concurrent TCP connections used in the test
successes	Number of successes (always 1 or 0 for this test)
failures	Number of failures (always 1 or 0 for this test)
location_id	Please ignore (this is an internal key mapping to unit profile data)

Table B.3: curr_httpgetmt

unit_id	Unique identifier for an individual unit
dtime	Time test finished in UTC
target	Target hostname or IP address
address	The IP address of the server (resolved by the client's DNS)
fetch_time	Time the test ran for in microseconds
bytes_total	Total bytes downloaded across all connections
bytes_sec	Running total of throughput, which is sum of speeds measured for each stream (in bytes/sec), from the start of the test to the current interval
bytes_sec_interval	Throughput at this specific interval (e.g. Throughput between 25-30 seconds)
warmup_time	Time consumed for all the TCP streams to arrive at optimal window size (Units: microseconds)
warmup_bytes	Bytes transferred for all the TCP streams during the warm-up phase.
sequence	The interval that this row refers to (e.g. in the US, sequence=0 implies result is for 0-5 seconds of the test)
threads	The number of concurrent TCP connections used in the test
successes	Number of successes (always 1 or 0 for this test)
failures	Number of failures (always 1 or 0 for this test)
location_id	Please ignore (this is an internal key mapping to unit profile data)

Table B.4: curr_httppostmt

unit_id	Unique identifier for an individual unit
dtime	Time test finished in UTC
target	Target hostname or IP address
rtt_avg	Average RTT in microseconds
rtt_min	Minimum RTT in microseconds
rtt_max	Maximum RTT in microseconds
rtt_std	Standard Deviation in Measured RTT in microseconds
successes	Number of successes
failiures	Number of failures
location_id	Please ignore (this is an internal key mapping to unit profile data)

Table B.5: curr_ping & ICMP based

unit_id	Unique identifier for an individual unit
dtime	Time test finished in UTC
target	Target hostname or IP address
packet_size	Size of each UDP Datagram (Units: Bytes)
stream_rate	Rate at which the UDP stream is generated (Units: bits/sec)
duration	Total duration of test (Units: microseconds)
packets_up_sent	Number of packets sent in Upstream (measured by client)
packets_down_sent	Number of packets sent in Downstream (measured by server)
packets_up_recv	Number of packets received in Upstream (measured by server)
packets_down_recv	Number of packets received in Downstream (measured by client)
jitter_up	Upstream Jitter measured (Units: microseconds)
jitter_down	Downstream Jitter measured (Units microseconds)
latency	99th percentile of round trip times for all packets
successes	Number of successes (always 1 or 0 for this test)
failures	Number of failures (always 1 or 0 for this test)
location_id	Please ignore (this is an internal key mapping to unit profile data)

Table B.6: curr_udpjitter

unit_id	Unique identifier for an individual unit
dtime	Time test finished in UTC
target	Target hostname or IP address
rtt_avg	Average RTT in microseconds
rtt_min	Minimum RTT in microseconds
rtt_max	Maximum RTT in microseconds
rtt_std	Standard Deviation in Measured RTT in microseconds
successes	Number of successes (note: use failures/(successes+failures)) for packet loss)
failiures	Number of failures (packets lost)
location_id	Please ignore (this is an internal key mapping to unit profile data)

Table B.7: curr_udplatency & UDP based

unit_id	Unique identifier for an individual unit
dtime	Time test finished in UTC
target	Target hostname or IP address
rtt_avg	Average RTT in microseconds
rtt_min	Minimum RTT in microseconds
rtt_max	Maximum RTT in microseconds
rtt_std	Standard Deviation in Measured RTT in microseconds
successes	Number of successes
failures	Number of failures
location_id	Please ignore (this is an internal key mapping to unit profile data)

Table B.8: curr_ulping

unit_id	Unique identifier for an individual unit
dtime	Time test finished in UTC
target	Hostname of server
downthrpt	This metric records the average downstream throughput for the entire duration of the test. This average is calculated by taking the mean average of the speed of completing each of the blocks downloaded (as specified with the -css client parameter) in bytes/sec
downjitter	A measure of the standard deviation of the speed each block was downloaded at in microseconds
latency	The mean of all of the ping round-trip-times sent from client to server in microseconds
jitter	The standard deviation of the ping round-trip-times sent from client to server in microseconds
buffer_underruns	this metric records how many times the buffer was completely drained (i.e. the client could not receive data at the desired downstream rate, so the buffer emptied). A good connection will have zero buffer underruns.
buffer_delay	The total time in microseconds that the client spent waiting for the buffer to reach its minimum size
buffer_filltime	The total time in microseconds to fill the buffer
duration	Duration of the test in microseconds
bitrate	Stream rate in bytes/sec
buffer_size	Buffer size used in bytes
successes	Number of successes (always 1 or 0 for this test)
failures	Number of failures (always 1 or 0 for this test)
location_id	Please ignore (this is an internal key mapping to unit profile data)

Table B.9: curr_videostream

unit_id	Unique identifier for an individual unit
dtime	Time test finished in UTC
target	URL to fetch
address	IP address connected to to fetch content from initial URL
fetch_time	Sum of time consumed to download Html content and then concurrently download all resources (Units: micorsecounds)
bytes_total	Sum of HTML content size and all resources size (Units : Bytes)
bytes_sec	Average speed of downloading HTML content and then concurrently downloading all resources (Units: bytes/sec)
objects	Number of Resources (images, css etc) downloaded
threads	Maximum number of concurrent threads allowed
requests	Total number of HTTP requests made
connections	Total number of TCP connections established
reused_connections	Number of TCP connections re-used
lookups	Number of DNS lookups performed
request_total_time	Total duration of all requests summed together, if made sequentially
request_min_time	Shortest request duration
request_avg_time	Average request duration
request_max_time	Longest request duration
ttfb_total_time	Total duration of the time-to-first-byte summed together, if made sequentially
ttfb_min_time	Shortest time-to-first-byte duration
ttfb_avg_time	Average time-to-first-byte duration
ttfb_max_time	Longest time-to-first-byte duration
lookup_total_time	Total duration of all DNS lookups summed together, if made sequentially
lookup_min_time	Shortest DNS lookup duration
lookup_avg_time	Average DNS lookup duration
lookup_max_time	Longest DNS lookup duration
successes	Number of successes
failures	Number of failures
location_id	Please ignore (this is an internal key mapping to unit profile data)

Table B.10: curr_webget

unit_id	Unique identifier for an individual unit
dtime	Time test finished in UTC
wan_rx_bytes	Total bytes recevied via the WAN interface on the unit (incl. Ethernet/IP headers)
wan_tx_bytes	Total bytes transmitted via the WAN interface on the unit (incl. Ethernet/IP headers)
sk_rx_bytes	Bytes received as a result of active performance measurements
sk_tx_bytes	Bytes transmitted as a result of active performance measurements
location_id	Please ignore (this is an internal key mapping to unit profile data)

Table B.11: curr_netusage

Appendix C

Public safety

We propose Luxembourg as an ideal test bed to implement such a heterogeneous communication infrastructure by taking into account all possibly involved governmental actors in times of a catastrophe.

C.1 Networks

TETRA(TErrestrial Trunked RAdio)[123] was designed solemnly for governmental agencies, public safety, military and other actors. It is widely used across Europe. It offers some useful functionalities but comes at a high cost on end devices an investment in a new infrastructure. GSM(Global System for Mobile Communications) often compared to TETRA has fewer functionalities but the costs are much less for acquiring new equipment and infrastructure is already in place by private mobile operators. UMTS(Universal Mobile Telecommunications System) is the 3rd generation follow-up of GSM therefore at its core bears a similar architecture. Equipment cost is comparable to GSM. LTE(3GPP Long Term Evolution) is again a successor of UMTS and comes yet again with a higher bandwidth and lower latency compared to previous implementations. LTE however is still in the process of being deployed and therefore availability is quite limited to larger metropolitan areas. These cellular technologies are mostly deployed by private operator and therefore, one must consider that public safety is not their prime business. WiFi or WLAN is used if the communication is within a few 100 meters maximum. COTS[1] equipment can be use and they come at a low cost. WiMax(Worldwide Interoperability for Microwave Access) is another wireless technology but with a high reachability than WiFi(few kilometers). As mentioned before when landline and cellular/wireless network do not work anymore or if operations take place in a foreign country far from

[1] commercial of the shelf

the own communication infrastructure, satellite communication can be used to overcome this problem. However the latency largely depend on the type of satellite used for communication.

C.2 Database

The database we propose, illustrated in Figure C.1, serves as raw data model for our simulation environment. The database is separated into several sections each handling specific requirements from the communication infrastructure. First of all there need to be a two fold static inventory, one that represents the active stock of every end devices own by the actors involved and one that captures every network component deployed in the communication infrastructure. This two-fold inventory effectively separates the components which offer a certain network capabilities from those which ask for specific network requirements. This static inventory, highlighted in green in Figure C.1, will provide the essential information when we simulate the deployment of certain component as all of these static information still location and time independent however these two parameter are essential to a crisis. Therefore the next step consists of assigning each components a time parameter, highlighted in ocher in Figure C.1. For static and fixed devices this may not be needed but mobile end devices such as phones or laptops might not always be deployed by the same person or at the same time. Furthermore this allows us to create routine plan of users and end devices as well as mobile network equipment simulating maintenance runs, shift working and vacation of end device user. This represents the plans of officers of the people assigned during a specific time interval. We also have to take into account possible last minute change on such deployment schedules, highlighted in brown in Figure C.1. This represents for example the possibility of a certain end device user calling in sick or the spontaneous need of more users than actually planed. This last modification will serve as ground knowledge when starting up the simulation environment. The final part consist of assigning each component of the simulation to specific initial location, highlighted in violet in Figure C.1. This can either be fixed, for network components for example, or in case of mobile device, usually the agencies headquarters. Now we have everything prepared for being used in a simulation.

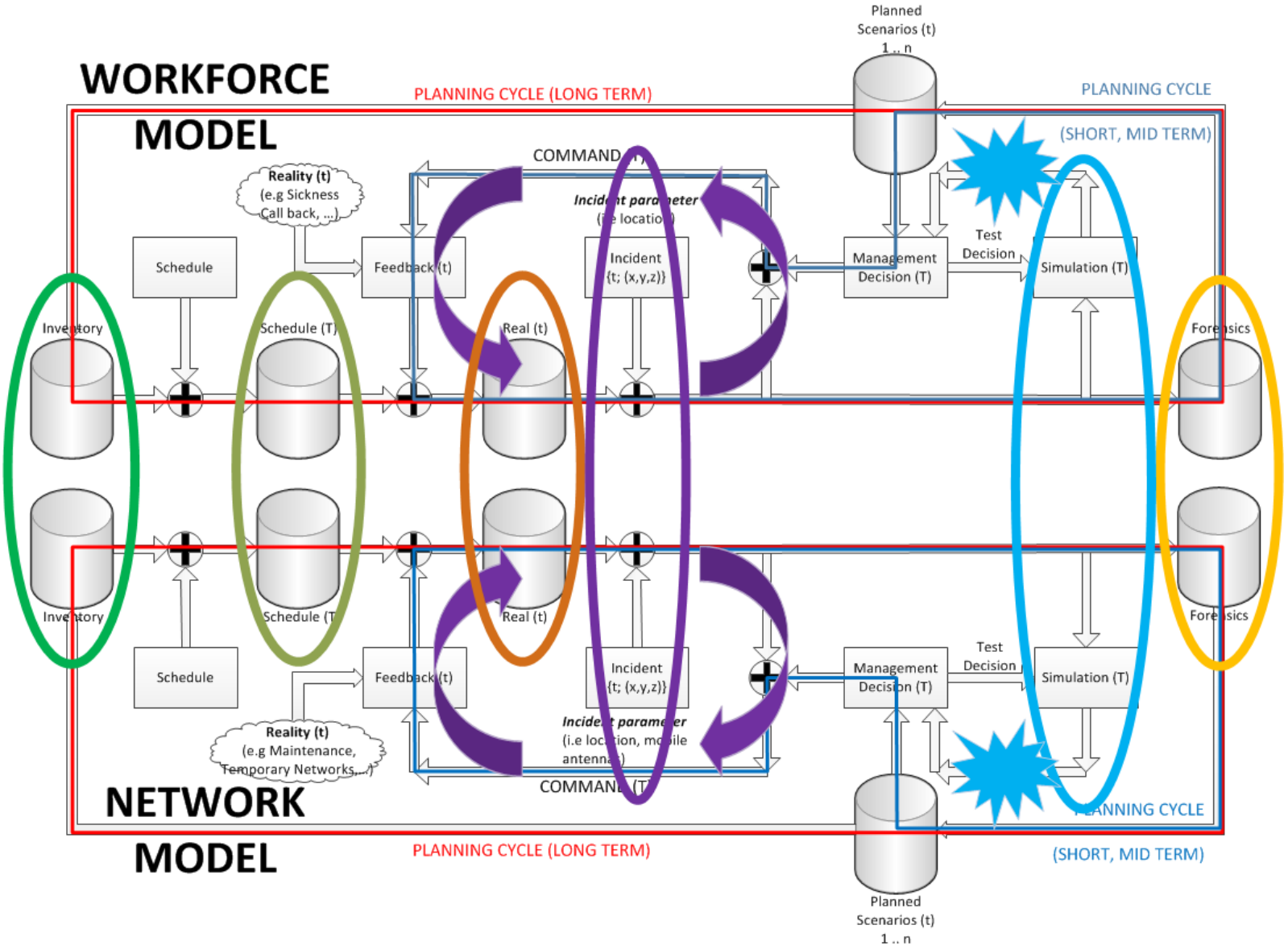


FIGURE C.1: Database model

In our environment we suppose that mobile device are able to transmit their physical location to the database which is then processed by the simulation environment. The data transmitted by the end devices is stores in two different data storages. One storage is use actively in the simulation environment and serves as real world data provider, highlighted in light blue in Figure C.1. This data will also be used to improve the underlying data models by comparing the predictions in the simulation with the actual available data and adjusting the simulation parameters if necessary. A second data storage is used for replayability of incidents and for forensic analysis, highlighted in orange in Figure C.1. The idea is to simulate past incidents and test various alternative decision and the impact of those on the incident and its resolution. Furthermore the forensic analysis allows for a long term decision and planning process to improve the current communication infrastructure. The system also provides a possible to store planned scenarios. These planned scenarios are prepared solutions for given incidents and are used as base in case of incidents that have a similar scale in terms of people involved, population and public safety personnel.

C.3 Visual representation GUI optimised for PSC operator

Our visual interface is designed to serve multiple actors. Public safety operatives will have an overview of the deployed units as well as their equipments. It provides all essential information to support decision taking upon real feedback form the deployed personnel. This involves hierarchical structures of the personnel to determine the people in charge and their subordinates as well as possible additional resources from neighbouring agencies. It also provides a mean to create operation task forces on the fly. This is important in crisis management as structures are create on the fly between actors that might usually not work together but must now for the greater good.

Network manager however do not need this kind of information. They are interested in the health of the communication infrastructure. For those operators a more logical view of the ongoing communication are on interest as they will see the communication flows among the different actors as well as the load produce on the infrastructure. This will help to determine possible bottlenecks in the communication infrastructures as well as determine the redundancies available. The simulation environment will also support decision taking for future investments in the communication infrastructure. Network operators can deploy virtual equipment pieces in a simulation and evaluate their impact on the overall communication infrastructure for effective planning and investment.

Bibliography

[1] S. R. Veil, T. Buehner, and M. J. Palenchar, "A work-in-process literature review: Incorporating social media in risk and crisis communication," *Journal of Contingencies and Crisis Management*, vol. 19, no. 2, pp. 110–122, 2011. [Online]. Available: 10.1111/j.1468-5973.2011.00639.x

[2] D. E. Alexander, "Social media in disaster risk reduction and crisis management," *Science and Engineering Ethics*, pp. 1–17, 2013. [Online]. Available: http://dx.doi.org/10.1007/s11948-013-9502-z

[3] F. Atefeh and W. Khreich, "A survey of techniques for event detection in twitter," *Computational Intelligence*, vol. 0, pp. n/a–n/a, 2013. [Online]. Available: http://dx.doi.org/10.1111/coin.12017

[4] S. Karimi, J. Yin, and C. Paris, "Classifying microblogs for disasters," in *Proceedings of the 18th Australasian Document Computing Symposium*, ser. ADCS '13. New York, NY, USA: ACM, 2013, pp. 26–33. [Online]. Available: http://doi.acm.org/10.1145/2537734.2537737

[5] M. Imran, S. Elbassuoni, C. Castillo, F. Diaz, and P. Meier, "Practical extraction of disaster-relevant information from social media," in *Proceedings of the 22Nd International Conference on World Wide Web Companion*, ser. WWW '13 Companion. Republic and Canton of Geneva, Switzerland: International World Wide Web Conferences Steering Committee, 2013, pp. 1021–1024. [Online]. Available: http://dl.acm.org/citation.cfm?id=2487788.2488109

[6] I. Helsloot and J. Groenendaal, "Twitter: An underutilized potential during sudden crises?" *Journal of Contingencies and Crisis Management*, vol. 21, no. 3, pp. 178–183, 2013. [Online]. Available: http://dx.doi.org/10.1111/1468-5973.12023

[7] C.-C. Chang and C.-J. Lin, "Libsvm: A library for support vector machines," *ACM Trans. Intell. Syst. Technol.*, vol. 2, no. 3, pp. 27:1–27:27, May 2011. [Online]. Available: http://doi.acm.org/10.1145/1961189.1961199

[8] M. Imran, S. M. Elbassuoni, C. Castillo, F. Diaz, and P. Meier, "Extracting information nuggets from disaster-related messages in social media," in *ISCRAM' 13: Proceedings of the 10th International ISCRAM Conference*, vol. 26, 2013. [Online]. Available: http://chato.cl/papers/imran_elbassuoni_castillo_diaz_meier_2013_extracting_information_nuggets_disasters.pdf

[9] J. D. Lafferty, A. McCallum, and F. C. N. Pereira, "Conditional random fields: Probabilistic models for segmenting and labeling sequence data," in *Proceedings of the Eighteenth International Conference on Machine Learning*, ser. ICML '01. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 2001, pp. 282–289. [Online]. Available: http://dl.acm.org/citation.cfm?id=645530.655813

[10] M. Hall, E. Frank, G. Holmes, B. Pfahringer, P. Reutemann, and I. H. Witten, "The weka data mining software: An update," *SIGKDD Explor. Newsl.*, vol. 11, no. 1, pp. 10–18, Nov. 2009. [Online]. Available: http://doi.acm.org/10.1145/1656274.1656278

[11] F. Abel, C. Hauff, G.-J. Houben, R. Stronkman, and K. Tao, "Semantics + filtering + search = twitcident. exploring information in social web streams," in *Proceedings of the 23rd ACM Conference on Hypertext and Social Media*, ser. HT '12. New York, NY, USA: ACM, 2012, pp. 285–294. [Online]. Available: http://doi.acm.org/10.1145/2309996.2310043

[12] G. Valkanas and D. Gunopulos, "Location extraction from social networks with commodity software and online data," in *Proceedings of the 2012 IEEE 12th International Conference on Data Mining Workshops*, ser. ICDMW '12. Washington, DC, USA: IEEE Computer Society, 2012, pp. 827–834. [Online]. Available: http://dx.doi.org/10.1109/ICDMW.2012.128

[13] C. Weidemann and J. Swift, "Social media location intelligence: The next privacy battle-an arcgis add-in and analysis of geospatial data collected from twitter. com." *International Journal of Geoinformatics*, vol. 9, no. 2, 2013.

[14] A. MacEachren, A. Jaiswal, A. Robinson, S. Pezanowski, A. Savelyev, P. Mitra, X. Zhang, and J. Blanford, "Senseplace2: Geotwitter analytics support for situational awareness," in *Visual Analytics Science and Technology (VAST), 2011 IEEE Conference on*, Oct 2011, pp. 181–190. [Online]. Available: http://dx.doi.org/10.1109/VAST.2011.6102456

[15] G. Valkanas, D. Gunopulos, I. Boutsis, and V. Kalogeraki, "An architecture for detecting events in real-time using massive heterogeneous data sources," in *Proceedings of the 2Nd International Workshop on Big Data, Streams and*

Heterogeneous Source Mining: Algorithms, Systems, Programming Models and Applications, ser. BigMine '13. New York, NY, USA: ACM, 2013, pp. 103–109. [Online]. Available: http://doi.acm.org/10.1145/2501221.2501235

[16] T. Sakaki, M. Okazaki, and Y. Matsuo, "Earthquake shakes twitter users: Real-time event detection by social sensors," in *Proceedings of the 19th International Conference on World Wide Web*, ser. WWW '10. New York, NY, USA: ACM, 2010, pp. 851–860. [Online]. Available: http://doi.acm.org/10.1145/1772690.1772777

[17] D. Fox, D. Schulz, G. Borriello, J. Hightower, and L. Liao, "Bayesian filtering for location estimation," *IEEE pervasive computing*, vol. 2, no. 3, pp. 24–33, 2003.

[18] J. Hightower and G. Borriello, "Particle filters for location estimation in ubiquitous computing: A case study," in *UbiComp 2004: Ubiquitous Computing*, ser. Lecture Notes in Computer Science, N. Davies, E. Mynatt, and I. Siio, Eds. Springer Berlin Heidelberg, 2004, vol. 3205, pp. 88–106. [Online]. Available: http://dx.doi.org/10.1007/978-3-540-30119-6_6

[19] A. Ritter, Mausam, O. Etzioni, and S. Clark, "Open domain event extraction from twitter," in *Proceedings of the 18th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, ser. KDD '12. New York, NY, USA: ACM, 2012, pp. 1104–1112. [Online]. Available: http://doi.acm.org/10.1145/2339530.2339704

[20] R. D. Waters and J. M. Williams, "Squawking, tweeting, cooing, and hooting: analyzing the communication patterns of government agencies on twitter," *Journal of Public Affairs*, vol. 11, no. 4, pp. 353–363, 2011. [Online]. Available: http://dx.doi.org/10.1002/pa.385

[21] F. Schultz, S. Utz, and A. Göritz, "Is the medium the message? perceptions of and reactions to crisis communication via twitter, blogs and traditional media," *Public Relations Review*, vol. 37, no. 1, pp. 20 – 27, 2011. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0363811110001281

[22] T. Sakaki, F. Toriumi, K. Uchiyama, Y. Matsuo, K. Shinoda, K. Kazama, S. Kurihara, and I. Noda, "The possibility of social media analysis for disaster management," in *Humanitarian Technology Conference (R10-HTC), 2013 IEEE Region 10*, Aug 2013, pp. 238–243. [Online]. Available: http://dx.doi.org/10.1109/R10-HTC.2013.6669048

[23] R. Power, B. Robinson, and C. Wise, "Comparing web feeds and tweets for emergency management," in *Proceedings of the 22Nd International Conference on World Wide Web Companion*, ser. WWW '13 Companion.

Republic and Canton of Geneva, Switzerland: International World Wide Web Conferences Steering Committee, 2013, pp. 1007–1010. [Online]. Available: http://dl.acm.org/citation.cfm?id=2487788.2488103

[24] J. Yin, S. Karimi, B. Robinson, and M. Cameron, "Esa: Emergency situation awareness via microbloggers," in *Proceedings of the 21st ACM International Conference on Information and Knowledge Management*, ser. CIKM '12. New York, NY, USA: ACM, 2012, pp. 2701–2703. [Online]. Available: http://doi.acm.org/10.1145/2396761.2398732

[25] M. A. Cameron, R. Power, B. Robinson, and J. Yin, "Emergency situation awareness from twitter for crisis management," in *Proceedings of the 21st International Conference Companion on World Wide Web*, ser. WWW '12 Companion. New York, NY, USA: ACM, 2012, pp. 695–698. [Online]. Available: http://doi.acm.org/10.1145/2187980.2188183

[26] R. Chen and Y. Sakamoto, "Perspective matters: Sharing of crisis information in social media," in *System Sciences (HICSS), 2013 46th Hawaii International Conference on*, Jan 2013, pp. 2033–2041. [Online]. Available: http://dx.doi.org/10.1109/HICSS.2013.447

[27] A. M. MacEachren, A. C. Robinson, A. Jaiswal, S. Pezanowski, A. Savelyev, J. Blanford, and P. Mitra, "Geo-twitter analytics: Applications in crisis management," in *25th International Cartographic Conference*, 2011, pp. 3–8. [Online]. Available: http://www.geovista.psu.edu/publications/2011/MacEachren_ICC_2011.pdf

[28] F. Abel, C. Hauff, G.-J. Houben, R. Stronkman, and K. Tao, "Twitcident: Fighting fire with information from social web streams," in *Proceedings of the 21st International Conference Companion on World Wide Web*, ser. WWW '12 Companion. New York, NY, USA: ACM, 2012, pp. 305–308. [Online]. Available: http://doi.acm.org/10.1145/2187980.2188035

[29] R. Stronkman, "Twitcident: Filtering twitter to obtain real-time intelligence," June 2012. [Online]. Available: http://irgc.org/wp-content/uploads/2012/06/Twitcident-OECD-IRGC-Expert-Workshop.pdf

[30] T. Terpstra, A. de Vries, R. Stronkman, and G. Paradies, "Towards a realtime twitter analysis during crises for operational crisis management," in *ISCRAM'12: Proceedings of the 9th International ISCRAM Conference*, 2012. [Online]. Available: http://www.iscramlive.org/ISCRAM2012/proceedings/172.pdf

[31] FCC's Office of Engineering and Technology and Consumer and Governmental Affairs Bureau, "2012 measuring broadband america: July report," Technical Appendix, United States Federal Communications Commission, Tech. Rep., 2012. [Online]. Available: http://www.fcc.gov/measuring-broadband-america/2012/july

[32] V. D. Blondel, M. Esch, C. Chan, F. Clerot, P. Deville, E. Huens, and F. Morlot, "Data for Development: the D4D Challenge on Mobile Phone Data," 2013.

[33] R. of Austria, "Austrian federal acton provisions facilitating electronic communications with public bodies," Federal law Gazette, part I, Nr.10/2004, 2004.

[34] H. Leitold, R. Posch, and T. Rössler, "Media-break resistant esignatures in egovernment ? an austrian experience," in *Emerging Challenges for Security, Privacy, and Trust - 24th IFIP SEC*, ser. IFIP Advances in Information and Communication Technologies, J. L. Dimitris Gritzalis, Ed., vol. IFIP AICT 297. Springer, 2009, pp. 109 – 118.

[35] ——, "Reconstruction of electronic signatures from edocument printouts," *Computers & security*, vol. 29, pp. 523 – 532, 2010.

[36] R. Posch, "A federated identity management architecture for cross-border services in europe," in *BIOSIG*, 2008, pp. 141–152.

[37] H. Leitold and R. Posch, *STORK - Technical Approach and Privacy book title: Digital Enlightenment Yearbook 2012*. Bus, J. , Crompton, M. , Hildebrandt, M. , Metakides, G, 2012, pp. 289 – 306.

[38] J. Seedorf and E. Burger, "Application-Layer Traffic Optimization (ALTO) Problem Statement," RFC 5693 (Informational), Internet Engineering Task Force, Oct. 2009. [Online]. Available: http://www.ietf.org/rfc/rfc5693.txt

[39] R. Alimi, R. Penno, and R. Yang, "ALTO protocol," Working Draft, IETF Secretariat, Internet-Draft draft-ietf-alto-protocol-27.txt, Mar. 2014.

[40] "The Network Simulator ns-2." [Online]. Available: http://nsnam.isi.edu/nsnam/index.php/Main_Page

[41] T. Issariyakul and E. Hossain, *Introduction to Network Simulator NS2*, Springer Berlin Heidelberg, Ed. Springer, 2009.

[42] "OPNET Modeler." [Online]. Available: http://www.opnet.com/solutions/network_rd/modeler.html

[43] "Qualnet: Scalable Network Technologies." [Online]. Available: http://www.scalable-networks.com/

[44] "The Network Simulator ns-3." [Online]. Available: http://www.nsnam.org/wiki/index.php/Main_Page

[45] G. F. Riley and T. R. Henderson, "The ns-3 Network Simulator," in *Modeling and Tools for Network Simulation*. Springer Berlin Heidelberg, 2010. [Online]. Available: http://dx.doi.org/10.1007/978-3-642-12331-3_2

[46] A. Khan, S. Bilal, and M. Othman, "A performance comparison of open source network simulators for wireless networks," in *Control System, Computing and Engineering (ICCSCE), 2012 IEEE International Conference on*, Nov 2012, pp. 34–38. [Online]. Available: http://dx.doi.org/10.1109/ICCSCE.2012.6487111

[47] E. Weingartner, H. vom Lehn, and K. Wehrle, "A performance comparison of recent network simulators," in *Communications, 2009. ICC '09. IEEE International Conference on*, June 2009, pp. 1–5. [Online]. Available: http://dx.doi.org/10.1109/ICC.2009.5198657

[48] T. Camp, J. Boleng, and V. Davies, "A survey of mobility models for ad hoc network research," *WIRELESS COMMUNICATIONS & MOBILE COMPUTING (WCMC): SPECIAL ISSUE ON MOBILE AD HOC NETWORKING: RESEARCH, TRENDS AND APPLICATIONS*, vol. 2, pp. 483–502, 2002.

[49] X. Zeng, R. Bagrodia, and M. Gerla, "GloMoSim: A Library for Parallel Simulation of Large-scale Wireless Networks," in *in Workshop on Parallel and Distributed Simulation*, 1998.

[50] L. Bajaj, M. Takai, R. Ahuja, K. Tang, R. Bagrodia, and M. Gerla, "GloMoSim: A Scalable Network Simulation Environment," University of California, Los Angeles; Computer Science Department, Tech. Rep., 1999.

[51] *ns-3 Tutorial*, August 2010.

[52] *ns-3 Reference Manuel*, August 2010.

[53] "The U-2010 project." [Online]. Available: http://www.u-2010.eu/

[54] *NATO Architecture Framework*, 3rd ed., November 2007.

[55] T. Buckman, "NATO Network Enabled Capability Feasibility Study Executive Summary: Version 2.0," NATO Consulation, Command and Control Agency, Tech. Rep., October 2005.

[56] A. Domingo and M. A. Rico, "NATO Network Enabled Capability Frequently Asked Questions," NATO Allied Command Transformation, C4I/NNEC, Tech. Rep., March 2010.

[57] K. Arnell and D. Akerstrom and J. Busch and M. Eriksson and U. Johansson and F. Jordan and P. Lee and L. Nystrøm and M. Roper and W. Ressel and B. Sjogren and M. Sjoquist and H. Stolt and T. Skaar and K. Veum and J. Vlug, "NATO and Sweden Joint Live Experiement on NEC: A first step towards a NEC realization," NATO Consulation, Command and Control Agency, Tech. Rep., March 2009.

[58] V. Jacobson, D. K. Smetters, J. D. Thornton, M. F. Plass, N. H. Briggs, and R. L. Braynard, "Networking named content," in *Proceedings of the 5th international conference on Emerging networking experiments and technologies*, ser. CoNEXT '09. New York, NY, USA: ACM, 2009, pp. 1–12.

[59] L. Zhang, D. Estrin, J. Burke, V. Jacobson, J. D. Thornton, D. K. Smetters, B. Zhang, G. Tsudik, kc claffy, D. Krioukov, D. Massey, C. Papadopoulos, T. Abdelzaher, L. Wang, P. Crowley, and E. Yeh, "Named Data Networking (NDN) Project," October 2010.

[60] H. Schulze and K. Mochalski, "Internet study 2008/2009," 2009.

[61] D. R. Cheriton and M. Gritter, "Triad: A new next-generation internet architecture," July 2000.

[62] B. Baccala, "Data-oriented networking," INTERNET-DRAFT, August 2002.

[63] T. Koponen, M. Chawla, B.-G. Chun, A. Ermolinskiy, K. H. Kim, S. Shenker, and I. Stoica, "A data-oriented (and beyond) network architecture," in *Proceedings of the 2007 conference on Applications, technologies, architectures, and protocols for computer communications*, ser. SIGCOMM '07. New York, NY, USA: ACM, 2007, pp. 181–192.

[64] M. Särelä, T. Rinta-aho, and S. Tarkoma, "RTFM: Publish/Subscribe Internetworking Architecture," *ICT-MobileSummit Conference*, 2008.

[65] C. Dannewitz, M. Herlich, E. Bauer, M. Becker, F. Beister, N. Dertmann, R. Hrestic, M. Kionka, M. Mohr, M. Mühe, D. Murali, F. Steffen, S. Stey, E. Unruh, Q. Wang, and S. Weber, "Opennetinf documentation design and implementation," September 2011.

[66] "Named Data Networking." [Online]. Available: http://named-data.net

[67] A. Ghodsi, S. Shenker, T. Koponen, A. Singla, B. Raghavan, and J. Wilcox, "Information-centric networking: seeing the forest for the trees," in *Proceedings of the 10th ACM Workshop on Hot Topics in Networks*, ser. HotNets '11. New York, NY, USA: ACM, 2011, pp. 1:1–1:6.

[68] V. Jacobson, M. Mosko, D. Smetters, and J. J. Garcia-Luna-Aceves, "Content-centric networking: Whitepaper describing future assurable global networks," Response to DARPA RFI SN07-12, 2007.

[69] "Content Centric Networking." [Online]. Available: http://www.ccnx.org

[70] V. Jacobson, D. K. Smetters, N. H. Briggs, M. F. Plass, P. Stewart, J. D. Thornton, and R. L. Braynard, "VoCCN: voice-over content-centric networks," in *Proceedings of the 2009 workshop on Re-architecting the internet*, ser. ReArch '09. New York, NY, USA: ACM, 2009, pp. 1–6.

[71] G. Carofiglio, M. Gallo, L. Muscariello, and D. Perino, "Modeling data transfer in content-centric networking," in *Proceedings of the 23rd International Teletraffic Congress*, ser. ITC '11. ITCP, 2011, pp. 111–118. [Online]. Available: http://dl.acm.org/citation.cfm?id=2043468.2043487

[72] S. DiBenedetto, P. Gasti, G. Tsudik, and E. Uzun, "Andana: Anonymous named data networking application," *CoRR*, vol. abs/1112.2205, 2011. [Online]. Available: http://dblp.uni-trier.de/db/journals/corr/corr1112.html#abs-1112-2205

[73] T. Lauinger, "Security & Scalability of Content-Centric Networking," Master's thesis, TU Darmstadt, 2010. [Online]. Available: http://tubiblio.ulb.tu-darmstadt.de/46912/

[74] D. Smetters and V. Jacobson, "Securing Network Content," PARC, Tech. Rep., October 2009.

[75] N. Cristianini and J. Shawe-Taylor, *An introduction to support Vector Machines: and other kernel-based learning methods*. New York, USA: Cambridge University Press, 2000.

[76] L. Wang, Ed., *Support Vector Machines: Theory and Applications*, ser. Studies in Fuzziness and Soft Computing. Springer, 2005, vol. 177.

[77] R. Debnath, N. Takahide, and H. Takahashi, "A decision based one-against-one method for multi-class support vector machine," *Pattern Anal. Appl.*, vol. 7, no. 2, pp. 164–175, 2004.

[78] "CCNDSMOKETEST Manual Page." [Online]. Available: http://www.ccnx.org/releases/latest/doc/manpages/ccndsmoketest.1.html

[79] F. Avolio, "Firewalls and Internet security, the second hundred (Internet) years," *The Internet Protocol Journal*, vol. 2, no. 2, 1999.

[80] A. X. Liu, "A model of stateful firewalls and its properties," in *Proceedings of the 2005 International Conference on Dependable Systems and Networks*. IEEE Computer Society, 2005.

[81] E. S. Al-Shaer and H. H. Hamed, "Discovery of policy anomalies in distributed firewalls," in *INFOCOM*, 2004.

[82] E. Al-Shaer, H. Hamed, R. Boutaba, and M. Hasan, "Conflict classification and analysis of distributed firewall policies," *IEEE Journal on Selected Areas in Communications*, vol. 23, no. 10, pp. 2069–2084, 2005.

[83] H. Hamed, A. El-Atawy, and E. Al-Shaer, "On dynamic optimization of packet matching in high-speed firewalls," *IEEE Journal on Selected Areas in Communications*, vol. 24, no. 10, pp. 1817–1830, 2006.

[84] M. G. Gouda and A. X. Liu, "Structured firewall design," *Comput. Netw.*, vol. 51, no. 4, pp. 1106–1120, Mar. 2007.

[85] R. Marmorstein and P. Kearns, "A tool for automated iptables firewall analysis," in *Proceedings of the annual conference on USENIX Annual Technical Conference*, ser. ATEC '05. USENIX Association, 2005, pp. 44–44.

[86] "The netfilter.org iptables project." [Online]. Available: http://www.netfilter.org/projects/iptables/index.html

[87] A. Lahmadi and O. Festor, "Secsip: a stateful firewall for sip-based networks," in *Proceedings of the 11th IFIP/IEEE international conference on Symposium on Integrated Network Management*, ser. IM'09, 2009, pp. 172–179.

[88] R. Bebawy, H. Sabry, S. El-Kassas, Y. Hanna, and Y. Youssef, "Nedgty: Web services firewall," in *Proceedings of the IEEE International Conference on Web Services*, ser. ICWS '05, 2005.

[89] S. M. Bellovin, "Distributed firewalls," *Login magazine, special issue on security*, vol. 24, no. 5, pp. 37–39, Nov 1999.

[90] S. Ioannidis, A. D. Keromytis, S. M. Bellovin, and J. M. Smith, "Implementing a distributed firewall," in *Proceedings of the 7th ACM conference on Computer and communications security (CCS)*. New York, NY, USA: ACM Press, 2000, pp. 190–199. [Online]. Available: http://dx.doi.org/10.1145/352600.353052

[91] R. N. Smith and S. Bhattacharya, "A protocol and simulation for distributed communicating firewalls," in *Proceedings of Computer Software and Applications Conference (COMPSAC)*, 1999, pp. 74–79.

[92] R. Janakiraman, M. Waldvogel, and Q. Zhang, "Indra: A peer-to-peer approach to network intrusion detection and prevention," in *Proceedings of IEEE WETICE*, Jun. 2003, pp. 226–231.

[93] J. Francois, I. Aib, and R. Boutaba, "Firecol: A collaborative protection network for the detection of flooding ddos attacks," *Networking, IEEE/ACM Transactions on*, no. 99, p. 1, 2012. [Online]. Available: http://dx.doi.org/10.1109/TNET.2012.2194508

[94] D. H. Crocker and P. Overell, "Augmented bnf for syntax specifications: Abnf," Internet RFC 4234, 2005.

[95] T. Segaran and J. Hammerbacher, *Beautiful Data: The Stories Behind Elegant Data Solutions*. O'Reilly Media, 2009, ch. 14.

[96] P. Kolb, "DISCO: A Multilingual Database of Distributionally Similar Words," in *KONVENS 2008 – Ergänzungsband: Textressourcen und lexikalisches Wissen*, 2008.

[97] S. van den Elzen, D. H. Jorik Blaas, J.-K. Buenen, J. J. van Wijk, R. Spousta, A. Miao, S. Sala, and S. Chan, "Exploration and Analysis of Massive Mobile Phone Data: A Layered Visual Analytics approach," in *NetMob*, 2013.

[98] R. Trestian, F. Zaman, and G.-M. Muntean, "Spotted: Connecting People, Locations and Real-World Events in a Cellular Network," in *NetMob*, 2013.

[99] J. ao Pedro Craveiro, F. M. V. Ramos, E. Kanjo, and N. E. Mawass, "Towards an early warning system: the effect of weather on mobile phone usage A case study in Abidjan," in *NetMob*, 2013.

[100] M. Cerinšek, J. Bodlaj, and V. Batagelj, "Symbolic clustering of users and antennae," in *NetMob*, 2013.

[101] J. Smith, J. Stevens, and M. Y. Idris, "NVizABLE: A Web-Based Network Visualization Interface," in *NetMob*, 2013.

[102] Z. Huang and U. Kumar, "Combining call records and road data for strategic disaster response planning," in *NetMob*, 2013.

[103] G. Krings, F. Calabrese, C. Ratti, and V. D. Blondel, "Urban Gravity: A Model For Intercity Telecommunication Flows," *Journal Of Statistical Mechanics: Theory And Experiment*, vol. 2009, 2009. [Online]. Available: http://dx.doi.org/10.1088/1742-5468/2009/07/L07003

[104] V. A. Traag, A. Browet, F. Calabrese, and F. Morlot, "Social Event Detection in Massive Mobile Phone Data Using Probabilistic Location Inference," in *Privacy, Security, Risk and Trust (PASSAT), 2011 IEEE Third International Conference on Social Computing (SocialCom)*, 2011.

[105] L. Page, S. Brin, R. Motwani, and T. Winograd, "The pagerank citation ranking: Bringing order to the web," 1999.

[106] T. Haveliwala, "Efficient computation of pagerank," Stanford University, Tech. Rep., 1999.

[107] S. Brin and L. Page, "The Anatomy of a Large-scale Hypertextual Web Search Engine," in *Proceedings of the Seventh International Conference on World Wide Web 7*, ser. WWW7. Amsterdam, The Netherlands, The Netherlands: Elsevier Science Publishers B. V., 1998, pp. 107–117. [Online]. Available: http://dl.acm.org/citation.cfm?id=297805.297827

[108] C. T. Chu, S. K. Kim, Y. A. Lin, Y. Yu, G. R. Bradski, A. Y. Ng, and K. Olukotun, "Map-Reduce for Machine Learning on Multicore," in *NIPS*, B. Schölkopf, J. C. Platt, and T. Hoffman, Eds. MIT Press, 2006, pp. 281–288.

[109] J. Dean and S. Ghemawat, "Mapreduce: simplified data processing on large clusters," *Commun. ACM*, vol. 51, no. 1, pp. 107–113, Jan. 2008. [Online]. Available: http://doi.acm.org/10.1145/1327452.1327492

[110] J. Leibiusky, G. Eisbruch, and D. Simonassi, *Getting Started with Storm*. O'Reilly Media, 2012.

[111] [Online]. Available: http://storm-project.net

[112] F. D'Elia, G. Di Stasi, S. Avallone, and R. Canonico, "Bittorrent traffic optimization in wireless mesh networks with ALTO service," in *World of Wireless, Mobile and Multimedia Networks (WoWMoM), 2011 IEEE International Symposium on a*, June 2011, pp. 1–6.

[113] K. Kokten, I. Kirkgul, and M. Sunay, "P2p file sharing application with ALTO server," in *Signal Processing and Communications Applications Conference (SIU), 2012 20th*, April 2012, pp. 1–4.

[114] M. Arumaithurai, J. Seedorf, G. Paragliela, M. Pilarski, and S. Niccolini, "Evaluation of ALTO-enhanced request routing for cdn interconnection," in *Communications (ICC), 2013 IEEE International Conference on*, June 2013, pp. 3519–3524.

[115] M. Scharf, V. Gurbani, T. Voith, M. Stein, W. Roome, G. Soprovich, and V. Hilt, "Dynamic vpn optimization by ALTO guidance," in *Software Defined Networks (EWSDN), 2013 Second European Workshop on*, Oct 2013, pp. 13–18.

[116] J. Jacobs and B. Rudis, *Data driven security: Analysis, visualization and dashboards*. John Wiley and Sons, 2014.

[117] W. F. Sharpe, "Mutual Fund Performance," *The Journal of Business*, vol. 39, no. 1, pp. 119–138, 1966. [Online]. Available: http://dx.doi.org/10.2307/2351741

[118] D. Goergen, V. Mendiratta, R. State, and T. Engel, "Identifying abnormal patterns in cellular communication flows," in *Proceedings of Principles, Systems and Applications on IP Telecommunications*, ser. IPTComm '13. New York, NY, USA: ACM, 2013, pp. 5:1–5:6. [Online]. Available: http://doi.acm.org/10.1145/2554666.2554671

[119] D. Goergen, V. Gurbani, and R. State, "Aggregating large-scale measurements for application layer traffic optimization (alto) protocol," IETF Internet-Draft, July 2013. [Online]. Available: http://tools.ietf.org/html/draft-goergen-lmap-fcc-00

[120] ——, "Of maps and costs: Aggregating large-scale broadband measurements for the application layer traffic optimization (alto) protocol," Presented at IIT RTC Conference, October 2013.

[121] D. Goergen, T. Cholez, J. François, and T. Engel, "Security monitoring for content-centric networking," in *Data Privacy Management and Autonomous Spontaneous Security*, ser. Lecture Notes in Computer Science, R. Pietro, J. Herranz, E. Damiani, and R. State, Eds. Springer Berlin Heidelberg, 2013, vol. 7731, pp. 274–286. [Online]. Available: http://dx.doi.org/10.1007/978-3-642-35890-6_20

[122] ——, "A semantic firewall for content-centric networking," in *Integrated Network Management (IM 2013), 2013 IFIP/IEEE International Symposium on*, May 2013, pp. 478–484. [Online]. Available: http://ieeexplore.ieee.org/xpl/freeabs_all.jsp?arnumber=6573021

[123] K. Wesolowski, *Mobile Communication Systems*, J. W. . Sons, Ed. John Wiley & Sons, 2002.