Självständigt arbete på avancerad nivå

Independent degree project – second cycle

Computer engineering

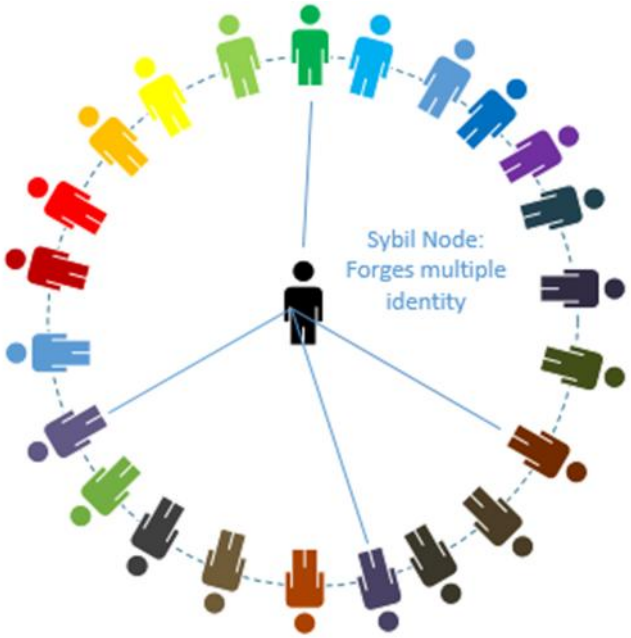
Attacks on structured P2P overlay networks
Simulating Sybil Attacks

Mismaku Tefera

Mittuniversitetet
MID SWEDEN UNIVERSITY

Campus Härnösand Universitetsbacken 1, SE-871 88. **Campus Sundsvall** Holmgatan 10, SE-851 70 Sundsvall.
Campus Östersund Kunskapens väg 8, SE-831 25 Östersund.
Phone: +46 (0)771 97 50 00, Fax: +46 (0)771 97 50 01.

Mid Sweden University
The Department of Information Technology and Media (ITM)
Author: Mismaku Tefera
E-mail address: mite0901@student.miun.se
Study programme: Computer engineering, 120 credit points
Examiner: Prof. Tingting Zehang ,ting.zehang@miun.se
Tutors: Forsström Stefan, Miun, Stefan.Forsstrom@miun.se
Victor Kardeby ,Miun, victor.kardeby@miun.se
Scope: 11400 words exclusive of appendices
Date: 2014-01-25



M.Sc. Thesis
within Computer Engineering , 30 credit points

Attacks on structured P2P overlay network

Simulating Sybil attacks

Mismaku Tefera

Abstract

Structured peer-to-peer network overlays emerged as an alternative mechanism in distributed systems in order to tackle the challenges encountered in the familiar unstructured peer-to-peer architectures and, at the same time, to harness the potential of fully peer-to-peer architectures. Structured peer-to-peer overlays are informally known as Distributed Hash Tables. These provide an efficient mechanism for both searching and for look up. Distributed Hash Tables have been utilized in the most commonly accepted structured peer-to-peer overlays, for example, Chord, Pastry, Tapestry, and P-Grid. Following the widespread use of the structured peer-to-peer overlays, some parties have derived a mechanism to take advantage of the weakness of the peer-to-peer overlays and impair the intended function. In this regard, the most common and efficient attacks have been the Sybil attack, Eclipse attack, Denial of Service Attack and Look up attack. Therefore, the objective of this Master's thesis has been to study, evaluate and simulate a Sybil attack. OMNeT++ and OverSim have been used for the simulation and to study the behaviour of the Sybil attack.

Keywords: P2P, DHT, Chord, CAN, Pastry, Tapestry, P-Grid and MediaSense, Sybil attack, DDos attack, OMNeT++, OverSim.

Acknowledgements

I would like to thank my friends and classmates in Sundsvall, who participated in this thesis by providing me with feedback, suggestions, advice and the necessary resources. However, I would like to pass offer warm gratitude to my tutor, Stefan Forsström, PhD Student at Mid Sweden University for providing me with both important tools and guidance throughout the period of this thesis work. I would also like to thank my second supervisor Victor Kardbey, a PhD Student at Mid Sweden University for providing me with both the guidance and confidence to begin my thesis.

Table of Contents

1 Introduction

Peer-to-peer technologies are becoming increasingly important within the area of file sharing, distributed social network, multiplayer online games, media streaming applications and distributed online storage. Moreover, peer-to-peer technologies provide a better choice within the area of the Internet-of-things, ubiquitous computing, and Cloud computing. In today's Internet, the majority of the traffic is caused by the peer-to-peer applications.

Peer-to-peer systems provide a high performance distribution when compared to the traditional client-server architecture. Peer-to-peer networking has been evolving from the point that the concept was introduced, approximately two decades ago. Peer-to-peer networks are broadly divided into unstructured (For example: Hybridp2p, pure p2p and centralized p2p) and Structured (DHT) peer-to-peer architectures. The early peer-to-peer overlays were not structured thus resulting in huge costs and inefficiency, based on the growth in the number of nodes. However, structured p2p overlays offer a better performance, even as the number of nodes increases massively. As compared to the traditional client-server topology, peer-to-peer topology offers decentralisation, dynamism, self-organization, fault tolerance and load balancing. However, peer-to-peer systems are exposed to security dangers. Adversaries could break in and disrupt part of the global peer-to-peer network. The security related issues must be taken into consideration i.e. security breaches should be sorted out and properly dealt with each and every route adversaries might take to access private information from structured peer-to-peer systems.

This thesis studies the efficient attacks against structured peer-to-peer overlay networks. Common structured peer-to-peer overlay networks will be chosen for this purpose. In addition, different routing algorithms, that could help to improve security against threats, will be studied.

1.1 Background and problem motivation

Structured peer-to-peer overlays act as a substrate upon which large scale, decentralized applications such as storage, massively multiplayer online games, streaming applications, distributed online social networks and file-sharing applications are built. Structured peer-to-peer overlays can route messages correctly, even in a situation where a majority of peers are down. Therefore, such a network structure is scalable and resilient to faults, dynamics and loads. On the other hand, these overlays are prone to attacks from both insiders and from outsiders.

This is particularly the case in open peer-to-peer networks, where different parties come together with conflicting interests, thus distrust will exist and are therefore very susceptible to attacks. Adversaries could setup a node that might introduce malicious code or, an adversary might delay the message coming from one node to another. Therefore, the problem in this thesis is to evaluate and simulate attacks on prominent structured peer-to-peer overlay networks. To this end, a mechanism will be proposed to prevent such attacks and which will also evaluate the existing prevention mechanisms. Sybil attacks are considered as particularly serious attacks on peer-to-peer overlay networks. Sybil attacks take advantage of the fact that, within peer-to-peer networks, no central authentications exist. Therefore, such an attack impersonates multiple identities at any one time or through time. The simulation of a Sybil attack is crucial in relation to determining the level required in order to make peer-to-peer networks secure against these malicious activities, thus leading to mitigation mechanisms.

1.2 Overall aim

The overall aim of this thesis is to study the commonly accepted existing and emerging structured peer-to-peer overlay networks and to then present efficient attacks against these overlays. In addition, part of the overall aim of this thesis has been to evaluate the underlying algorithms related to each prominent structured peer-to-peer network and to determine the security breach with regards to each protocol. Finally, part of the overall aim has been to simulate an overlay network attack using network simulators.

1.3 Scope

The focus in this thesis has been to study the list of efficient attacks and to simulate one of the cases using network simulators. The author's individual analysis and evaluation will be provided for that particular attack. To this end, the thesis will study the details of underlying algorithms relating to both existing and emerging structured peer-to-peer overlays. In this regard, the specific focus is on distributed hash tables (DHTs).The thesis does study and compares different techniques of attack and the corresponding prevention mechanisms in addition to the security requirements.

1.4 Concrete and verifiable goals

The concrete goals of this thesis are the following four core points:

The first goal of this thesis is to determine efficient attacks on structured peer-to-peer overlay networks and what the security breaches are which provide the favourable route for the attack.

The second goal is to simulate the attack on network simulators.

The third goal is to evaluate the impact of these attacks on the performance of the overlay network.

The fourth goal is to evaluate the currently available mitigation mechanisms against these attacks.

1.5 Outline

The thesis has been organized into six chapters.

Chapter one presents a brief introduction, problem statement and goals.

Chapter two describes the widely used structured peer-to-peer overlay networks, security issues and threats in relation to structured peer-to-peer overlay networks.

Chapter three briefly describes the methodology and the approach followed. The chapter explains the strategy followed to answer the research question stated in the chapter one.

Chapter four presents the implementation section. In this section a description regarding how the existing Chord Libraries have been used to simulate a Sybil attack and a comparison of these two simulations.

Chapter five presents the results section. In this section the behaviors of a Sybil attack have been presented and illustrated.

Chapter six provides the conclusions and some future works.

2 Theory

Cyberspace has witnessed a huge shift with regards to the number of peer-to-peer applications, the data traffic size and the innovative technologies related to the distributed systems during the last decade. Peer-to-peer systems offer the freedom continuously being sought by humans. The anonymity of the nodes participating in file sharing offers freedom for a number of counterfeiters and others. Compared to the traditional architecture the advantage of peer-to-peer systems is immense. However, the security risk posed on peer-to-peer systems is a significant challenge. In this chapter different peer-to-peer overlay technologies, the security risks and the technologies to be used to simulate the security threat, stated in the first chapter, will be explained in detail.

2.1 Peer-to-peer system

A decade has passed since the Internet Napster emerged as a distributed technology for online file sharing, though not as fully peer-to-peer. Since then, a number of protocols and novel applications have been developed and a number of research groups are still actively working in order to exploit the potential of peer-to-peer networks. The advent of distributed hash tables around 2000 has contributed much enthusiasm into the area of peer-to-peer research. Based on the DHT, the pioneering structured peer-to-peer overlay networks, Tapestry and Chord, have been developed.

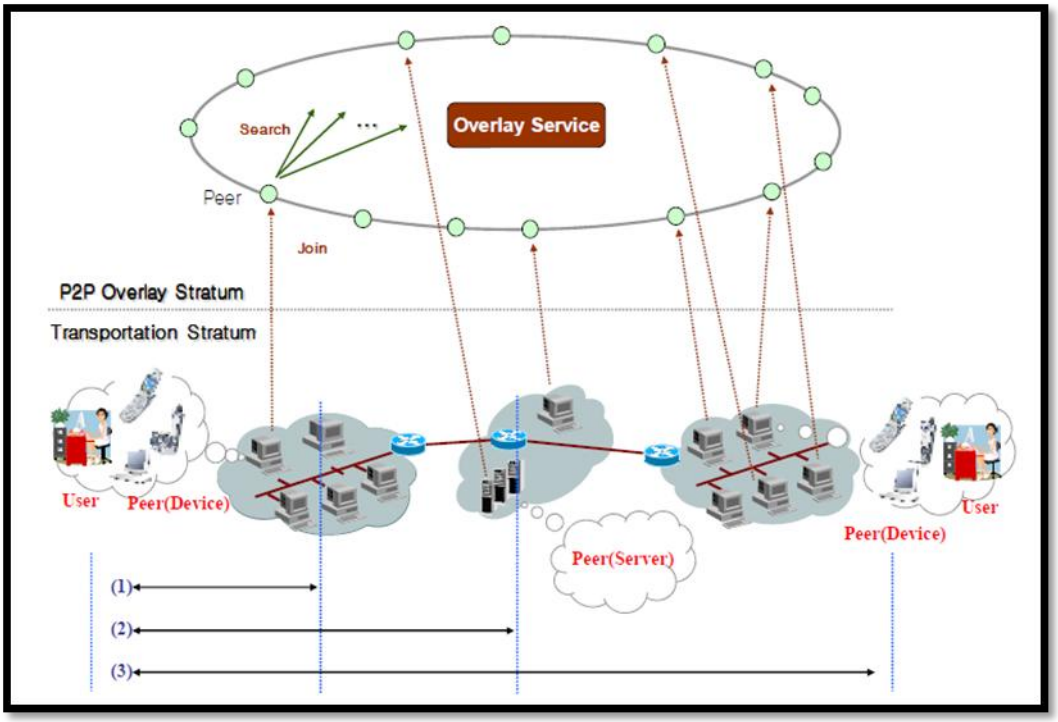


Figure 1 Architectural Reference Model for Peer-to-peer overlay [1]

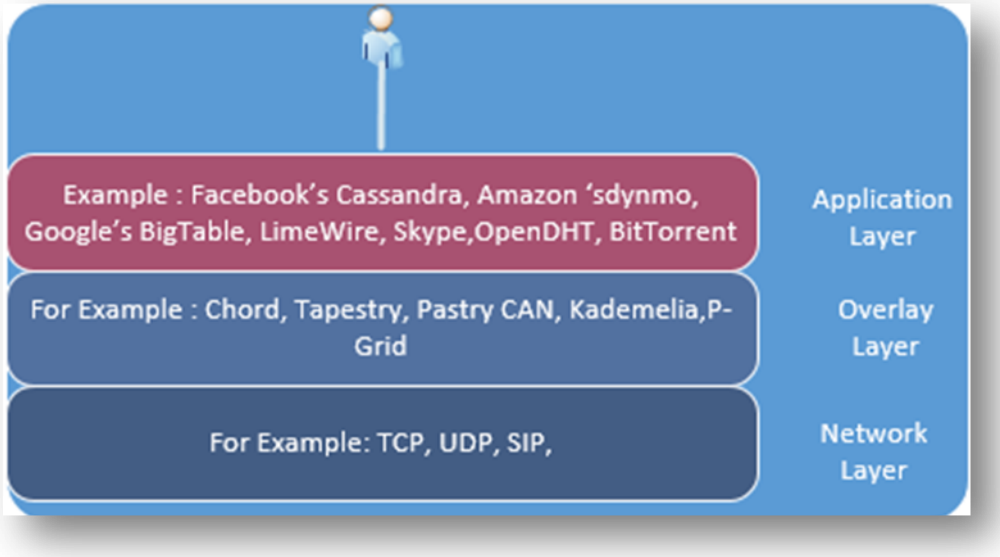


Figure 2 Approximate location of overlay layer in the distributed architecture

"A peer-to-peer overlay is a distributed collection of autonomous end-system computing devices called peers that form a set of interconnections called an overlay to share resources of the peers such that peers have symmetric roles in the overlay for both message routing and resource sharing." [2]

A Collection of participating nodes pursuing the same purpose is called the overlay network or overlay system.

In a communication technology different (clients) parties demand specific requirements or, the preference might be for extra features. However, in relation to security, everybody must be secure. Nobody wants to share private data with the public against his/her will. Despite the popularity of the peer-to-peer paradigm, there is a significant reason for not using peer-to-peer technology in some cases. Security is always a risk in such cases, especially if dealing with open peer-to-peer networks, where anyone can join and leave the network. In this regard, there are inevitable security risks even if different security requirements and prevention mechanisms exist. The security requirements and prevention mechanisms are briefly discussed at a later stage in this chapter.

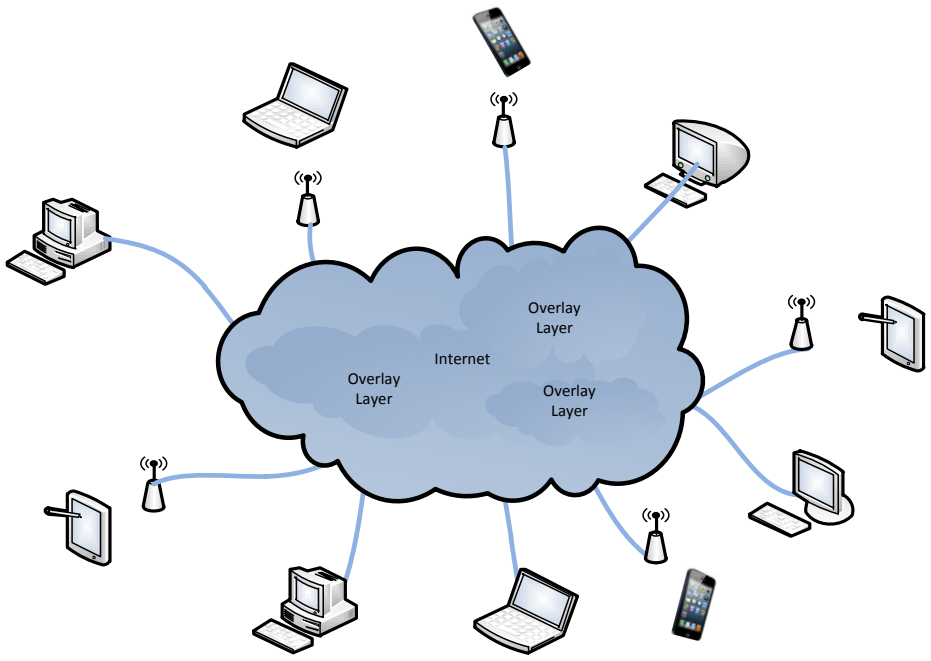


Figure 3 Overlay networks and the Internet

In the early peer-to-peer systems nodes were unable to route in relation to a specific node with a unique ID and Key pair in the network as occurs in a structured peer-to-peer network. In this case, nodes are instead used to search for other nodes, flooding the entire network with a searching data packet. In the present Internet data traffic from peer-to-peer networks, this forms 50% of the total data traffic for both

structured and unstructured peer-to-peer systems, which each contribute their own share [2].

Peer-to-peer systems are broadly categorized as structured, unstructured and hierarchical overlays. Peer-to-peer systems, where the interconnections of peers are not imposed with some kind of structured manner, are known as unstructured overlays. Some of the peer-to-peer systems are pure peer-to-peer while others empower themselves by combining features from the centralized systems and peer-to-peer systems. Unstructured peer-to-peer systems use different algorithms, for example, Flooding, Expanding Ring, Random walk, whereas, the majority of structured peer-to-peer systems use DHT.

2.1.1 Pure peer-to-peer system

Pure peer-to-peer systems are set up with a group of nodes having equal rights and with no different relationship with specific nodes. Any type of super node or some kind of server does not exist in this case. Nodes are interconnected to one another in a peer-to-peer manner, see figure 4 below. The example in this case is Gnutella 0.4.

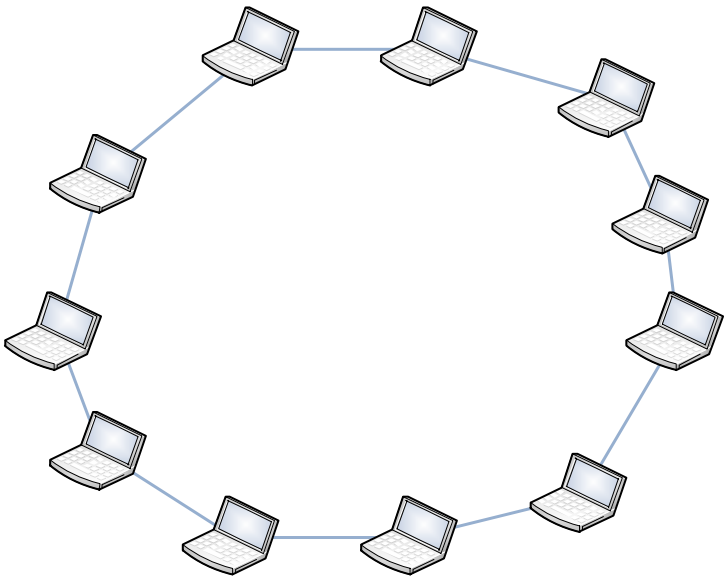


Figure 4 Pure peer-to-peer system

2.1.2 Hybrid peer-peer system

Hybrid peer-to-peer systems are set up on nodes with two different levels of rights, namely, super peers and normal peers. The super peers act as a local central dynamic server for other normal peers. Therefore

normal peers do not directly intercommunicate but, normal peers require to interconnect through super peers. See figure 5 below. Gnutella 0.6 is hybrid peer-to-peer system.

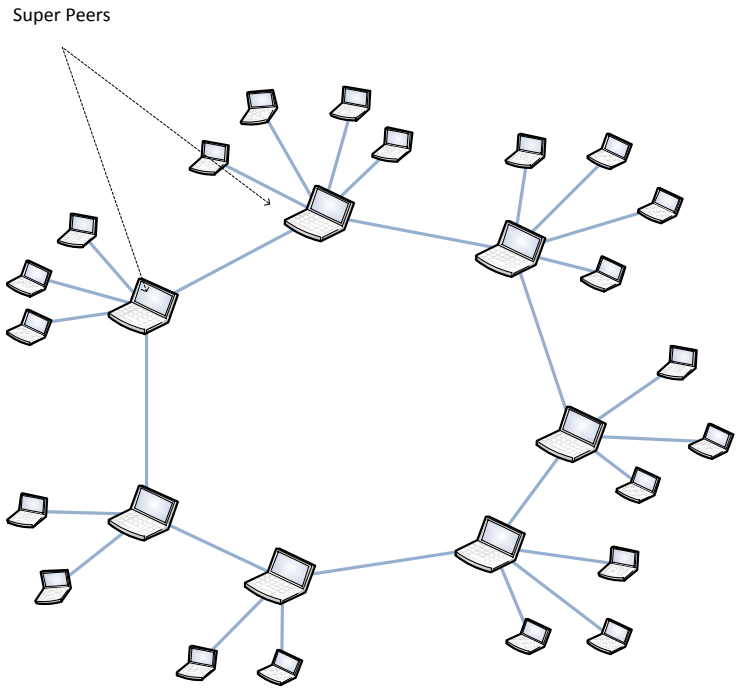


Figure 5 Hybrid peer-to-peer system

2.1.3 Centralized peer-to-peer system

Centralized peer-to-peer systems are different from the above two categories in that such systems rely on a centralized server for some types of resource. In fact, the peers are interconnected in a pure peer-to-peer fashion, however, each peer has to locate the actual storage of the resource that is not within the node but is somewhere outside in a central server. The typical example in this case is Napster.

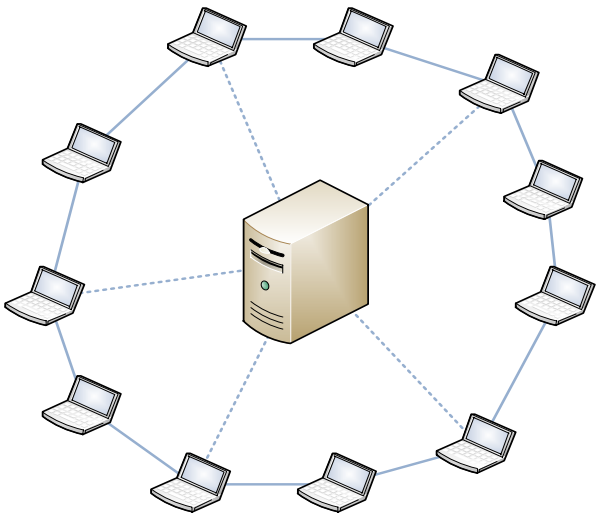


Figure 6 Centralized peer-to-peer system

2.2 Structured peer-to-peer system

Peer-to-peer overlays provide a substrate on which distributed systems are built. Recently, there has been an increase in interest with regards to the emerging structured peer-to-peer overlays as result of an increase in applications in the distributed storage, content distribution and other distributed applications. Moreover, building systems on peer-to-peer overlays offers redundancy, anonymity and fault tolerance. Chord has been made available to the public after the introduction of the first fully structured peer-to-peer overlay. There have been a number of protocols with similar aims and with different incorporated features (Chord, Pastry, Tapestry, P-Grid, and CAN). These protocols are based on the Distributed Hash Tables.

2.2.1 DHT

One of challenges in the realm of peer-to-peer systems has been in relation to routing and discoverability issues while retaining system scalability - known as a *lookup problem*. Peer-to-peer technologies, for example Napster, have dealt with such a challenge by using a centralized server. These technologies, where routing information were stored

in a centralized location, are followed by those groups categorized under the first generation of peer-to-peer systems. Likewise, the so called second generation peer-to-peer systems, for example Gnutella, use a *flooding search*. However, the most scalable technique used by overlay networks, for example, Chord, Pastry, Tapestry, P-Grid is known as the Distributed Hash Table (DHT).

Distributed Hash tables have been considered as a step ahead in relation to designing and deploying highly scalable distributed systems. DHTs are meant to answer the important question "*Where to store node information and how to find the routing information of distributed nodes without using a centralized server?* DHTs provide deterministic routing schemes within a structured search space as compared to unstructured distributed hash tables where routing information is not related to the location of the actual node.

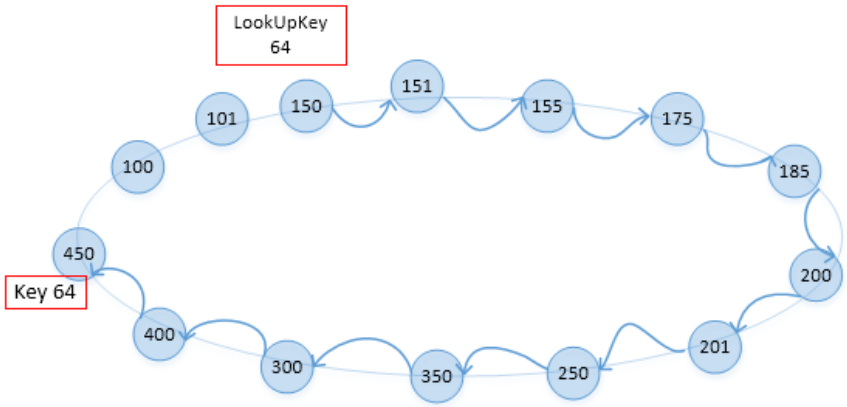


Figure 7 Path taken by node 150 for key 64.

Routing:

Routing is the main functionality of Distributed Hash Tables. Routing information carrying Destination ID's is forwarded to a neighboring node recursively until the destination ID matches the node ID. Message forwarding is based on a kind of metric, for example, the closeness of the neighboring nodes to the target node. See the Figure 8

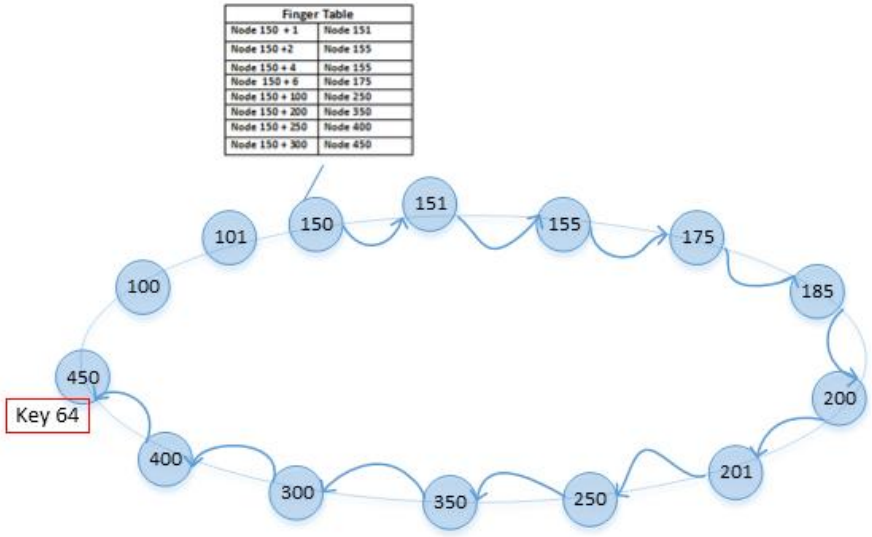


Figure 8 Look Up and Finger table

Data Storage:

Distributed data storage and retrieval is the main purpose of the distributed hash tables. In order to store distributed data DHTs use two mechanisms: Direct Storage and Indirect Storage. In the case of direct storage, nodes copy the entire data into the peer-to-peer system upon insertion. This means that when the node, which owns the data, leaves the peer-to-peer network the data, stays in the system. However, the other mechanism involves the node only carrying reference to the data storage. In the latter case there is a small overhead on the network and bandwidth.

2.2.2 Chord

Chord is a classical structured P2P routing protocol based on DHT, which is proposed by Ion Stoica of the University of California and Robert Morris of MIT et al[3]. Chord has been conceived with the aim of developing a scalable look up protocol in a dynamic peer-to-peer system. Chord maps keys to node/value (a value could be an address, files, or an arbitrary data item) by using consistent hashing. A Chord node does not store information regarding the entire nodes but rather only stores information about a few other nodes. Furthermore, consistent hashing load balances the number of keys on each of the nodes. In an N-node system, each node maintains information of *O(logN)* number

of other nodes in an steady state and resolves look up in *O*(*logN*) messages[4].

Chord can provide a good foundation for different applications, for example, cooperative mirroring, time-shared storage, distributed indexes and large-scale combinatorial searches. A typical application structure is shown in Figure 9.

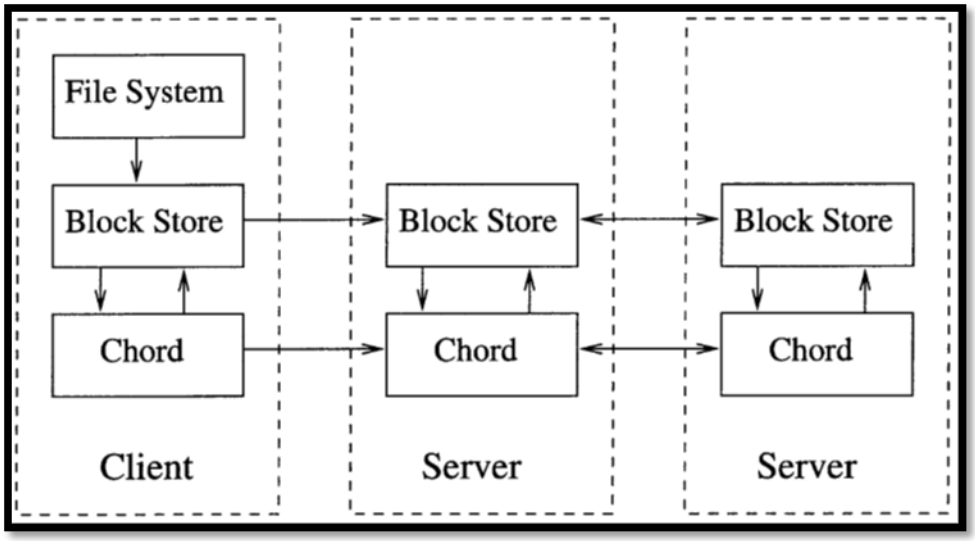


Figure 9 An example for structure of Chord-based distributed storage system [4]

Characteristics of Chord overlay

Load Balance: Chord spreads the number keys evenly over the nodes.

Decentralization: No node is more important than any other node. Thus Chord is a fully distributed protocol.

Scalability: Chord scale in O(log N). Thus, even a network with a large number of nodes offers a feasible look up.

Availability: Chord adjusts the information on each node in a situation where nodes join and leave the network.

Flexible Naming: No constraint is placed on the structure of keys.

2.2.3 Pastry

Pastry is a self-organizing overlay network of nodes, where each node routes client requests and interacts with local instances of one or more applications [5].Pastry is intended as a general substrate for the construction of a variety of peer-to-peer internet applications including global file sharing, file storage, group communication and naming systems [5]Pastry takes network locality into account. It seeks to minimize the distance that messages travel, according to a scalar proximity metric such as the number of IP routing hops. One of the features of Pastry has been its consideration for locality, which means that each node holds proximity information about the neighboring nodes. The proximity parameters could be, for example, the number of routing hops or a geographic distance. Each node is assigned a random 28-bit long node ID. Given a key, Pastry can route to the numerically closest node in a circular node space in *O(logN) messages* where b is a configuration parameter with a typical value of 4. Each of the nodes in a Pastry scheme keeps: a routing table, a neighborhood set and a leaf set.

2.2.4 Tapestry

Tapestry is a peer-to-peer substrate, which provides a high performance, scalable, location-independent routing of messages to close-by end points by using only the localized resources[6].Tapestry is categorized under the group of substrates that utilize the basic key-based routing (CAN, Chord and Pastry). These groups of substrates are also known as second generation peer-to-peer systems. As compared to Chord, Tapestry is locality-aware which means that an optimal routing table is constructed, taking into consideration the location of the nodes.

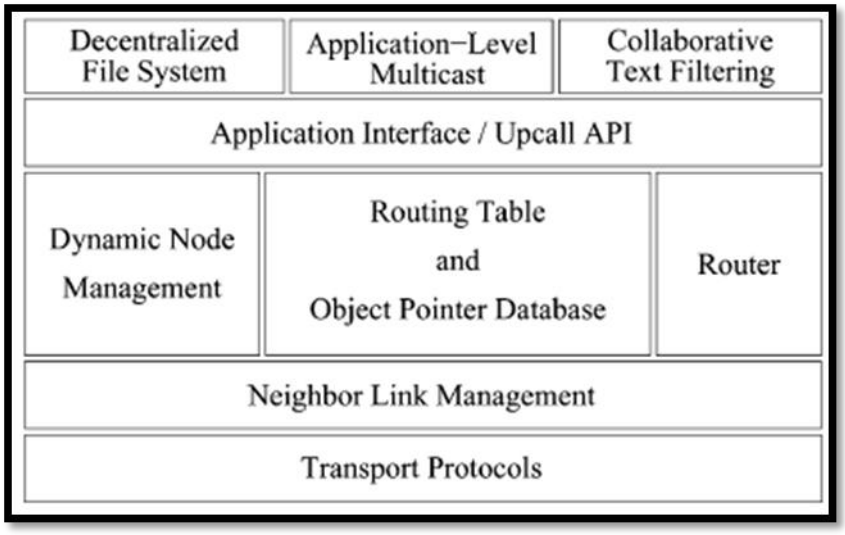


Figure 10 Tapestry Component architecture [6]

2.2.5 P-Grid

P-Grid is a virtual search tree based on a peer-to-peer look up system. Each of the nodes maintains part of the overall tree. Each node is identified by a string of binary bits as seen in Figure 11. The salient feature of a P-Grid, in contrast to the DHT-based P2P systems, is the separation of concern between the peer identifier and the peer's path. In a P-Grid, peer paths are not determined a priori but, are acquired and changed dynamically through negotiation with other peers as part of the network maintenance protocol. Thus P-Grid's prefix-routing infrastructure is constructed by means of a decentralized, self-organizing process in which it adapts to a given distribution of data keys stored by the peers [7].

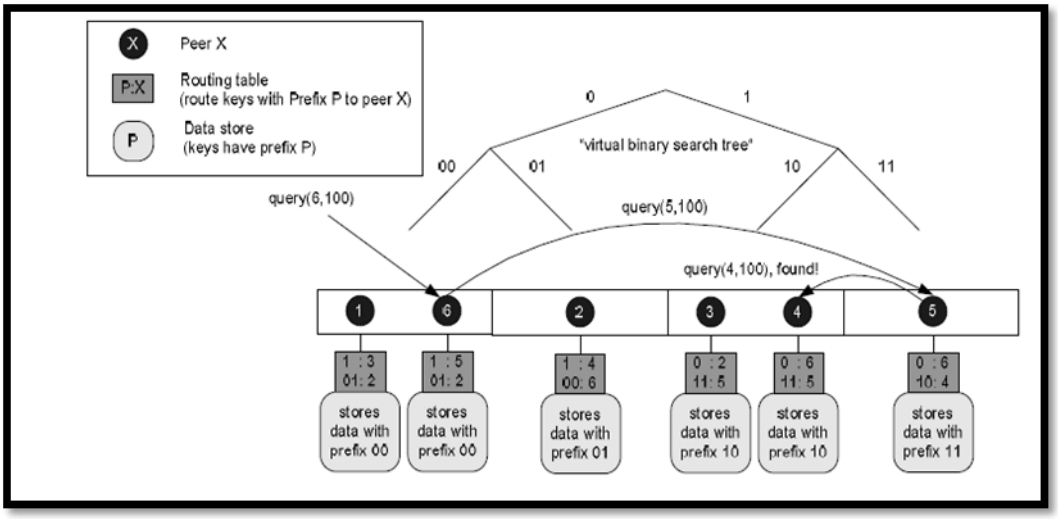


Figure 11 P-Grid Node Search [7]

2.2.6 MediaSense

MediaSense is an Internet-of-things platform for sharing sensor data in a fully distributed manner. MediaSense offers scalable, seamless, real-time access to global sensors and actuators via a heterogeneous network infrastructure [8].The MediaSense platform is a distributed architecture overlay, which enables IoT (Internet of things) applications based on sensor and actuator information [9]. The MediaSense platform has been proposed to interconnect global sensors on a heterogamous network in a peer-to-peer manner and with this platform, scalability, real time data access and seamless integration are given due consideration. The MediaSense platform could be used in a wide range of scenarios including health care, smart home environments, object tacking and social applications [9].

2.3 Major Security problems on structured peer-to-peer overlays

Security problems have been a significant challenge in structured peer-to-peer systems. These issues are a trade off which could not be eliminated as the extreme of pure peer-to-peer systems was being used. This inherent problem arises out of the fact that peer-to-peer systems do not possess a central authentication body. Different mechanisms have been proposed to stop a number of attacks. In the next subsection, the major security problems and security requirements and mechanisms are discussed.

Major Security Problem in a Peer-to-Peer Overlay Networks

- Sybil attack.
- Node ID Attack.
- DoS attack.
- Eclipse Attack
- Message- Forwarding attacks.

2.3.1 Sybil attacks

Sybil nodes compromise the integrity and security of the entire peer-to-peer network by impersonating the identity of other nodes. In fact, among the existing peer-to-peer security threats and attacks, a Sybil attack is considered to be the most challenging and difficult problem to solve [10]. A Sybil node could create a situation in which unfair resources are allotted to it and which can also prevent legitimate nodes from accessing the resources properly. The result of this might be that more resources are available for the attacker thus leading to a greater distraction. Moreover, a Sybil attack opens a breach for other types of attacks.

The primary compromise of a Sybil attack in a P2P network is in relation to destroying the redundancy. A Sybil attack can occur in any network that requires entry and identity mapping, such as P2P networks[11].It destroys the relation of one-to-one mapping of an entity to identity by means of a malicious peer, in other words, a malicious entity acts as a number of multiple identities as shown in figure 12.

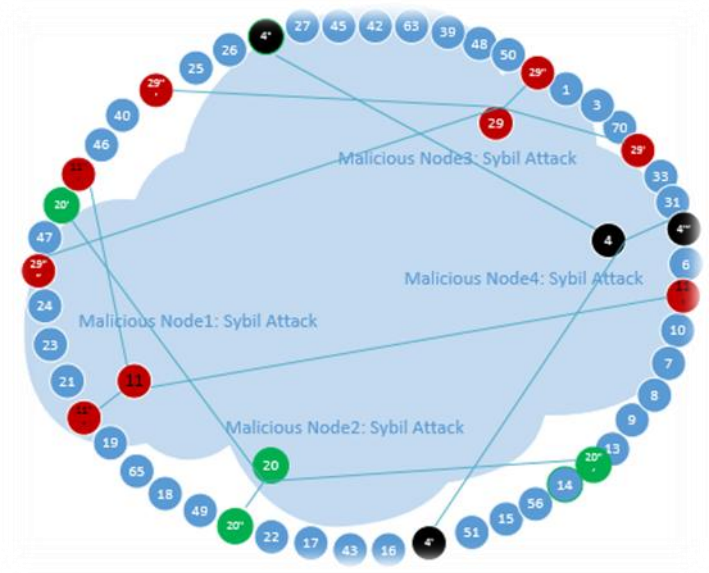


Figure 12 Sybil node forging multiple identities

Sybil attacks could be classified based on the weaknesses and breaches exploited

Direct and Indirect communication:
In direct communication, Sybil nodes directly communicate with a legitimate node. In the indirect communication, legitimate nodes are not allowed to communicate directly but malicious nodes route to the Sybil node.

Fabricated and Stolen identities:
Fabricated identities refer to identities fabricated by the Sybil node whereas; the stolen identities refer to identities stolen from legitimate nodes.

Simultaneous and non-simultaneous:
The attacker might participate by using all its identities simultaneously or the attacker might simulate a situation in which it appears that one node is leaving and another new node is joining.

2.3.2 NodeID /ID mapping attacks

Node ID attack / ID Mapping attack occurs when a malicious node or a group of malicious nodes are able to gain a particular identifier on the overlay network [12]. Obtaining a particular identity will provide an opportunity to gain access to certain parts of the network. The malicious node mediates the access of the victim peer or censors the data traffic.
Three kinds of attacks have been identified under mapping attacks in paper [13].

Complete Attack:
In this case the attacker's aim is to control the target resource completely. In order to control the target resource completely, the attacker impersonates all the closest nodes.

Dictionary Attack:
Dictionary attack is the same as a complete attack but it is not limited to a single resource but, rather, its target is to exploit the entire indexing space.

Partial Attack:
The difference between a complete attack and a partial attack is that, in the case of partial attack the attack does not impersonate all the closest nodes but works on a fraction of the closest nodes.

2.3.3 Denial of Service attacks

Denial of service attack is one of the serious problems which is not easy to detect and to mitigate for. The aim of a DoS attack is to stop the normal operation by bombarding the resources with a bogus amount of data. A DoS attack compromises the availability of data. This attack is not imposed on data integrity or confidentiality. A Distributed Denial of service is a kind of DoS attack in which a volume of malicious traffic arises from distributed devices known as Zombies. A DoS attack takes different forms: Normal DoS attack and Distributed DoS attack (also known as DDoS attack).A normal DoS attack originates from a single host or from a small number of hosts and the only threat they might impose is to exploit some software or a design flaw [14].DDos attacks, on the other hand, are usually generated by a very large number of hosts. These hosts might be amplifiers or reflectors of some kind, or might even be "zombies" (agent program, which connects back to a pre-

defined master host) which have been planted on remote hosts and have been waiting for the command to "attack" a victim [14].

2.3.4 Eclipse attacks

Eclipse attack is a coordinated attack by malicious peers in which a modest number of malicious nodes conspire to fool the legitimate nodes into adopting the malicious nodes as their peers and whose goal is to dominate the neighboring legitimate nodes [22].The final target of such an attack is to eclipse the legitimate nodes from being able to communicate correctly among themselves. An Eclipse attack relates closely to the Sybil attack if those malicious nodes impersonating multiple identities are considered as colluding nodes, however, in a situation where such multiple identities are completely stopped, an Eclipse attack still might occur through a single legitimate peer, which carries malicious intent.

2.3.5 Attacks on Data Forwarding

Under normal circumstances, a message intended to be delivered to another node will be delivered correctly even in a situation where a large proportion of the overlay has crashed. However, any node carrying malicious intent could hinder correct data forwarding in a structured peer-to-peer overlay network.

2.4 Security Requirement and strategy

A strong security policy is considered as being a key to maintaining security of the overlays. It has been discussed in the attacks above that it is not possible to stop a few types of malicious activities, but, the magnitude of the attack can be minimized. A number of tools and technologies could be used and implemented, for example, encryption, authentication, DoS countermeasures and a secured node ID assignment.

Encryption

Encryption is a cryptographic means that provides the most effective solution in relation to security issues. As defined in [16]*Encryption* is the transformation of data into a form that is as close to impossible to read without the appropriate knowledge. It serves as a primary mechanism to implement the confidentiality of data. Encryption generally requires the use of a secret key to encrypt the information and to retrieve back the hidden information at a later stage. Two basic types of encryption techniques exist: symmetric-key or private-key and public-key encryption. Symmetric-key refers to a kind of encryption during which the same key is used at both ends. However, the public-key encryption techniques use different encryption and decryption keys. Decryption keys are only available for the authorized body.

Secured Messaging

In peer-to-peer overlay networks, the message receiving peer is expected to trust the other peer and, additionally, the receiving peer has to be sufficiently honest not to modify the message on its way to its destination. However, such a trust does not guarantee secured messaging. Therefore, there should be a secured messaging mechanism. One technique involves every message being sent to its destination through multiple routes and by this means, at least one copy of the message will be forwarded to its destination with a high probability. The other mechanism to ensure secure messaging is by means of message authentication in which, if an authenticity of a message fails, then another copy will be forwarded to its destination by a different route.

Secured routing table Maintenance

Secure routing table maintenance ensures that the fraction of faulty nodes that appear in the routing tables of correct nodes does not exceed, on average, the fraction of faulty nodes in the entire overlay [17].

Secured assignment of node identifiers

One of the favourable situations that would assist the adversaries in performing malicious activities is the case when nodes joining the overlay are allowed to choose their own node identifier. In such a case, adversaries might take an appropriate node identifier in order to carry out attacks. Therefore, secured assignment of node refers to the assignment of a node identifier chosen by another body and not by the node itself.

2.5 Mitigation Mechanism against Sybil attacks

As explained in the previous sections, the security of a DTH based peer-to-peer system faces a significant challenge when a Sybil attack is carried out by adversaries. The paper by Douceur [18] states that DHT based fully distributed systems cannot be made Sybil free. Some argue that a Sybil attack can only be eliminated if a few trusted nodes exist inside an overlay network and, in effect, such an overlay network is not able to be considered as a fully distributed system. However, it is still possible to reduce the magnitude of the attacks while still enjoying the distributed features of the network. It would not have been possible to forge multiple identities in order to compromise the security of the overlay network if a mechanism to control assignment of node identities existed. The issue that needs to be taken into consideration involves how to manage the assignment of unique node identification only to genuine nodes.

2.5.1 Resource Testing

The resource testing mechanism, proposed by Douceur [18] has been viewed as a mechanism to prevent a Sybil attack. In resource testing, it is assumed that each node or physical entity owns physically limited resources. The resources might be of computation, communication and storage devices.

2.5.2 Computational Puzzle

In the computational puzzles, the verifier sends a large random identity to every entity it would like to verify. The entities are then expected to send a solution within the given time interval. The idea is that if the entity owns multiple identities it will fail to send the solution within the given period of time.

2.5.3 Self-Registration

The self-registration mitigation mechanism received inspiration from the concept of 'Self-contained' in P=Grid in which the idea is proposed to prevent problems arising from using a DHCP (Dynamic Host Configuration Protocol) based IP assignment. The basic idea, in this case, is to properly manage the node identification so that only legitimate nodes obtain a unique ID.

The Node Identification process is divided into assignment, verification and limitation as its foundations in relation to preventing a Sybil attack. However, the important stage is the node identification assignment and, depending on the assignment method, verification and limitation stages do exist.

In the self-registration nodes, their identifiers are calculated based on their IP addresses and, additionally, the port of the connection is taken into account. The node then registers its ID in the P2P network at those nodes which have already been successfully registered. The registration is only based on the IP address or parts of the IP address [19]

Centralized identifier assignment:

Identification is assigned by means of a central entity. The central entity might be part of the overlay network. Such a mechanism might prevent a Sybil attack; however, a central point of failure might occur for nodes joining the network.

Distributed assignment with external identifiers:

Identification is derived from external identifiers, for example, an IP address. This assignment could be made to be secure as the external identification and it could be Sybil-proof. The other advantage is that it requires no additional cost as in the centralized identifier assignment, where a cost is required in order to set up external entities, which assign an identifier.

Distributed assignment with free identifiers:

Each node has the right to choose one or more identifiers and there is also a means to make this identifier globally unique. However, there is no mechanism to guarantee that each node has only one unique identifi-

er. Therefore, such an assignment cannot offer a Sybil-proof environment even if techniques, such as crypto puzzle, are utilized.

Distributed group-based identifier assignment:

In this case, a group of nodes asserts the legitimacy of a particular node. As in a distributed assignment with free identifiers, this mechanism does not guarantee a Sybil-proof environment.

2.6 Network Simulators

Although very common network simulators do exist , for example, Narses,NeuroGrid,PeerSim,P2PSim,Overlay weaver ,ns-2 and ns-3, in this thesis the decision has been taken to use OMNeT++ as a framework plus OverSim , for the sake of convenience and for the scope of the thesis.

2.6.1 OMNeT++

OMNeT++ is a framework for developing networking simulations. OMNeT++ is a free tool for network related simulations. OMNeT++ provides all the necessary tools and infrastructures for writing simulators. The basic building components of the infrastructure are in the form of modules. These modules are written using the C++ language. The simulations on OMNeT++ could be run though graphics (animation) or by a command Line Interface. OMNeT++ supports parallel distributed simulations. In detail, the OMNeT++ framework can be used in the following problem domains:

- Modeling Communication networks.
- Modeling protocols.
- Modeling distributed systems.
- Studying the performance of complex software systems.
- Modeling and simulation of any system where the discrete event approach is suitable, and can be conveniently mapped into entities communicating by exchanging messages [20].

2.6.2 OverSim

OverSim is an open-source overlay and peer-to-peer network simulation framework for an OMNeT++ simulation environment [21].OverSim is a flexible overlay network simulation framework for Linux, Windows, Mac OS X and Nokia Internet Tablets (Maemo) based on OMNeT++[15]. The libraries include simulation packages for Chord, Pastry, Bamboo, Koorde and Broose, among others. OverSim supports the simulation of up to 100,000 nodes.

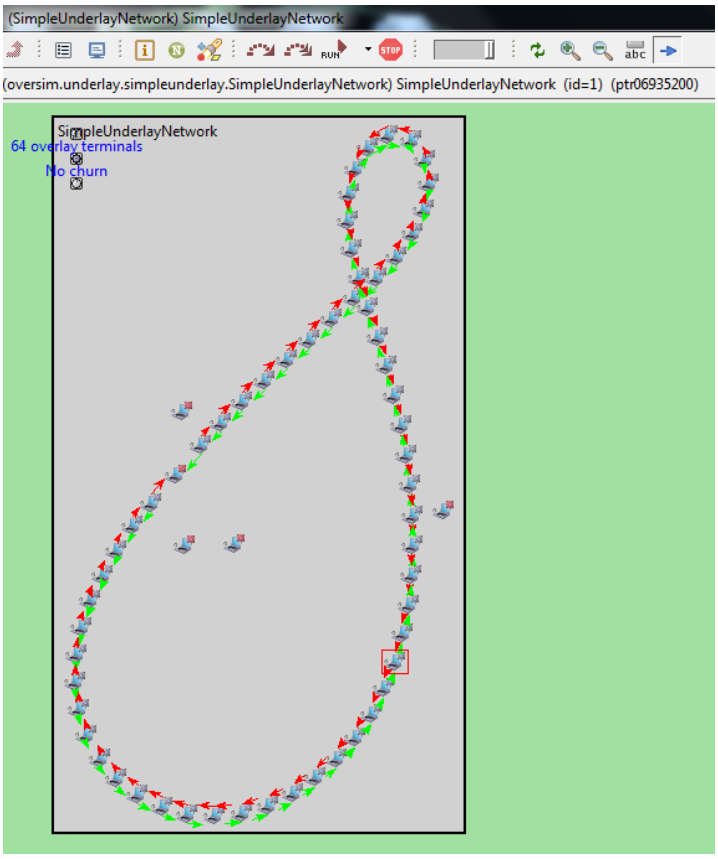


Figure 13 Simulation of peer-to-peer nodes using OverSim on OMNeT++

OverSim offers three models of underlay abstraction layers, which have different levels of complexity and accuracy.

***Simple underlay (Default)*:** Simple underlay provides less computational overhead and high accuracy. Simple underlay is preferable for simulating large overlay networks.

INET: The INET underlay provides a good model to simulate heterogeneous access networks, backbone routers and terminal mobility.

SingleHost underlay: A middleware for supporting a real network developed for OverSim.

Others widely used P2P Simulators:

Narses: Narses is a scalable, discrete event, flow based application-level network simulator. It allows the modelling of the network with different levels of accuracy and speed to efficiently simulate large distributed applications [23].Narses runs a maximum of 600 nodes and it does not allow distributed simulation. In addition, no hint could be found regarding the existence of built-in overlays.

NeuroGrid: NeuroGrid is a peer-to-peer search protocol, originally intended for the simulation of the NeuroGrid protocol, Freenet and Gnutella [24]. NeuroGrid could be used to simulate both structured and unstructured peer-to-peer networks. It does not support Churn simulations.

PeerSim: PeerSim is an event-based p2p simulator written in Java [29]. Using PeerSim, it is possible to simulate structured and unstructured overlays. PeerSim supports predefined protocols: OverStat, SG-1 and T-Man. It does not support distributed simulations.

P2PSim:P2PSim is a free, multi-threaded, discrete event simulator to evaluate, investigate, and explore peer-to-peer (P2P) protocols [26]. P2PSim runs in several UNIX-like operating systems. P2PSim provides six built-in protocols: Chord, Accordion, Koorde, Kelips, Tapestry and Kademelia.

NS2: Ns-2 is a discrete event simulator targeted at networking research [25]. Ns-2 provides substantial support for the simulation of TCP, routing, and multicast protocols over wired and wireless (local and satellite) networks. NS2 is packaged with only one peer-to-peer protocol: Gnutella.

Overlay Weaver: A toolkit for easy development and testing of peer-to-peer protocols [30]. It is possible to simulate structured and unstructured overlays using Overlay weaver. The tool provides built-in proto-

cols: Chord, Kademelia, Koorde, Tapestry and Pastry. It supports distributed simulation; however, the documentation is poor.

PlanetSim: PlanetSim is an object oriented simulation framework for overlay networks and services [27]. This framework presents a layered and modular architecture with well-defined hotspots, documented using classical design patterns. PlanetSim provides built-in Chord-SIGCOMM and Symphony as well as supporting the implementation of structured and unstructured overlays. There are no functionalities to extract statistics.

GPS: GPS is a discrete message level event simulator with a built-in implementation of BitTorrent [28].It allows the simulation of structured and unstructured overlays; however, distributed simulation is not supported.

Advantages of using OverSim in this project

- OverSim is a free and open tool.
- OverSim runs on top of OMNeT++ which is also a free tool and which allows distributed simulation.
- Using OverSim it is possible to simulate the widely accepted DHTs: Bamboo, Broose, **Chord**, Kadmelia, Koorde, pastry and so on. OverSim provides more built-in overlays than any of the overlay simulators mentioned above.
- It has been well designed for the simulation of networks and matches the demand of this project, though not entirely. In the last chapter, an explanation has been provided regarding the difficulties with which to achieve modifications within the code. The OverSim only provides malicious free DHT and thus, no specific attack is implemented, which is the reason for the modifications.
- Provides functionalities to extract statistics.
- OverSim was designed with performance in mind. On a modern desktop PC, a typical Chord network of 10,000 nodes can be simulated in real-time [21].

3 Methodology

The method approaches the problem statement presented in chapter one by dividing it into four parts. Each part of the problem statement has been properly dealt with in order to achieve the goals outlined in the introduction section. Accordingly, the commonly accepted existing and emerging structured peer-to-peer overlay networks, for example, CAN, Chord, Tapestry, Pastry, Kademelia, P-Grid and others have been studied. In addition, there have been discussions regarding the MediaSense overlay network from the point of view of security and related issues. Following this, underlying protocols and algorithms related to each prominent structured peer-to-peer network have been thoroughly studied.

Therefore, the corresponding sections for each of the four goals have been organized as follows: firstly, security threats and attacks on peer-to-peer overlays have been presented. Secondly, simulation of a Sybil attack on one of the common DHT has been conducted. Thirdly, evaluation and analysis of a Sybil attack has been performed and results have been presented, with illustrations.

3.1 Security treats and attacks on structured peer-to-peer overlay

The first goal of this thesis has been achieved by thoroughly studying attacks on peer-to-peer overlays. In this regard, security breaches of the underlying protocols and algorithms related to each prominent structured peer-to-peer networks have been briefly presented. In this part, the common threats and previously successful attacks in peer-to-peer network are discussed; for example, look up attacks, denial of service attacks, attacks on routing table entries, attacks on data forwarding, attacks on placement schemes, forging of multiple identities for malicious intent, and others. In addition, existing defense mechanisms against threats and attacks on structured peer-to-peer networks, in addition to security requirements for such overlays, have been explained.

3.2 Simulation

Different methods could be adopted in order to conduct the simulation of a Sybil attack and study the behaviour of a Sybil node, namely, test beds, real environments, simulation with virtual nodes and simulations with real nodes. Furthermore, each of these techniques has its own advantages and disadvantages. However, considering the time budget and the resources allocated, particularly the cost in terms of money, conducting the entire simulation on a single machine, with virtual nodes, is the preferred method.

The purpose of the simulation of a Sybil attack is to study the behaviour of the attack. This involves examining the number of messages exchanged between nodes and the impact of the Sybil node on the neighbouring nodes and on the entire overlay network.

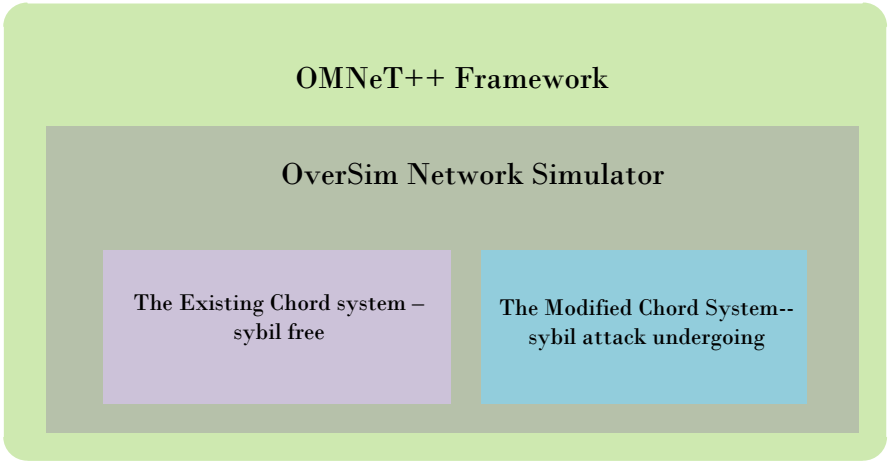


Figure 14 Simulation Tools and Components

3.2.1 Simulation tools, architecture and frame

Concerning the second goal, which aims to simulate a Sybil attack on a network simulator, the simulation of a Sybil attack has been conducted using OMNeT++ and OverSim. To this end, the popular DHT, known as the Chord overlay network, has been used. See figure 14 with regards to the simulation components.

OMNeT++:

OMNeT++ is a framework for developing networking simulations and is a free tool for network related simulations. OMNeT++ pro-

vides all the necessary tools and infrastructures for writing simulators.

The primary factor for using OMNeT++ in this project has been the fact that OverSim runs on OMNeT++. In addition, the GUI of OMNeT++ has been found to be of use for this project. Moreover, the OMNeT++ framework and the OverSim simulation tools are open source and free to use. The advantages of these tools have been explained in detail in the implementation chapter.

Version: omnetpp-4.2.2

OverSim:

OverSim is a flexible overlay network simulation framework for Linux, Windows, Mac OS X and Nokia Internet Tablets. OverSim runs on OMNeT++.

Using OverSim, it is possible to simulate the widely accepted DHTs: Bamboo, Broose, **Chord**, Kadmelia, Koorde, pastry and so on. OverSim provides more built-in overlays than any of the overlay simulators mentioned above.

It has been well designed for the simulation of networks and matches the requirement of this project.

The tool provides functionalities to extract statistics.

Version: OverSim-20121206

Inet:

INET underlay provides a model to simulate heterogeneous access networks, backbone routers and terminal mobility. This is part of OMNeT++ framework.

Version: inet-20111118

Chord:

Chord DHT provides the overlay network on which messages are exchanged. Chord DHT is part of the packages in OverSim framework.

On the top of the existing Chord DHT, Sybil nodes have been introduced.

Simulation Machine:

Specification: Toshiba processor Intel(R) Core(TM) i3-2377M CPU @1.5 1.5GHz, RAM: 4GB

3.2.2 Simulation Scenario

In the real scenario, the number of nodes participating in one overlay network is extremely large (thousands), however, due to the limited computing ability of the resources, this thesis has worked on a total of 500 nodes.

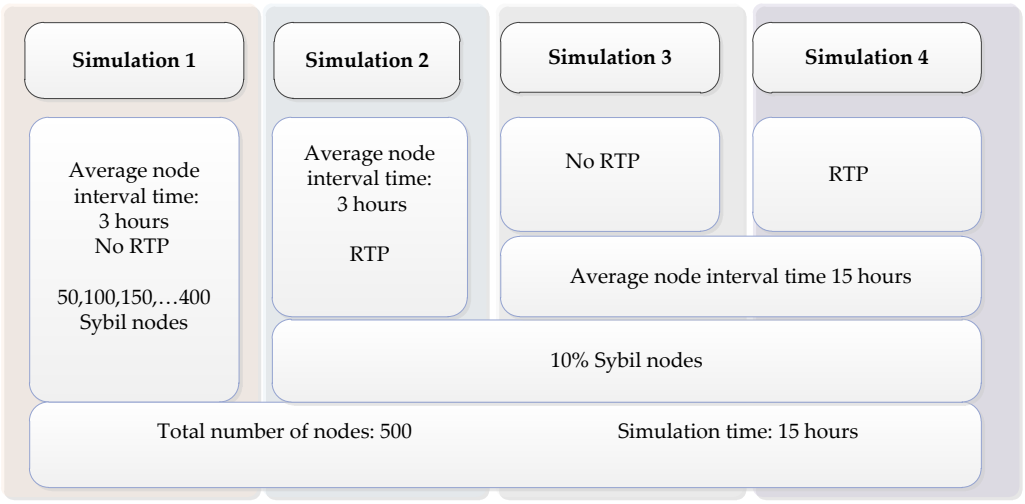


Figure 15 Simulation Scenario: the parameters used in the experimentation to study the behaviour of Sybil nodes and its significance.

In the thesis, four cases have been considered during the experimentation. The main factors, which could expand the influence of a Sybil attack as are known from the literature review, are additional malicious activity such as RTP and the time involved during which each of the nodes is alive. Therefore, in the simulation expanding these cases, four scenarios (see figure 15) have been worked on. The details of the mechanisms for accomplishing these simulations are presented in section 3.3 and the next chapter.

In the documentation of the OverSim framework, it is mentioned that in excess of thousands of nodes could be simulated on a laptop with 2GB

memory. However, it has been determined by the author that for nodes in excess of 500, the performance of the machine is not tolerable.

3.2.3 Experimental data and statistical analysis

OMNeT++ provides important tools necessary to examine the messages exchanged between nodes. The number of packets exchanged between each node, the type of message exchanged, the sources of the messages and related detailed information, could be collected. Therefore, the messages coming from the Sybil nodes and other genuine messages could be identified involving only a minor amount of time and energy, using the tools from OMNeT++ GUI. Details of the tools used for data gathering are presented in chapter 4.

For example: Node X [genuine node]

Run the simulation

With an interval of 1 hour, pause the simulation and examine the routing table of the node and the list of messages exchanged. Count the number of Sybil nodes and genuine nodes listed under node X.

After the details of the messages exchanged between the nodes and a list of statistics from the routing table had been successfully obtained, illustrations were drawn in order to compare the results.

3.3 Evaluation

The third goal has been achieved by studying the behaviour of the Sybil node that has been introduced into the simulation. Therefore, in this stage, an evaluation has been conducted based on the experimental environment set up in section 3.2. In this case, data has been collected regarding the number of messages exchanged and these messages have been examined.

In the evaluation of the Sybil attack, factors which could aggravate the seriousness of the Sybil attack have been considered. In this case, two factors have been taken into consideration. The first factor, is the case with long living Sybil nodes and the second factor is the case when an adversary systematically disseminates forged information about the already running Sybil nodes. Finally, the results have been clearly

illustrated. Furthermore, discussions and conclusions have been drawn regarding the significance of the Sybil attack on a structured peer-to-peer overlay.

3.3.1 Sybil attack with Routing Table Poisoning

Systematic introduction of forged information into the routing tables could be conducted in addition to the Sybil attack in order to foster a Sybil attack .The routing table poisoning (or RTP) attack has been introduced by feeding the routing table with some of the legitimate nodes with information of malicious intent. The aim in this section is to forward the data coming from the legitimate nodes through the Sybil nodes.

In practical cases, in order to gain a higher fraction of the entire network, malicious peers would forward forged information about the Sybil node. Doing so increases the influence of the Sybil attack as compared to a Sybil attack conducted without RTP.

3.3.2 Sybil attacks with Long-Living nodes

The other important factor which might contribute to an increased Sybil influence involves the time the Sybil node are allowed to run within the peer-to-peer network. The longer the Sybil node stays alive within the DHT, the higher the percentage of Sybil influence. Therefore, in this case, the network is run for 15 hours, setting the average node interval time of 15hrs, however, in the first experiment the average node interval time is 3hours.

3.3.3 Sybil attacks with Routing Table Poisoning and with Long-Living nodes

The influence of Sybil attack becomes very high when the two fostering factors happen to exist at the same time. This section of the simulation assists in comparing the previous scenarios with the maximum possible influence from a Sybil attack

3.4 Theoretical analysis of Implemented techniques

The thesis has outlined the goals to simulate one of the particularly harmful attacks known as the Sybil attack. In this section, the theoretical analysis of the actual work conducted, approach techniques and an evaluation of its ethical impact are discussed.

3.4.1 Theoretical project output evaluation

The simulation of a Sybil attack assist in gaining knowledge of factors that create a favorable environment for such harmful malicious activity. Moreover, studying such attacks will also assist in knowing what kind of other adversary activities could aggravate the volume and the seriousness of the attack. In this thesis, in order to simulate Sybil attack, OMNeT++ and OverSim have been chosen. For the sake of convenience and resource issues, this thesis has run the simulation using 500 nodes but, in a real environment, the nodes are in the hundred thousands. However, the belief is that what have been performed using 500 nodes can be extended to any number of nodes. Thus, based on this thesis, it is possible to know what a Sybil attack is, how the attack occurs, factors that make the attack trivial and the mitigation mechanisms that are available to create a Sybil-proof environment.

3.4.2 Approach technique evaluation

The aim, at this point, is to explain the effort and the advancements regarding the research tools employed in the thesis. Effort is a subjective term but it could be expressed in terms of cost, for example, the lines of code written, the number of people participating, the resources required and so on. In addition, the tools and knowledge that are demanded are also important in relation to knowing the standard of the conducted work. Regarding the former, the lines of codes are written in C++ and the number references collected from IEEE, Springer, and different web pages, are the proof. Furthermore, regarding the tools employed, OverSim is an open and free environment for overlay network simulation. OverSim is a recent addition, specifically targeting simulation of Overlay networks and it has an active mailing list as well as strong user base.

3.4.3 Ethical deliberation

The tasks performed within this thesis work have contributed to the research in the area of structured peer-to-peer networks, especially with regard to one of the most harmful attack known as Sybil. Nevertheless, any negative impacts of the results on society have been ruled out as far as the problem statement and the goals are concerned. Furthermore, scientific rules with regards to the referencing of an article have been followed in order to avoid plagiarism.

4 Implementation

Peer-to-peer overlays and attacks on structured peer-to-peer overlays could be simulated using overlay network simulators, for example, OMNeT++, p2psim, PeerSim, PlanetSim, and Overlay Weaver. However, in this section OMNeT++ and OverSim have been used as the simulation environment and the simulation package, respectively. OMNeT++ is used because it is more convenient for this thesis and, in addition, it is possible to run a number of commonly known DHTs by means of OverSim. Moreover, the OMNeT++ framework and the OverSim simulation tools are open source and free to use.

The following are some of the points in relation to favouring OverSim as compared to the other possibilities mentioned in Chapter 2 section 2.6.

- OverSim is a free and open tool.
- OverSim runs on top of OMNeT++, which is also a free tool and which allows a distributed simulation.
- Using OverSim it is possible to simulate the widely accepted DHTs: Bamboo, Broose, **Chord**, Kadmelia, Koorde, pastry and so on. OverSim provides more built-in overlays than any of the overlay simulators mentioned above.
- It has been well designed for the simulation of networks and matches the demand of this project, though not entirely. In the last chapter, an explanation has been given regarding the difficulties of modifications within the code. The OverSim only provides malicious free DHT and so there is no specific attack implemented. Therefore, the modifications are necessary to that end.
- Provides functionalities to extract statistics.
- On a desktop PC, a typical Chord network of 10,000 nodes can be simulated in real-time.

It has been mentioned in chapter three that modifications are performed on the existing simulation of Chord in order to simulate the Sybil attack

and examine the impact of the attack, plus an in-depth examination of the data traffic. See figure 16 below. In this chapter a detailed explanation is presented for each subsection that a modification has been conducted to the existing simulation. The strategy followed in this section has been to introduce 10% of the entire nodes as Sybil nodes and, additionally, routing table poisoning has been introduced. Routing table poisoning could be used to raise the security compromise to a serious level if such an attack is carried out together with a Sybil attack.

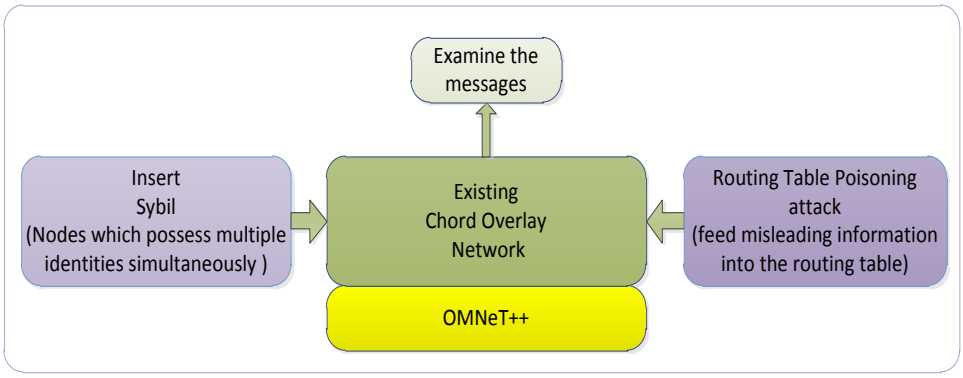


Figure 16 Sybil attack Simulation on OverSim

4.1 Simulator

OverSim is imported onto the OMNeT++ framework. OverSim could be downloaded for free on www.oversim.org. A number of versions are available for windows and Linux platforms. The version chosen for this thesis work is *oversim-20121206* and *OMNeT++ 4.2.2*(March 2012 release) is used. OMNeT++ could be downloaded for free from http://omnetpp.org . In addition, inet-20111118 libraries have been downloaded fromwww.oversim.org. The inet framework provides several applications models and protocols, for example, UDP, TCP, IPv4, IPv6, and SCTP. After following the installation *of OMNeT++4.2.2* and then importing O*verSim* and *inet*, the environment (See figure 17) is ready for the simulation of both supported structured and unstructured overlay networks.

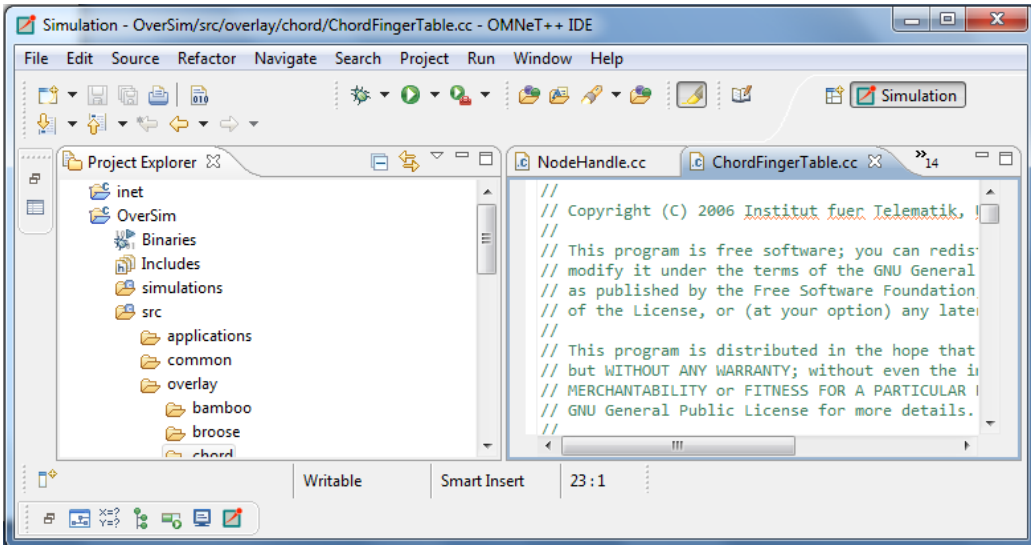


Figure 17 OMNeT++ and OverSim

4.2 Chord DHT

The existing chord simulation tool on OverSim fully supports all the features of the Chord DHT. Moreover, the implementation of the Chord overlay module strictly follows the paper by I.Stoica[1].

4.2.1 Existing Chord System

Multiple simulation options are available, based on the number of nodes and other parameters. For example, Chord, ChordInet, ChordReaSe, ChordSimpleSemi, ChordFastStab and ChordLarge. For this thesis,**ChordLarge** has been chosen in order to work on a maximum of 500 nodes and study the data traffic as the Sybil attacks are progressing. The Chord simulations have been built up by the ChordModules as shown the figure 18 below. The functions of each these ChordModules has been explained here.

[1]Chord: A Scalable Peer-to-Peer Lookup Protocol for Internet Applications" by I. Stoica et al. published in Transactions on Networking

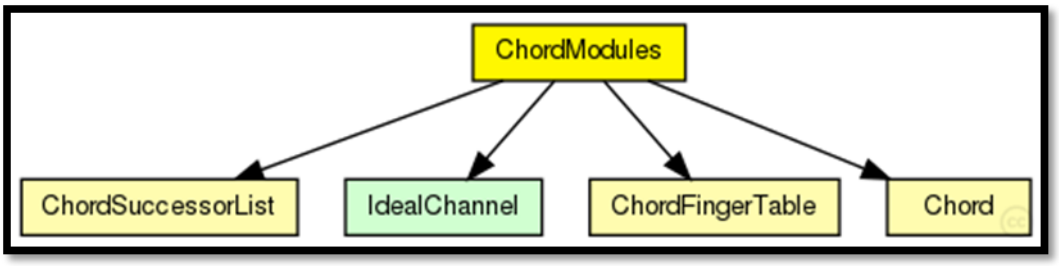


Figure 18 Chord Modules[21]

ChordModules	
ChordSuccessorList	Contains the List of Chord successor implementations.
ChordFingerTable	Contains finger table of the Chord implementation.
Chord	Contains the Chord KBR (Key-based routing) overlay implementation.

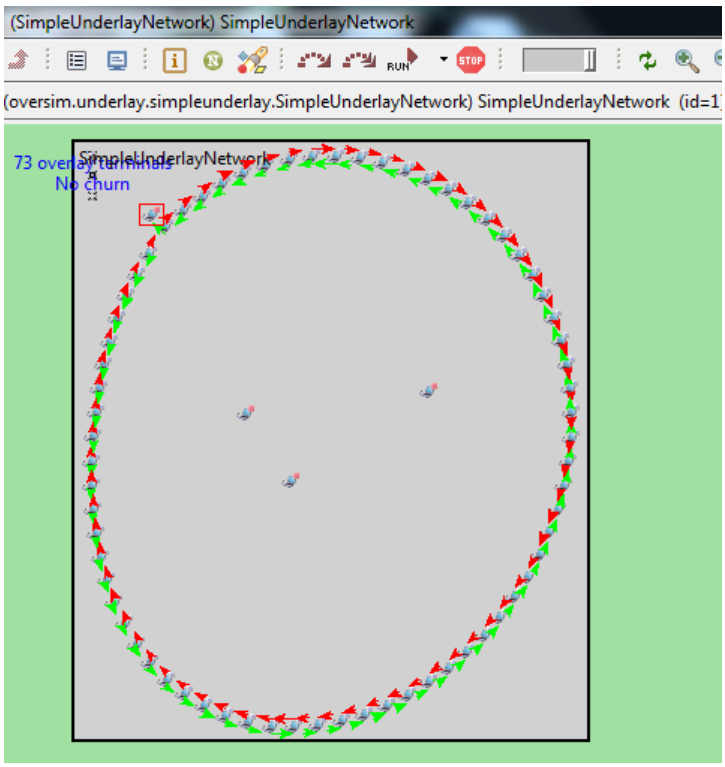


Figure 19 Simulated Chord network

4.2.2 The modified Chord system

In this section, the OverSim Simulation libraries have been modified so that it is possible to introduce Sybil nodes. Moreover, the finger tables have been altered in order to study the impact of a Sybil attack as it occurs together with some other malicious activity that could seriously compromise the security. The following sections of the existing libraries have been altered.

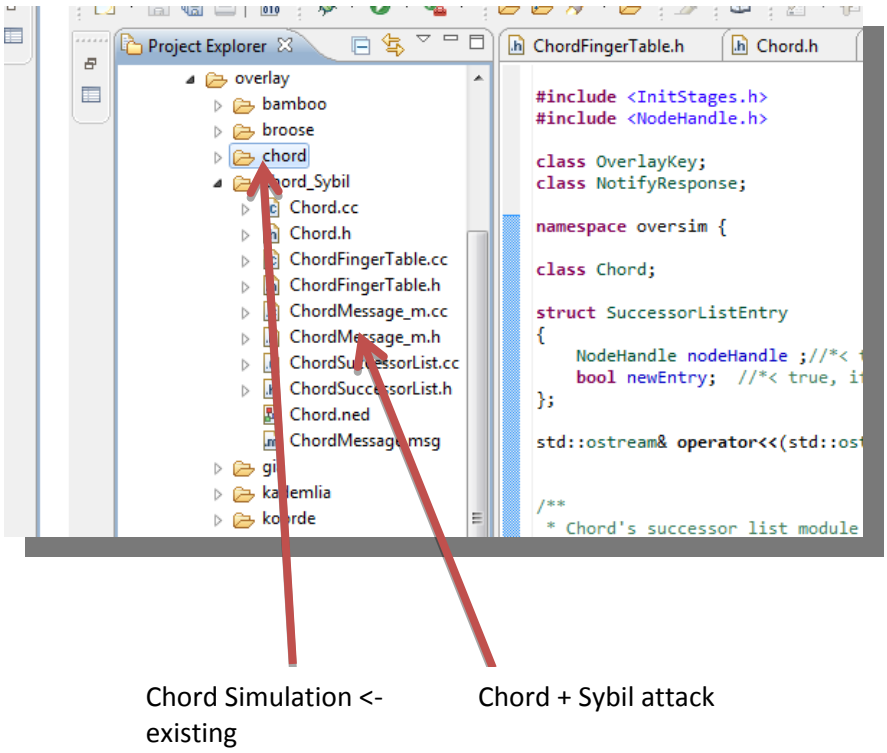


Figure 20 Modifications and Coding on Existing System

ChordFingerTable:
Chord Finger Tables have been modified for the Sybil nodes but for the remainder of the nodes no modifications have been made.

Chord:
Default.ini has been modified in order to set the maximum number of nodes to simulate

ChordSuccessorList:
No Modifications have been made in this section. The number of the successor list remains as eight

4.3 The Sybil node

Chord implementation on the OverSim libraries has been used to execute a Sybil attack. A maximum of 500 peers has been used to simulate the attack and to examine the behaviour of the Sybil attack as well as to study the traffic in a situation where such an attack occurs. Of the 500 nodes, fifty nodes have been introduced as malicious nodes representing

Sybil nodes. In this section, the approach is to introduce 10% of Sybil nodes from the total of 500 participants into the DHT. Thus, a close examination of the routing table entries of each node can be conducted for a period of 15hours. The study aims to determine what fraction of the routing table entries become malicious in an interval of one hour. As can be seen in the figure 21, it is possible for a single node to impersonate representing multiple identities at a time. In chapter two it has been presented that a single node may not necessarily represent multiple identifications at a time, as , in some situations, the Sybil may only possess a single identification and may change its identification with malicious intents. The simulation, intended to be used in this thesis, involves a Sybil node, which represents multiple identities at a time see figure 21.

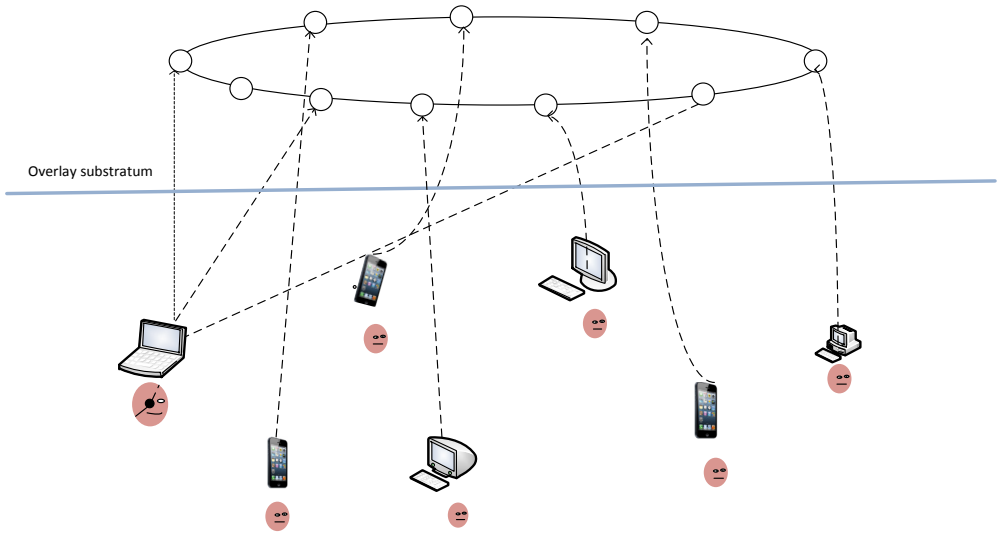


Figure 21 Simultaneous fabricated multiple identity impersonation.

4.3.1 The Sybil Node – the Model

As has been clearly explained in the theory section, a Sybil node possesses multiple identities and thus it is necessary, in this thesis, to create a model in order to simulate a Sybil behavior.

The existing system offers no friendly means of changing the automatically generated key - value pair. Therefore, it is necessary to assign identifications that could simulate the Sybil behaviour inside the code. It has proved to be possible to change the identification of some of the Sybil nodes to be redundant identifiers at the code level. The Sybil

nodes are inserted in a random pattern chosen before launching the simulation. For example, the first Sybil node is located at the 10th position, the second at the 15th position, the third at the 21th position, and so on.

The strategy of running the simulation has been presented in a flow-chart shown in figure 22.

1. Out of 500 nodes used in the simulation 10% (50 nodes) have been introduced as Sybil nodes.

2. 50 Sybil nodes have been assumed to possess similar identities

3. The Sybil node does not participate in any other malicious activity other than compromising the privacy of confidential information intended to be forged as a fake personality.

4. Finally, to show the seriousness and volume of attack that might result from using some other kind of malicious activities on the top of the Sybil attacks, another attack, known as routing table poisoning, has been added.

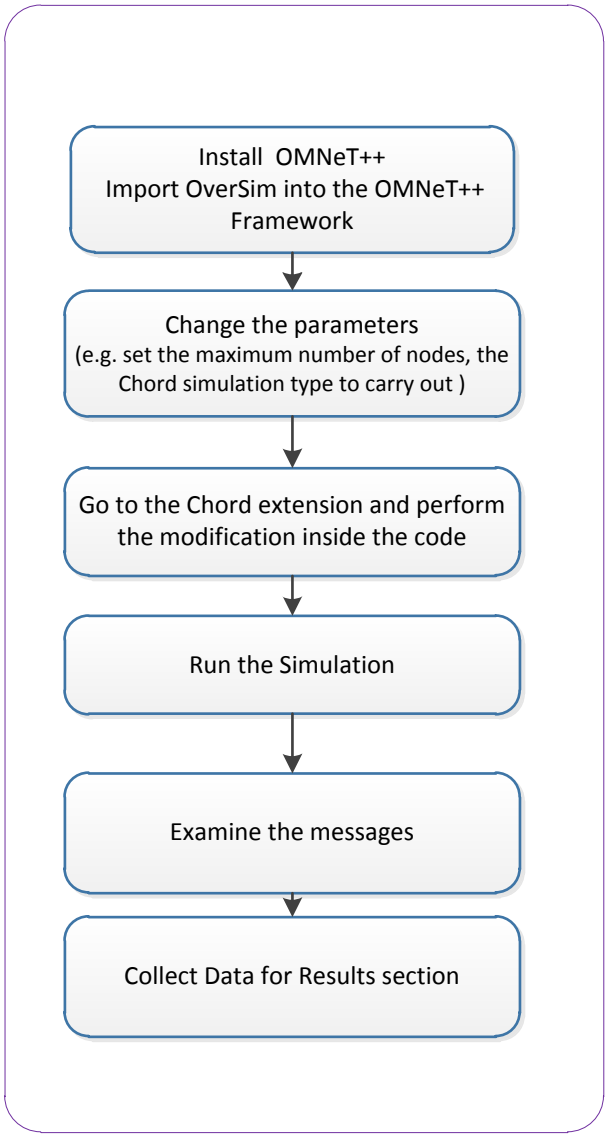


Figure 22 Strategy for simulation of Sybil behavior.

4.3.2 The Model on OverSim

The existing system for the Chord simulation and model can be seen in figure 16 for the simulation of Sybil behaviors and it follows the same procedure, the exception being that the developed model has nodes with multiple identities, simultaneously. Although these nodes are unable to own any identification that they might desire, there is still the chance to defraud by disseminating malicious information. On the top of a Sybil attack, a routing poisoning attack could be used to increase the volume of attack, by feeding the wrong routing information to the legitimate nodes. In effect, the legitimate nodes are led to communicate through the Sybil nodes.

4.4 Routing Table Poisoning

The routing table poisoning (or RTP) attack has been introduced by feeding the routing table with some of the legitimate nodes with information of malicious intent. The aim in this section is to forward the data, coming from the legitimate nodes, through the Sybil nodes.

In practical cases, in order to gain a higher fraction of the entire network, malicious peers would forward forged information about the Sybil node. Doing so increases the influence of the Sybil attack as compared to a Sybil attack carried out without RTP. However, in the simulation, the RTP could be performed directly rather than using malicious nodes to distribute forged information. Therefore, false routing information has been fed into the finger table.

Each Chord node retains entries of nodes necessary to trace out to neighboring nodes. The finger table holds entries of nodes directly in connection with a node. Thus, changing the entries in the finger table will definitely poison the normal routing.

4.5 Data Traffic Analysis and Examination

In the simulation, identifying the Sybil nodes is not being a problem since they have been provided with different labels. However, to discover which of the legitimate nodes are affected by the routing table poisoning and which of the nodes have been manipulated to consider the Sybil nodes as a real, a close examination of the messages has been conducted.

OMNeT++ provides important tools necessary to examine the messages exchanged between nodes. The number of packets exchanged with each node, the kind of message exchanged, the sources of messages and related detailed information could be collected using the OMNeT++ features (See figure 23, 24, 25). Therefore, the messages coming from the Sybil nodes and other genuine messages could be identified in such a way that involves a little time and energy being used with the tools shown below.

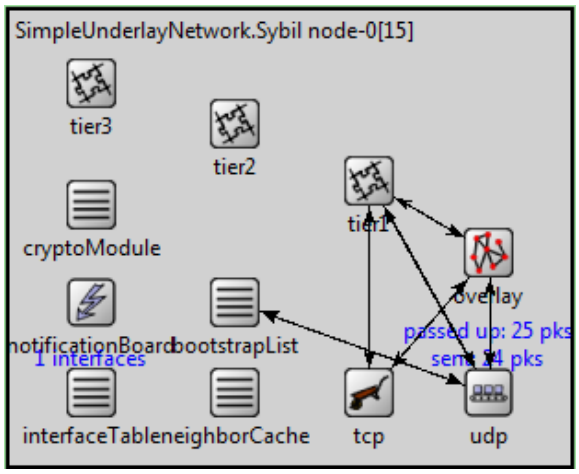


Figure 23 Sybil Node: shows the packets sent and passed up

Figure 23 shows the total number of packets forwarded as well as the total number of packets sent from Sybil node 15. The number of packets forwarded and sent will continue increasing as the simulation time goes on. Thus, the influence of the Sybil attack increases as more and more nodes are deceived into considering that the Sybil node is a genuine node which could assist in routing to another node. In other words, more and more nodes retain the Sybil nodes in their routing table entries. This is particularly true when the node interval time is increased and the malicious routing entries also increase, see chapter five.

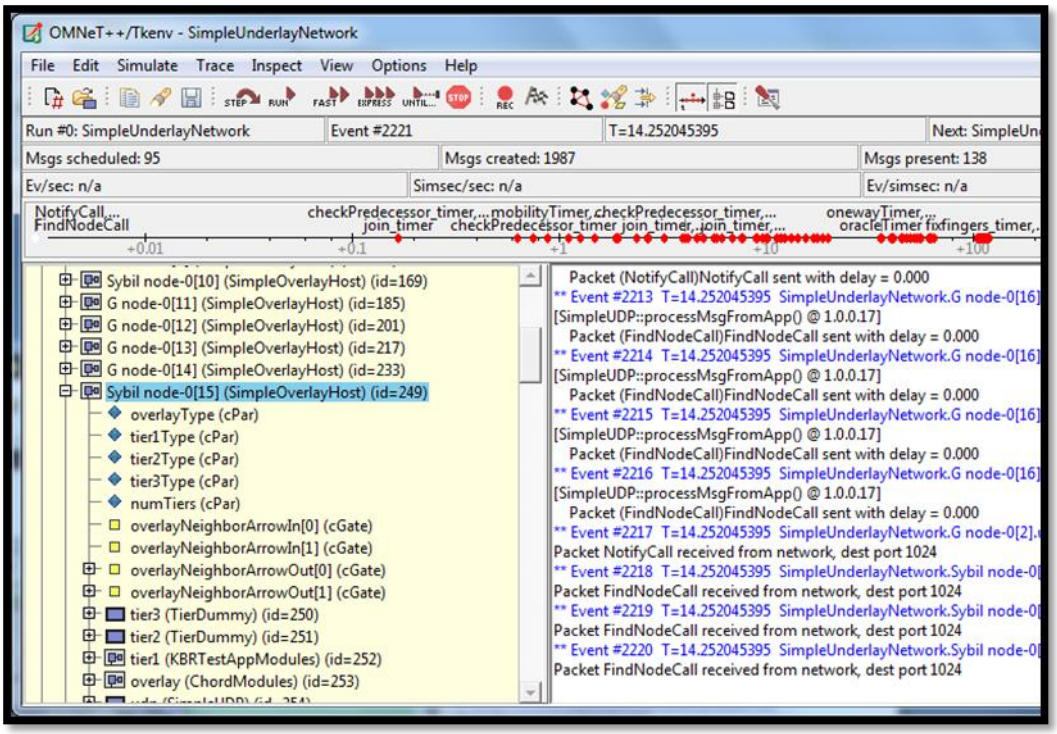


Figure 24 Sybil node shown with an identification of 249

On the left-hand side of the framework, as shown in figure 24, are shown a list of nodes in the overlay network and on the right-hand side of the IDE are a list of packets coming from each of the nodes. Moreover, under each of the nodes listed, the packets received and transmitted could be retrieved.

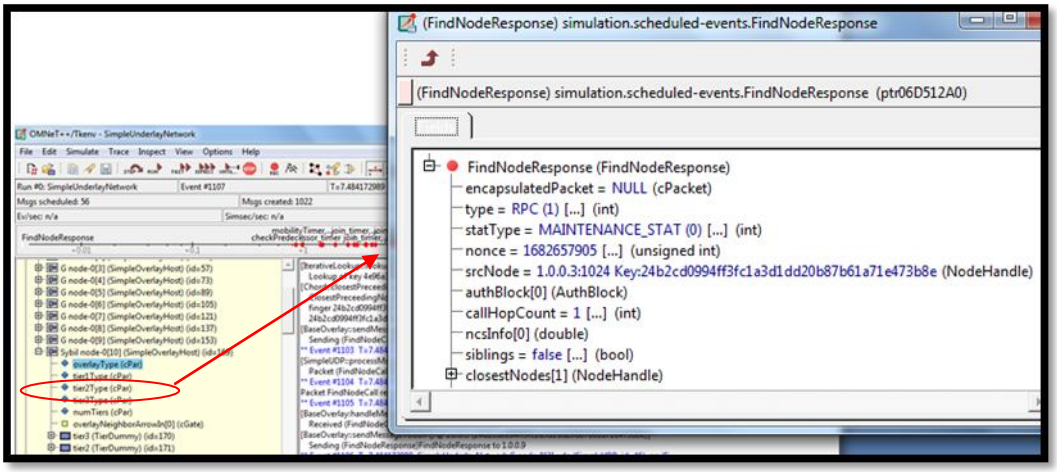


Figure 25 Node response Screen for Sybil node: provides information of the incoming message

Figure 25 shows in detail the content of the packet, the message type and the source node. The message "*FindeNodeReponse*" is received from the node with an IP address: *1.0.0.3* is to be handled by the Sybil node as shown in the figure. A list of messages sent and received could be examined this way. The message type "*FindNodeRespose*" is a basic rpc node find call. Other messages are *BaseOverlayMessage* and *NewSuccussorHintMessage*..

5 Results

In this chapter, results based on experiments conducted in this section, considering the factors that could intensify the magnitude of Sybil attack are presented. In order to study the impact of the Sybil nodes within the structured peer-to-peer networks, two factors have been considered. The first factor is the time that the Sybil attack is allowed to stay alive and the second factor is the fact that additional attacks, such as routing table poisoning, could be used to expand the influence of the Sybil attack. Accordingly, in this section four different observations have been made based on these factors and the results are illustrated by means of graphs.

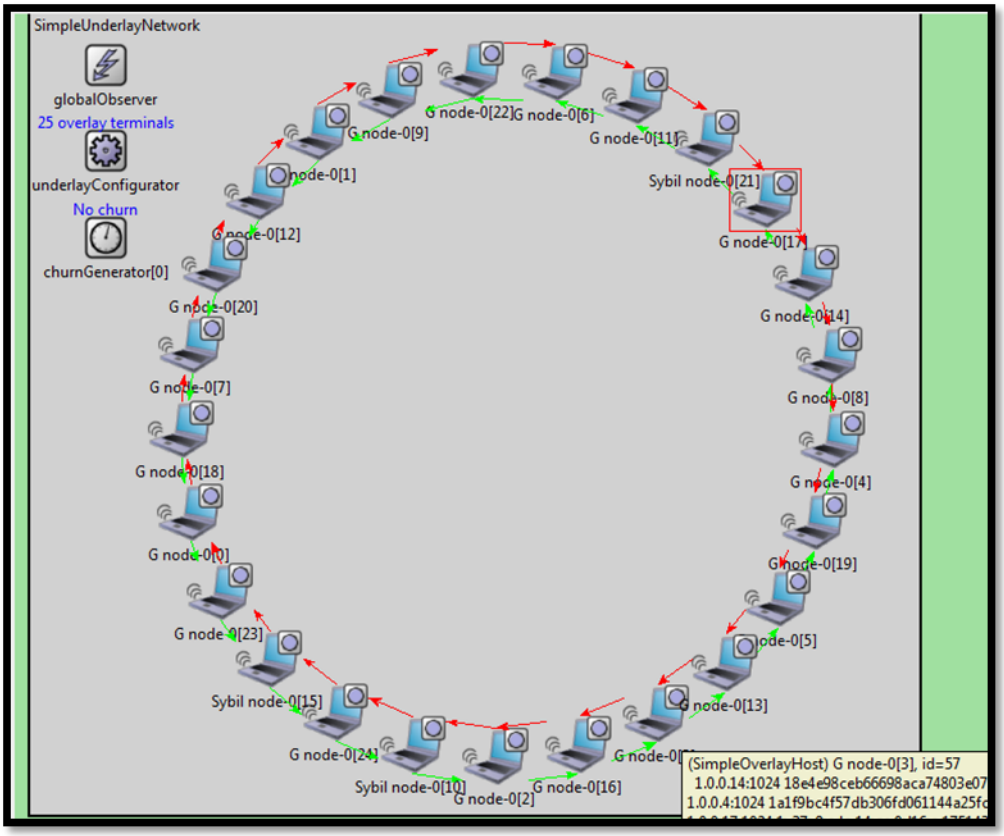


Figure 26 Sybil attacks simulation using OverSim. G node represents Genuine and Legitimate nodes and three Sybil nodes.

The modified version of OverSim displays which nodes are a Sybil and which nodes are genuine during the simulation as in figure 26. More-

over, the finger table of the Sybil nodes carries entries whose aim is to masquerade their real identity and which assist in performing malicious activity.

5.1 Simulation

The simulation of the Sybil attack on the existing OverSim simulator has been carried out using 500 nodes, some of which are Sybil nodes. The details of each of the nodes could be queried as shown in figure 23.In this case the insertion of the Sybil nodes is based on predefined identifications, for example, the 10th ,15th, 21th ... and so on have been assigned(See figure 26). In the previous chapter it is explained that in order to assign multiple identities, coding must be carried out. In the existing system, by intervals of 10s, a new node joins the DHT until the maximum number of nodes 500 is achieved. Thus, in order to label some of the nodes as Sybil, the following piece of code is added:

```
std::string nameStr = "G node";
std::string nameStr_ = "Sybil node";

if (churnGenerator.size() > 1) {

    nameStr += "-" + convertToString<int32_t>(type.typeID);
}
if (overlayTerminalCount == 10 || overlayTerminalCount == 15
        || overlayTerminalCount == 21||(overlayTerminalCount>30&&overlayTerminalCount%10=0))
    nameStr = nameStr_;

nameStr += "-" + convertToString<int32_t>(type.typeID);
cModule* node = moduleType->create(nameStr.c_str(), getParentModule(),
        numCreated + 1, numCreated);

std::string displayString;

if ((type.typeID > 0) && (type.typeID <= NUM_COLORS)) {
    ((displayString += "i=device/wifilaptop_l,") += colorNames[type.typeID
            - 1]) += ",40;i2=block/circle_s";
} else {
    displayString = "i=device/wifilaptop_l;i2=block/circle_s";
}
```

Unless otherwise stated, the following parameters and tools have been used in the simulation: SimpleUnderlay used as the underlay model, a total of a maximum of 500 Chord nodes, a 15min DHT refresh interval and 3 hours of average node interval. The simulation has been run for a total of 15 hours.

5.1.1 Data gathering strategy

The Sybil simulation conducted in this section is based on the knowledge gained from the literature study. Therefore, the expectations are to be able to show the behavior using the tools chosen.

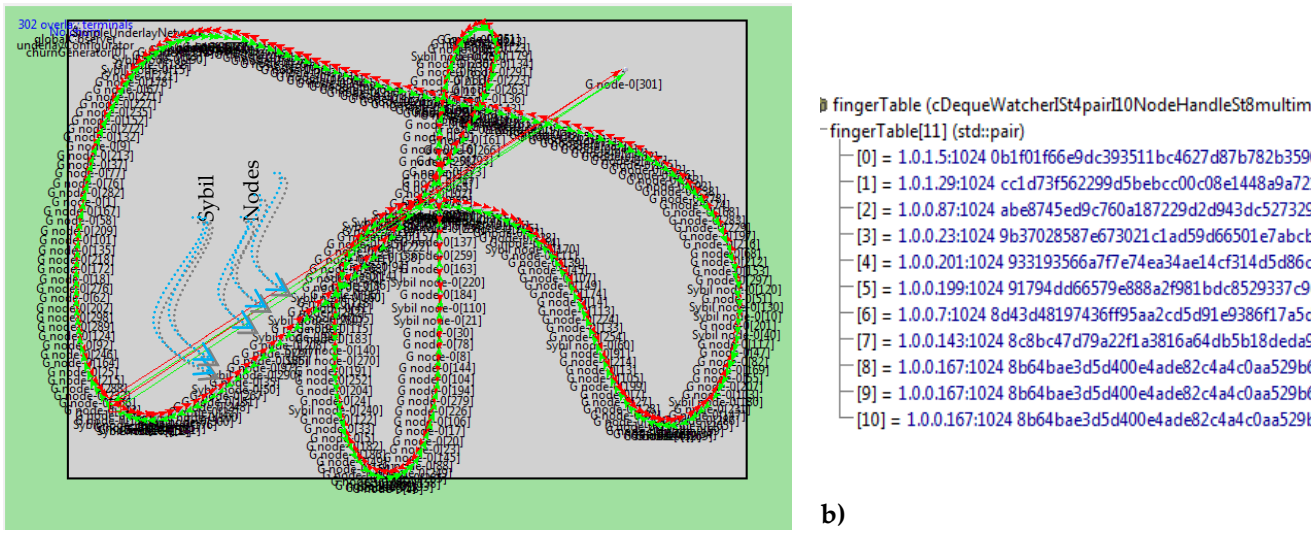


Figure 27 Sybil attack on Chord DHT a) DHT with 302 number of nodes b) Finger Table containing list of Genuine and Sybil entries.

As shown in figure 27b, each node maintains a list of entries of a node in the finger table, thus, through time, the list of nodes within the finger table changes. In order to discover the influence of the Sybil nodes, the finger tables are used. The finger table displays the IP addresses of entries of nodes, therefore, to determine the influence of the Sybil node means that the IP address of the Sybil nodes is known, which is obvious and to count those malicious entries over a period of time, as shown in the next section.

In this section there are four cases considered:

Experiment One:

In this experiment malicious activity being simulated is merely Sybil behavior so is based on solid knowledge gained in the literature review, which explains that the Sybil nodes are unable to increase in number by themselves once they are part of the DHT, however, they still carry out their harmful malicious intent as planned beforehand. Therefore, to simulate this behavior means to insert nodes with multiple identities. Here, the simulation is run for 15 hours and through time genuine nodes as well as Sybil nodes were added. The observation in this case shows that the influence of the Sybil behavior (the number of Sybil nodes, the messages exchanged) is linear, as it is expected.

Experiment Two:

In this experiment the goal is to force genuine nodes to route through a Sybil node by feeding the routing tables (or finger tables) with misleading routing information. As a result, it is possible to bring the topologically distant peer as close as possible to the harmful Sybil node. In this experiment, the nodes forwarding messages through Sybil nodes have been counted. Therefore, data which reconciles with knowledge gained from the literature study has been gathered and presented in the next section.

RTP – Routing Table Poisoning

In this case attackers might systematically introduce malicious information to increase the influence of the already existing Sybil activities. Thus, in this case, nodes disseminate forged information about Sybil nodes. As a result, an increasing number of nodes will place these malicious nodes in their routing entries.

The parameters:

- Total number of nodes: 500
- Sybil nodes: 10% in all observations (except experiment one which takes increasing number of Sybil nodes).
- Average node interval time: 3 hours, 15 hours.
- Simulation time: 15 hours.
- Other factors: RTP , No RTP

Case One (experiment one):

1. Average node interval time 3 hours
2. No RTP
3. 50,100,150,…400 Sybil nodes
4. Simulation time: 15 hours

Case Two:

1. Average node interval time 3 hours
2. RTP
3. 10% Sybil nodes

4. Simulation time: 15 hours

Case Three:

1. Average node interval time 15 hours
2. No RTP
3. 10% Sybil nodes
4. Simulation time: 15 hours

Case Four:

1. Average node interval time 15 hours
2. RTP
3. 10% Sybil nodes
4. Simulation time: 15 hours

5.2 Evaluation and analysis

In order to quantify the seriousness of the attack posed by the Sybil nodes, each entry was examined in detail and the routing table of each node was used to discover if there are Sybil entries. Moreover, in this evaluation and analysis it has been observed that the percentage of Sybil nodes and the fraction of routing-table entries are proportional, as presented in figure 28. To investigate in what situation a Sybil attack could have a huge impact on the security of an overlay network, routing table poisoning have been introduced through time. In this case, what has been examined is, over an interval of one hour, the routing tables containing entries forwarding other legitimate nodes to Sybil nodes have been counted. Thus, as presented in table 1, the percentage of Sybil nodes and the percentage of malicious routing entries have been collected. In order to avoid any fallacies and arrive at the right conclusion, the simulation is run multiple times. In the last experiment 10% (50 nodes) Sybil nodes have been introduced.

In addition , the other important factor which might contribute to an increased Sybil influence, has been the time Sybil node are allowed to run within the peer-to-peer network. The longer the Sybil node stays alive within the DHT the higher the percentage of Sybil influence. Thus, in the second experiment, the network is run for 15 hours and the average node interval time of 15hrs is set, however, in the first experiment, the average node interval time is 3hours.

In experiment two, for a total simulation time of 15 hours each time, four observations have been made. The first scenario has been the case where the RTP has been used in addition to the Sybil attack and where the average node lifetime is 3 hours. The second observation has been the case where the average node lifetime has been increased to 15 hours and there is no RTP attack. In both the third and final observation, they have been conducted when the average node lifetime remains as 15 hours and an RTP is used to control a larger part of the peer-to-peer network. In addition, for comparison purposes, the experimental results from experiment one have been added (See the table 1). Therefore, in this experiment only 10% of Sybil nodes have initially been inserted but, in the cases where the RTP and/or the average node interval time has been increased, the percentage of malicious routing table entries has increased.

Table 1 Result – Percentage of malicious routing table entries through time

Simulation Time (hrs)	**% of malicious RTE[2]without RTP** avg. node life time = 3hrs	**% of malicious RTE2 with RTP** avg. node life time = 3hrs	**% of malicious RTE2– without RTP** avg. node life time = 15hrs	**% of malicious RTE[2]- with RTP** avg. node life time = 15hrs
0.75	10%	10%	10%	10%
2.4	10%	20%	10.4%	20%
3	10.3%	25%	10.7%	25%
4.8	10.3%	27.5%	11%	28.1%
5	10.4%	29%	14%	30.2%
6	10.4%	29.5%	16.7%	33.9%
7.2	10.4%	30%	19%	37.1%
8	10.4%	34%	22%	38.5%
9.6	10.4%	35%	23.6%	42.2%
10	10.4%	38%	23.7%	45.3%
11	10.5%	39.6%	23%	47.0%
12	10.4%	41.4%	23%	48.6%
13	10.4%	43.7%	22.8%	50.2%
14	10.4%	46.1%	23%	51.9%
15	10.4%	46.8%	23.6%	55.8%

[2] RTE – Routing Table Entries

Based on the results using RTP, a Sybil attack is able to cover more than 45 % of the entire network by only using 10% of nodes. Moreover, if the Sybil attacker is allowed to run for longer period of time , they are able to cover a larger part of the network (in the experiment such attacks are able to cover more than double of the case when average node time interval is 3hours.). The other serious case observed shows that a Sybil node being allowed to run for a long period of time (in the experiment 15hrs) means that the percentage of malicious entries could reach more than 50 percent (See table 1).

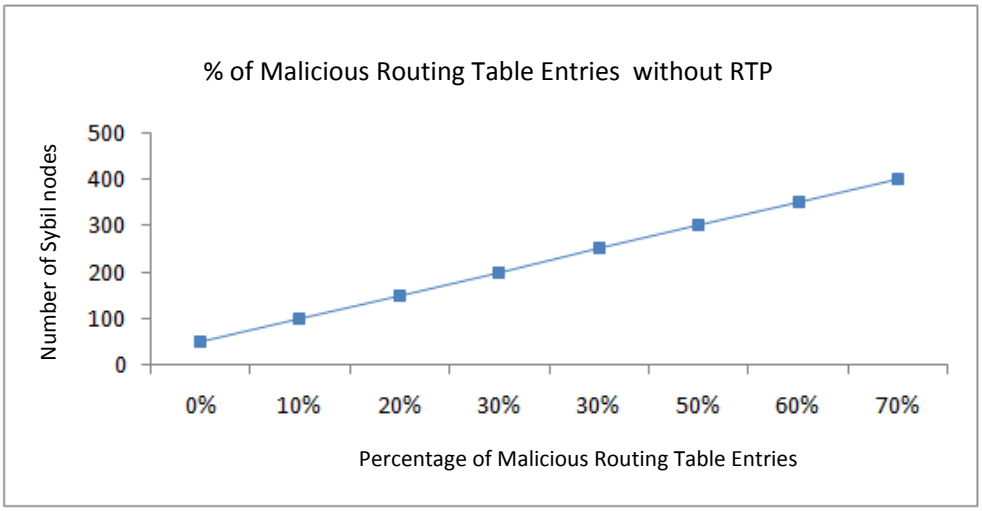


Figure 28 Sybil attack without routing table poisoning

The graph shown in figure 28 above reveals the linearly increasing nature of the Sybil behaviour. In other words, for a maximum of 500 nodes, if a constant number of Sybil is introduced, for example, 10% of Sybil nodes, then only 10% of the entire overlay network experiences a Sybil attack. Therefore, this behaviour shows a linear increase in the entire overlay network throughout the simulation time. In some cases, when the attacker gains access to run Sybil nodes for a longer period of time, the number malicious routing entries increases to some extent. In the experiment, the longer living Sybil nodes are given 15hrs and this resulted in an increase in malicious routing entries as compared to Sybil nodes with an average node interval of 3hrs see graph in figure 29..

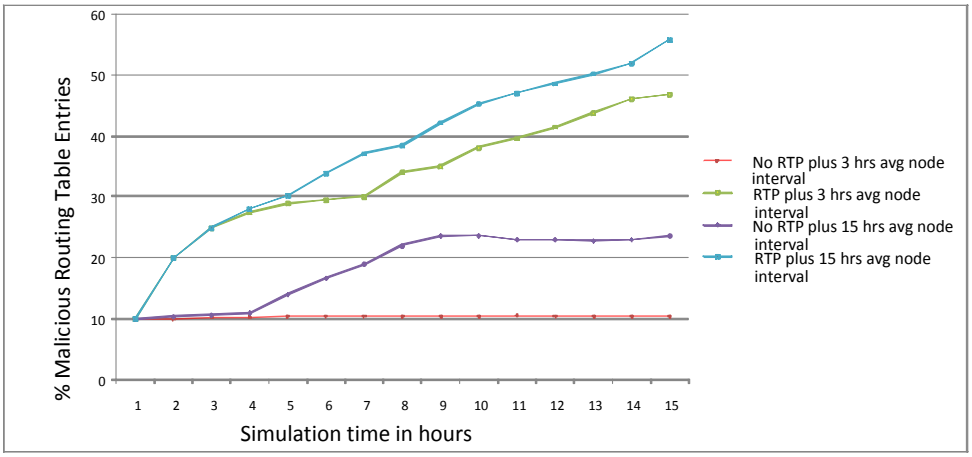


Figure 29 Sybil attack plus routing table poisoning through time

Moreover, during the simulation, it was observed that the Sybil behaviour could be used to compromise the security and privacy of a DHT network and cover a greater number of nodes by including some other malicious activities, which could disseminate the false identities of the Sybil nodes resulting in more and more influence of Sybil behaviour in the entire overlay. Therefore, legitimate nodes are misled into forwarding messages through malicious nodes. The routing table poisoning could be used, together with a Sybil attack, to cover a larger part of the overlay network than an attack which is carried out by the Sybil nodes alone. Throughout the time of the simulation, the number of Sybil nodes remains constant; however there is an increase in the number of legitimate nodes passing messaging through the Sybil node. Figure 29 shows that there is a logarithmic increase in the fraction of malicious routing table entries. The result in the graph shows that more and more messages from legitimate nodes are being forwarded through the Sybil nodes. This is thus a better opportunity for those Sybil nodes interested in compromising the security and privacy of the victim. For Sybil nodes, the opportunity such as this could mean a higher possibility to attack any node with only a minor challenge of topological "distance".

Note: The curves illustrated in figure 29 are not smooth and some of them appear to be rather different to logarithmic curves. The reason for this is based on the size of data collected. In addition, smooth logarithmic looking curves could be drawn if the simulation time is much longer (for example 3 days or more).

5.3 Proposed Mitigation mechanisms

As mentioned in the previous chapters, fully distributed systems are unable to completely eliminate Sybil attacks. However, it is possible to reduce the attack. The issue of mitigation of the Sybil attack causes a return to the traditional centralized system, at least for the assignment of node identifications. If there is a central authority, which could assign node identification, then even if a node requests multiple IDs it will not have an ID, which might serve its adversarial purposes.

It has been explained and discussed in the previous sections that a fully distributed system does not have any central authority at any level. Thus, if zero tolerance is adapted to compromise the fully distributed nature of the overlay then a Sybil attack is inevitable. In this case, the assumption is of a network in which any node can join for free and is able to pick its own identification. However, the proposal here is to use external identifiers as a means to create a Sybil-proof environment. In this regard, the convenient and free identifier is the IP address. Thus, each of the nodes participating in the overlay will have a node ID derived from an IP address and through this mechanism it is possible to provide a unique identification for each node, however, a node having one IP address might not be able to own more than an allocated maximum limit per IP.

5.4 Sybil attacks on the MediaSense Internet-of-Things platform

Sybil attacks are harmful attacks posed in any peer-to-peer, sensory and ad hoc network. Taking into consideration the fully distributed nature of the Noble MediaSense, a Sybil attack causes no less trouble than as in the other cases. The Sybil attack could seriously affect the MediaSense Platform. A malicious peer, which is part of the MediaSense Platform, could disseminate misleading information regarding his/her situation or any general information. For example, in an application where peers could exchange their geographical locations, a Sybil could mislead other peers by impersonating its identity. As a result of such incidents, other peers might lose trust in the entire structure. In effect, the MediaSense loses its confidentiality. Serious and devastating consequences might follow in a situation when an adversary is able to impersonate multiple identities within, for example, the health monitoring system based on the MediaSense Platform.

5.4.1 How to carryout confidential data exchange on MediaSense Platform

The priority consideration for the MediaSense platform has been to serve as an overlay that could be used as seamless, scalable and real-time communication architecture over a heterogeneous network. Furthermore, the MediaSense platform has been cited as an alternative approach to be used in an area such health monitoring and objects tracking where privacy is a top priority. Therefore, other crucial parameters, for example, the security of the data must be considered in some situations. Nevertheless, the peer-to-peer nature of the MediaSense platform offers a little flexibility for secured data communication. It is believed by the author that the confidentiality of data communication could be slightly improved if one central body exists, which assigns the identification for each node during the join for some types of applications demanding high security. However, a central point of failure might occur and, in that case, there will be no joining nodes thus the entire system will fail eventually. The kind of mechanism for central identification assignment has been explained in section 2.5.3.

In addition, distributed group-based identifier assignment could be considered although this mechanism does not guarantee a Sybil-proof environment. As briefly explained in Chapter 2, in distributed group-based identifier assignments a group of nodes asserts the new nodes. This reduces Sybil attacks however; there is no guarantee as to whether the group by itself is Sybil free. One aspect for consideration is that a little more resistance is better than no resistance. However, this mechanism could be used with other means such as central identification assignment and provide a better means of handling the issue of a Sybil threat. In this case, it is possible to authenticate the group of nodes that assert the new nodes through a central authentication, which means that the asserting group of nodes can be assured to be Sybil free, thus, the new joining nodes are Sybil free.

6 Conclusions

It has been explained in the theory section that a Sybil attack is a serious security threat against structured peer-to-peer overlay networks. However, in order for a Sybil to control considerable resources, the Sybil node must offer a huge amount of resources. Nevertheless, malicious activities, such as routing poisoning attack could aggravate the attack and assist in covering more parts.

In this thesis, a simulation of a Sybil attack-multiple identity impersonation has been conducted on one of the most commonly available structured peer-to-peer networks, Chord. Based on the simulation and the subsequent examination, the number of Sybils and the resources they use are proportional, for example, if 20% Sybils are inserted then 20% of the entire Chord overlay experiences a Sybil attack. However, as each Sybil node is allowed to live longer within the network it has been observed that more genuine nodes are mislead into placing malicious nodes into their routing a tables.

The first goal of the thesis has been to study the efficient attacks on structured peer-to-peer overlay networks. Accordingly, security threats and attacks on a peer-to-peer overlay network have been presented: Sybil Attack, Node ID attack, Eclipse attack, Denial of Service attack, Data Forwarding attacks, attacks on placement schemes are the most serious threats to a structured peer-to-peer overlay network.

Regarding the second goal, which concerns the simulation of a Sybil attack, one of the popular structured overlay network has been chosen and simulations have been conducted on an OMNeT++ network simulator. In this case, the libraries from OverSim developed for Chord overlay have been reused.

The third goal is to evaluate the impact of these attacks on the performance of the overlay network. In this regard, it has been shown how the Sybil attack could affect the integrity and security of a structured overlay network. Furthermore, carrying out some other kind of attack, such as a routing table poisoning on the top of the Sybil attack could be used as the best means of controlling the vast majority of nodes.

The fourth goal is to evaluate the currently available mitigation mechanisms against these attacks. In this regard, the thesis has presented the commonly used mechanisms to create a Sybil-proof overlay. Self-registration, computational puzzles and resource testing have been presented.

6.1 Projects newsworthiness and contribution

Simulation of the Sybil behavior using the OverSim is being observed by concerned researchers who are engaged in related research work. The observation was made during the project proposal preparation stage. However, no simulation of such an attack on the OverSim during the writing of this thesis has been witnessed. In fact, there are recent and older research papers which attempted to study the Sybil behavior. The general truth is that the Sybil attack is one of the most serious attacks and thus gaining a good knowledge in relation to such an attack is very important. Therefore, the contribution is to confirm the theoretically agreed features using the experiment. Nevertheless, the ultimate contribution is to derive a simulation of Sybil behavior using one of the widely used network simulation frame work.

The advantages of conducting simulations of particularly serious attacks such as the Sybil attack are immense. Simulations performed in this thesis can assist in the understanding of the impact of the threat on structured peer-to-peer networks. Therefore, the work carried out shows the serious nature of the attack especially when combined with other malicious activities. The knowledge gained here would thus be a better ground for researchers interested in focusing on the mitigation of an attack. As compared to related research conducted to study the behavior of a Sybil attack, the results do not show any significant variations although the simulations used different DHTs.

The knowledge thus gained from this thesis work is twofold. Firstly, the simulation of Sybil attack has been conducted. As mentioned, the simulation shows the breaches regarding the attacks and the measures to be taken in order to avoid or at least to reduce the Sybil attack on structured peer-to-peer networks. Secondly, the seriousness of such an attack has been evaluated. In this case another mechanism, namely the routing

table poisoning attack that could aggravate the seriousness of the attack, has been studied in detail.

Thus, the possible impact of the research work carried out in this case would be that, the research confirms the theoretically gained knowledge from previous research as presented in chapter 2 and that the results gained from the research work could be used as a knowledge base for related future studies.

6.2 Challenges and difficulties encountered

The simulation of a Sybil attack using an OMNeT++ simulator is not as easy as it outwardly appears. The main challenges in the project are the following.

The first challenge was the non-availability of sufficient documentation and tutorials on the OverSim. In addition, the OverSim is unable to be considered as merely a simple tool to simulate DHTs as the coding and the libraries are so large. Thus, knowledge regarding the Chord DHT implementation has required much time and energy.

The second challenge was the requirement of knowledge with regards to rather unfamiliar programming languages within the entire program.

The last, but not least, difficulty has been how to forge multiple identities in a code that was never intended for such modification.

6.3 Ethical Deliberations

The result from this thesis can enable future research to be carried on the attacks and threats on structured peer-to-peer networks. In addition, the study and its results can assist in gaining knowledge in relation to one of the harmful attack to a structured peer-to-peer network. Therefore, parties interested in utilizing a peer-to-peer overly network could gain knowledge and better understanding regarding a Sybil attack as well as the mechanism to reduce the possibility of the attack.

However, as it has been explained in the previous chapters, the research conducted in the thesis has followed standard rules with regards to writing reports, interactions with concerned parties and the usage of resources. Therefore, any negative impact of the results on society has been avoided. In this case, one of the common problematic issues, which

are related to referencing previous works has been made so as to avoid plagiarism.

6.4 Future work

A Sybil attack is considered as a particularly serious attack not only on the structured overlay network but also on the sensory and Ad hoc networks. The future plan regarding an extension to the work of this thesis involves two dimensions. Firstly, the potential of a Sybil attack in both sensory and ad hoc networks deserves study; secondly, the next step is to run a simulation using a huge number of nodes, which was not possible in this study because of limited resources. The thesis has carried out a research on one type of DHT, known as Chord, but in future work the interest is to continue in the following directions:

1. The impact of a Sybil attack on sensory networks and the mechanisms to prevent the attack.

   Sensory networks are promising technology to enable socially and economically valuable solutions in different areas of life, for example, environmental protection, and national security, industrial and military sectors. A number of applications, based on the sensory network, require data communication. However, the issues in this case become somewhat complicated as the memory of the nodes would not allow for any larger data size-security overheard. Furthermore, serious security threats, for example, the Sybil nodes could further worsen the challenges. Therefore, future work would be to evaluate the significance of a Sybil threat on sensory networks as well as presenting the existing means for countermeasures and to propose any other means to counter and prevent the Sybil threat.

2. The impact of a Sybil attack on ad hoc networks and the mechanisms to prevent the attack.

   Ad hoc networks are an emerging mobile computing and are mechanisms for multicast communications; however, due to the inherent nature of ad hoc networks, security threats such as the Sybil attacks could cause concern. The future work in this case would be to study the significance of a Sybil attack on ad hoc networks, discuss the presentation mechanisms as well as propose mitigation mechanisms.

3. Implement the MediaSense Simulation module for OverSim.

   As explained in the theory sections, OverSim is a network simulation module, intended to run on the OMNet++ framework. Actually, the OverSim is only a C++ project package which can be imported onto the OMNeT++. So taking advantage of the open features of the OverSim, the belief is that implementing the simulation modules for the MediaSense could offer immense benefits for future and undergoing researches.

4. A Sybil attack on the MediaSense Internet-of-things platform and its consequences. In order to study the impact of a Sybil attack on the MediaSense platform, use the MediaSense Internet-of-things platform as the overlay network.

   In this thesis, as explained in detail, OverSim has been used to simulate the Sybil behaviour. Chord DHT has been chosen from the list of existing DHTs available on OverSim. However, the MediaSense is not available to simulate the Sybil attack. Furthermore, as mentioned in section 2.6.2 OverSim is open source, so the first step should be to implement the MediaSense Simulation module on OverSim. The implementation of a simulation module for OverSim could be, by itself, a full thesis work at the level of Masters in Computer Engineering. So if the module is on the OverSim, the next task would be simulating the Sybil behaviour on the MediaSense simulation module.

References

[1] Fujii, M. ; Watanabe, Y.; Hamai, T., ”ITU-T Recommendations on Peer-to-Peer (P2P) Network Security” , International Symposium on Autonomous Decentralized Systems, pp

[2] John F. Buford, H. Yu, Eng K. Lua,”P2P Networking and Applications”. Published 2008 ,pp 29

[3] F. chao, H. Zhang, X. Du, C. Zhang, "Improvement of Structured P2P Routing Algorithm Based on NN-Chord" 2011 7th International Conference on Wireless Communications, Networking and Mobile Computing (WiCOM), published 2011, pp

[4] Ion S., Robert M., David L., David R. Karger, M. Frans K., Frank D., and Hari B., “Chord: A Scalable Peer-to-Peer Lookup Protocol for Internet Applications”, IEEE Transactions on Networking, VOL.11, NO.1 published 2003, pp

[5] A. Rowstron, P. Druschel, “Pastry: Scalable, decentralized object location and routing for large-scale peer-to-peer systems? “ published 2001, pp 3

[6] Zhao, B.Y., Ling Huang ; Stribling, J. ; Rhea, S.C. ; Joseph, A.D. ; Kubiatowicz, J.D,”Tapestry: A Resilient Global-Scale Overlay for Service Deployment” published 2004 pp

[7] K. Aberer, P.Cudré-Mauroux, A. Datta, Z. Despotovic, M. Hauswirth, M. Punceva, R. Schmidt ”P-Grid: A Self-organizing Structured P2P System” Distributed Information Systems Laboratory ÉcolePolytechniqueFédérale de Lausanne (EPFL), published 2003, pp

[8] T. Kanter, S. Forsström, V. Kardeby, J. Walters,U. Jennehag, P. Österberg, “MediaSense – an Internet of Things Platform for Scalable and Decentralized Context Sharing and Control”, ICDT 2012, The Seventh International Conference on Digital Telecommunications , pp. 27

[9] The MediaSense Platform, www.mediasense.org Retrieved June 1, 2013

[10] Jyothi B S, D. Janakiram, ”SyMon: Defending Large Structured P2P Systems Against Sybil Attack ” published 2009,

[11] G.Urdaneta, G.Pierre, M. Van Steen, ”A Survey of DHT Security Tecniques”http://www.globule.org/publi/SDST_acmcs2009.pdf Retrieved March 5,2013

[12] M. Vestola, ”Security Issues in Structured P2P Overlay Networks” Helsinki University of Technology, 2010

[13] D.Cerri,A.Ghioni, S. Paraboschi, S. Tiraboschi,”ID Mapping Attacks in P2P Networks” Global Telecommunications Conference, volume 3, published 2005

[14] S. Farraposo, L. Gallon, P. Owezarski," Network Security and DoS Attacks" published 2005 ,Page 6

[15] I. Baumgart, B. Heep, S. Krause ,”OverSim: A flexible overlay network simulation framework ” IEEE Global Internet Symposium, published in 2007 pp. 1

[16] What is Cryptography? http://www.emc.com/emc-plus/rsa-labs/standards-initiatives/what –is-cryptography.htm Retrieved May 1 , 2013

[17] M. Castro , P. Druschel , A. Ganesh1, A. Rowstron, Dan S. Wallach,”Secure routing for structured peer-to-peer overlay networks”, Proceedings of the 5th symposium on Operating systems design and implementation, Published 2002pp 5

[18] Douceur,”The Sybil Attack” In 1st International Workshop on Peer-to-Peer Systems (IPTPS ’02) Springer, 2002.

[19] J. Dinger , H. Hartenstein ,”Defending the Sybil Attack in P2P Networks: Taxonomy, Challenges, and a Proposal for Self-Registration” Preceedings of the first international conference on availablity , reliability and security, published 2006 pp

[20] What is OMNeT++?
www.omnetpp.org
Retrieved April 1, 2013

[21] The Overlay Framework,
www.oversim.org,
Retrieved April 4, 2013

[22] Singh,A. ; Ngan, T.W. ; Druschel,P, ; Wallach, D:S "Eclipse Attacks on Overlay Networks" INFCOM 2006. 25th IEEE International Conference on Communications.

[23] Narses Network Simulator, 2003,
http://sourceforge.net/projects/narses
Retrieved 04-Nov-2013.

[24] S. Joseph, "NeuroGrid: Semantically Routing Queries in Peer-to-Peer Networks," Proceedings of the International Workshop on Peer-to-Peer Computing (2002).

[25] NS-2, http://nsnam.isi.edu/nsnam/index.php/Main_Page,
Retrieved 04-Nov-2013.

[26] P2Psim : A simulator for Peer-to-peer protocols,
http://pdos.csail.mit.edu/p2psim/ ,
Retrieved 04-Nov-2013.

[27] PlanetSim, An Overlay network Simulator
http://sourceforge.net/projects/planetsim/,
Retrieved 01-Nov-2013.

[28] W. Yang and N. Abu-Ghazaleh, "GPS: A General Peer-to-Peer Simulator and its use for modeling BitTorrent," Mascots, vol. 00, pp. 425–434, 2005.

[29] PeerSim: A peer-to-peer Simulator,
http://peersim.sourceforge.net/,
Retrieved 01-Nov-2013.

[30] Overlay Weaver: An Overlay Construction Toolkit,
http://overlayweaver.sourceforge.net/,
Retrieved 01-Nov-2013.