Anonymous Communication

Imran Liaqat, Muhammad Raheem
Linkoping university, Sweden
Email: {imrli827, muhib928}@student.liu.se

Abstract

This report contains different methods used to provide anonymity in order to provide secrecy. It describes the two broad techniques to achieve Anonymity (Mixes and DC-Nets).The main focus of the report is on the comparison and evaluation of the sub mechanisms involved in the above two techniques and how they actually work to provide anonymity. Additionally, it also highlights some attacks which will disrupt the process of the Anonymous Communication. At last, this report will give our own evaluation and results from the perspective of efficiency, how and what kind of anonymity is achieved.

1. Introduction

In the modern e-world, new technologies are rapidly increasing day by day. Security and privacy becomes the main issue over the last decade, by the immense growing popularity of the internet. Internet usage for online communication has increased a great deal since the last decade. People make online banking transactions, online shopping, emails, and e-tickets and so on. One can say that internet is an anonymous media; this is true in a sense that no one directly knows each other but the headers of the communication packets contain IP and email addresses. The browsers display emails and cookies stored on client machines having crucial information. Virtually we send every email, make any transaction or access any website, adversary can observe this information. Due to this lake of privacy over internet, anonymity is required for communicating parties, information and communication channels. With the cutting edge of technology, we can provide this communication anonymity. The main theme of this paper is to look at different communication anonymity infrastructures, their vulnerabilities, attacks against them, and their solution. Different mechanisms are also compared and analyzed according to criteria such as overhead, latency, time, complexity and performance.

2. Anonymous Communication

Providing Anonymous Communication in the open public networks is the problem arising in electronic world with the growing importance. Lot of Mechanisms introduces to provide the Anonymity of the systems, but mainly it is divided into two broad categories that are widely known in the real world, Mixes and the DC-Nets/Broadcasts.

2.1 Anonymity

With referring to human beings, if the identity of the person is not known, we say that the person is anonymous [1]. It is also true in the computer systems i.e., how to hide the information between the two communicating parties so that any other third party, intermediary, does not know about the true identity and the contents of the messages being sent over between sender and receiver. They should not know the sender, the receiver and the relation between them. Another same kind of word is used, called pseudo anonym which means the identity of the entity can not be known but it can be said that two acts were performed by the same entity [13].

2.2 Degree of Anonymity

The anonymity is measured using the concept of entropy (i.e., uncertainty).The metrics may be applied to measure the uncertainty of the attacker about the sender of the message, i.e., sender anonymity or the uncertainty of the attacker regarding the recipient of a message, i.e., recipient anonymity. [14]

2.3 Adversary

The adversary can have any form depending on his position. So there are so many possible adversaries, Eavesdroppers, administrators, servers, the receivers or target and individuals

2.4 Types of Anonymity

There are basically two type of anonymity but these also depend on the anonymity of other entities too. All of them are introduces in the preceding section. [7]

Sender-Anonymity

Sender anonymity means that hiding the true identity of the sender of information or resource user so that no one could say who has done it. It can have a form of true anonymity or pseudo anonymity. The negative usage of this anonymity can be, an adversary can anonymously do denial of service attacks, sending bogus emails containing malicious logic (viruses etc), or can violate integrity of information and we can not blame someone for this act because of anonymous entity.

Receiver-Anonymity

Receiver anonymity means hiding the identity of the receiver of the message from the world, even the sender should not know who the receiver as well as third party observer is. This is as important as sender anonymity.

Information-Anonymity

Information anonymity means that the information send over the network should have two properties; one is confidentiality and second is integrity. These rules must not be violated for true information anonymity. The tools used for this purpose are basically cryptography and encryption. If you want more security you may use source or destination authentication. [13]

Unlink ability

Unlink ability means that two parties taking part in a communication, not just between these two parties, should not be identifies by third party that they are communicating with each other. This is unlinking ability of sender and receiver. Unlink ability may take another form as if two transactions appear again and again in a communication, third party should not be able to link them that they are part of the same communication.

Node-Anonymity

Node anonymity means that the clients or servers taking part in a communication should not be identifies by third party as service providers and service takers. Otherwise this would eventually lead to sender or receiver anonymity violation.

Carrier-Anonymity

Carrier anonymity means that, the nodes in the path between sender and receiver should not be identifiable as communication links. They should quietly just receive and forward the data without knowing who is this distending for and where is it coming form. No one should be able to know that this carrier is used for communication between sender and recipient. [7]

2.5 Dining Cryptographer's (DC) Problem

David Chaum invented a cryptographic protocol in 1988, which is today called DC-Net. It is based on following scenario:

Three cryptographers are sitting in their favorite restaurant for a dinner. They noticed that NSA agent is also their in the restaurant. After eating their dinner, the waiter told them that the bill has been paid anonymously.

Now they all start wondering that who paid their bill, was it their favorite NSA (national security agency) agent or was it one of them because they respect each other's right to make an anonymous payment. [2]

So the solution is to find the protocol that allows a person to disclose knowledge of something without anyone else knowing that it was the same person who disclosed it.

That's how the DC problem started and the protocol is invented to solve it.

2.5 Types of Attacks and their prevention

There can be much type of attacks against communication such as social and technological attacks. The type of attacks against security or anonymity of communication depends upon, the adversary operating on the communication channel. As the security is as strong as the weakest link in the communication, therefore he looks for different kind of Anonymity such as sender, receiver, information, unlink ability, node and carrier anonymity. The one he finds weaker will be targeted. [7]

2.5.1 Communications nodes

The adversary is active in this kind of attacks as he is controlling of the communication nodes.

Denial of Service Attack> the system drops all the messages it receives. The prevention would be to send ping to the node before starting to send any data and analyze the response. If it has dropped some packets then it means this node is controlled by an adversary.

TracerouteCollusion Attack> distributed anonymous communication system is the prevention solution.
Cut-the-Channel Collusion Attack>distributed system is the prevention solution.
Trac Mangling Attack>the two prevention techniques are message encryption and strong partial anonymity.

2.5.2 Communications channels links

This attack is also based on active eavesdropping. The goal of adversary here is to identify the sender, receiver or their link ability. The attacks include the following.
Computational Attack: prevention is path encryption and encoding.
Message Coding Attack: prevention is end to end encryption and encoding.
Message Volume Attack: prevention is same size or random size messages in the system.
Trace route Replay Attack: the prevention technique is a nonce should be included in each message, and nodes should not resend a message that has already been sent.
Intersection Attack: no solution to this problem yet.

2.5.3 Communications channels edges

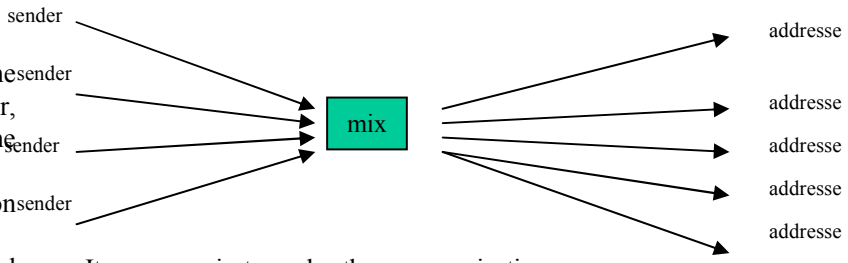
These category of attacks could be performed both by active and passive eavesdroppers at the edges of the communication channel.
Timing Attack: the prevention would be Adding a variable delay, latency and also dummy messages before sending real data.
Trickle Attack: no protection possible yet.
Identication Flooding Attack: prevention is dummy packets or latency
Identication Flooding Attack: prevention is dummy packets or latency
Pseudonymity Marking Attack: the prevention is, sender can defend against this attack by getting the receiver's name from a third party. [7]

4. Mechanisms used in Mixes

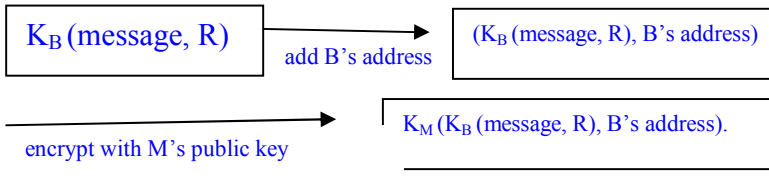
There are various mechanisms that are using the Mixes based methods in their communication scheme. The examples are chaum mixes, onion routing, web mixes, and crowds and so on. In this section we will discuss three of them. They are chaum mixes, web mixes and crowds.

4.1 Chaum Mixes

Chaum mixes was introduced as a solution to the secrecy problem of internet by Chaum. Chaum is a machine that sits between sender and recipient and intercepts all the communication between them.



Its purpose is to make the communication anonymous. The basic strategy deployed by this mechanism is to use cryptography, padding of constant size, constant rate, and mixing all the received messages to intermingle them. [13]
This mechanism provides only sender anonymity and works against traffic analysis.
The mechanism works as follows,
Let's suppose a communication between party A and B. A prepares the message by, a random value R is appended to the message, encrypted with the B's public key KB, adding the address of B. This packet is encrypted with the public key of Mix KM and send it to the Mix M. M opens it with its private key and wrap it up with public key of B and send to B.
At this stage, if an adversary is observing this communication, he can easily see that a message has sent to B from M and A to M. concluding that A and B communicated.



Hence the anonymity is only against the receiver. Eavesdropper uses a strategy to find out the exact sender, which is as follows. The message from M to B is intercepted and Bs address is attached to this message and replay it to M by sealing it with the public key of M. now the eavesdropper compares this message with the arriving messages at M. if a match is found then it confirms that A was the sender. To solve this eavesdropping problem, a random value is added to the message which is removed by the mix later on during the

transmission. We can also use series of Mixes instead of a single mix to provide greater protection. The message during the transmission from every mix is encrypted with the key of next mix and it makes an onion. This message is then decrypted at each mix with its private key and forwarded next. This way it reaches to its final destination.

Now the problem is that how can the receiver reply because the identity of the sender has been made anonymous. There should be a secure way to transmit the senders address to receiver so that he could also reply. The mechanism works as, the sender sends its address by adding two security parameters to it, one is random string and another is one time public key which is forwarded to Mix. The receiver creates the response by first decrypting the received message, taking the reply address of sender and finally using this address in the response. B sends this response to Mix with adding a random value and encrypting it with public key of M. after receiving by M; he opens it up through his private key and sends it to its recipient which was the sender.

Only the sender (now he is receiver) can decrypt this response because originally the random value and special key were created by him. So an eavesdropper intercepting this communication can no find out the sender's address. Hence this mechanism of chaum mixes provides good sender anonymity and protects against traffic analysis.

Drawbacks

1. The public key cryptography requires huge computation overhead, which makes this mechanism very slow, as explained in the section communication overhead of Mixes.
2. The latency rate is very high because the mix intercepts each message, opens its up and recreates it for the receiver.
3. This mechanism is vulnerable to timing attack, as the adversary can link a specific message between two parties by watching at the end points send and receive actions.
4. Sender anonymity from the first mix in the chain is not possible because he has to look for the receiver's address inside.
5. Mixes doesn't provide scalability because as the path becomes long, the packet length increases and more and more keys are required which creates overhead of latency and extra bandwidth requirement.

4.2 Onion Routing

Onion Routing is the mixture based architecture for private communication over the public network such as internet. It hides identities of both the sender and receiver not from each other but from the third party. It removes cookies from client machines which contain sensitive information. It works against traffic analysis, eavesdropping from inside onion routers as well as outside routers. [9]

Onion actually means, for the path of size n, the key for the last router is encrypted using the previous router and so on up to the first router. [13]

The communication goes through series of onion routers which starts by first the sender contacting one onion router which creates a path of onions to the receiver. In this sequence of onion routers, only neighbors have identity of each other. The process works as, the onion is sent to the first router which encrypts and each router in the sequence decrypts through their private keys extracting information such as key seed and next hop. Finally the data arrives at the destination in decrypted form because the previous router has removed the last encryption layer.

Currently onion routing is being used with all known application layer protocols. [10]

Drawbacks

1. Passive internal attacks are possible if at least two internal routers are compromised.
2. It is vulnerable to internal volume attack as, the adversary can analyze the different communication links to observe the size of the packet. If packet length of same size is found, the transmission path can be detected.
3. Onion routing can not protect against JavaScript, Java Applets and Active X controls if enabled by the browsers. [9]

4.3 Crowds Mixes

This mechanism protects against web servers during internet surfing. Basically crowds provide protection for web browsing privacy. This mechanism works as, a large group of diverse users requesting web services is created, this group is called crowd. [13] The request is initiated by one user and forwarded to crowd. On behalf of the user the crowed handles all the requests so that the server does not know where the request is coming from. Even the crowd members' collaborating each other can not know about each other. [11]

The requester first joins the crowd of other users, each crowd member is call jondo. When a jondo receives the traffic for the first time, it tosses a coin and diverts the traffic to another randomly chosen jondo or to its final destination. This jondo remembers the path of this traffic so that other subsequent traffic from the same user will be

forwarded on the same path. It will also take care of responses from the same jondo to its source. I.e. on jondo is responsible for the both side traffic in a path from specific source to destination.

Drawbacks

1. Local eavesdropper can identify the receiver if he is the last jondo and hence no sender anonymity against local eavesdropper.
2. Receiver anonymity is not possible.
3. Timing attacks because of its main usage for web surfing.
4. Once you find the initiator, you know the whole path between source and destination because this is fixed.
5. Internal DOS attacks are possible against crowd.
6. With firewalls, crowds can not be used because firewall normally does not support encryption.
7. Disclosure of information is possible because of the content of the message can be revealed to the jondos while preventing the identities.
8. Crowds can save you only from internal eavesdropper not from the global.

Communication overhead of Mixes

Anonymity comes with some cost of overhead in Mixes. Mixes basically rely on public key cryptography which is computationally very heavy process. If we look at the cost on the packet level, then we can easily figure out the cost associated with this mechanism.

Each packet goes through series of some extra steps, in Mixes, to successfully deliver it to its recipient and get back the reply as compared to normal delivery of packet in internet. In chaum mixes, at the sender, the packet is prepared by adding a random value R to the message. This R comes through some cost because we have to use some random function to generate this value which takes some time. It also adds to the payload of the packet which will take extra bandwidth and time for delivery. Then this packet is encrypted with public key of the Mix. This means that an extra header of encryption protocol is added to the packet at the network layer which adds to the load of packet. Now to send this packet on the communication line, this would cost extra bandwidth and time to deliver. Now this packet will be opened by the mix to find out the destination address, and to open it, the mix must decrypt the packet with its own private key. After getting the required fields from the packet, it is sealed again with the public key of the receiver and sends it to him. This decryption and again encryption at the mix consumes a lot of time and bandwidth of the communication channel.

The same process is repeated for the reply which simply doubles the overhead of delivery time and bandwidth. So, a packet during its complete journey from the sender to mix and from mix to the receiver and the reply coming back consumes extra bandwidth of packet load, computation overhead for cryptography such as encryption and decryption at sender, mix and receiver. Added to it is the extra cost of random value generation time and its delivery bandwidth. The mixing of messages and reordering also requires some extra time in the mix.

This overhead is almost same in chaum mixes and onion routing because both uses public key cryptography and also both uses intermediaries such as mix and onion routers. In these two mechanisms, as much as the traffic increases, it requires more extra bandwidth. And with the increase of path length, requires more cryptography resulting in huge communication overhead and delay. Crowds Mixes also use encryption which has the same kind of costs associated with it. And the random delivery to neighboring jondoes until it finds direct link to destination or to the web server is so time consuming. This is why we say that crowds is not scalable because as much as group members increase, the probability to find direct connection or web server becomes very low.

Hence secrecy over internet through mixes has extra bandwidth requirement and delay associated with it.

5. Mechanism used in DC-Nets

Like Mixes, several mechanisms use the methods based on DC-Nets like P5, Herbivore, Clique-Net.We will discuss them separately how they are helpful in secure AC.

5.1 P5 (peer to peer personal privacy protocol)

P5 used the methodology of Broadcast mechanisms to provide all three; sender, receiver and both sender-receiver anonymity. It does not need any special infrastructure to deploy in the system; else it works well independent of any kind of architecture provided and hence works over the current internet protocols.

At first, p5 starts with the broadcast ring of several users connected to it. Users send messages after the predefined intervals to everyone in the broadcast ring. If the message is intended for some user in the ring, he decrypts it and adds it into the forward messages so that nobody knows that the message has been decrypted. If the message is snot intended for the user he just adds it in his queue and sends to the other users

as well. In this way the data cannot be tempered and if the user has nothing to sent, he just generate random messages, so that nobody knows who the sender is .A bandwidth problem occurs when more users try to generate messages to the network.P5 combat with this situation by building a binary tree structure. A new user connects to the variety of broadcast rings but only listen on one node. Each message has a unique symbol indicating at what channel it is meant to travel [6]. So the message cannot take more than 3 to 4 hops to reach the destination if each user joins 2 or 3 rings, no matter how big the network is.

P5 provides anonymity for several different users communicating to each other at the same time as well as for two individual users. The opponent can able to monitor the traffic going on between the sending and receiving processes but unable to determine or read the contents of the encrypted messages sent or received between the two communicating parties. This kind of anonymity between sender-receiver is known as Unlink-ability property. The opponent cannot trace the message from a sender to a receiver because in the p5 every transmission of packets is encrypted hop-by-hop and even reading the message by one hop doesn't compromise the anonymity of the sender or receiver. The weakness of p5 is measured by the efficiency of the bandwidth because it does not provide high efficiency bandwidth, but it compensates it by controlling the bandwidth by a fixed amount of data to be sent. It is also possible to limit the number of people in a broadcasting system to increase the bandwidth utilization.P5 provides a complete control structure to communicate over the network anonymously.

Unlike other protocols ,many users can connect to each other at the same time without thinking about the bandwidth allocation in p5 ,as it cope well with the communication overhead problems without any hindrance between the communicating networks. Receiver anonymity is achieved in the DC-Net as they also use the public key infrastructure to broadcast to the entire group, but the sending time of the messages is not encrypted .Also in DC-Nets, one user sends at a time requiring extra bandwidth which is not the case when P5 protocol is used. So the sender-receiver anonymity is not at the same level in DC-Nets which makes P5 one level up then original DC-Nets.

The important addition in the P5 protocol is the noise packets to prevent passive attacks by the adversary because statistical attacks are possible by observing the streams of packets. P5 is invulnerable to different kind of attacks namely correlation attack which is as same as discussed above ,intersection attack,DOS attack ,Mob Attack.

The sending party for example X, has to encrypt every single packet except the noise packets, because there is a single public key decryption for the receiving entity. In P5 ,it depends on the user to go for the compromise between the anonymity and the bandwidth communication efficiency , so bandwidth efficiency is not an issue .It provides low latency, greater bandwidth and anonymity as well; unlike in plain unicast communication, which only provides the first two properties but doesn't provide anonymity[3]

5.2 Herbivore

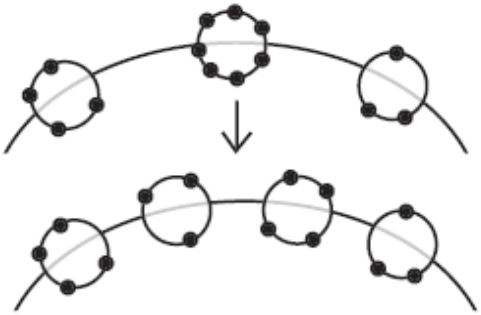
Herbivore is the anonymous communication system protocol that used the criteria of DC-Nets to implement anonymity in the networked applications. Herbivore provides scalability, as it has the capability to connect many users at a time, without disturbing the network efficiency. It also provides low latencies and high bandwidth when distributed over the internet. Herbivore hides the information of the two communicating entities even in the case of attacker with the wiretapping capabilities.

Herbivore used the basic operation of DC-Nets which only provides anonymity and doesn't work well with the scalability and efficiency problems. These problems are solved in Herbivore that's makes it efficient to use in many applications.

It broadens the basic DC-Net strategy with detecting tempering over the network which helps in long established connections. At the lowest level of herbivore, round protocol is used which regulate how bits are sent to the nodes. It also has some other features like reserving bandwidths for the nodes and detecting tampering by the attackers.

Herbivore uses *global topology control algorithm* to scale well with the networks as well as facing the malicious nodes. This algorithm divides the network into smaller cliques. Each clique will have a K node, and k is the predetermined constant that identifies the degree of the system .When the existing cliques in the network becomes overloaded; the new cliques are created automatically with the help of this algorithm. Global topology depends on entry control protocol .It

operates by assigning unique keys to the physical nodes. In every clique, there has to be at least k nodes and when the number of nodes in a clique becomes lesser, the nodes are automatically broadcast again over the network. Following the entry control protocol a node joins the herbivore network .The entry control protocol makes sure that the cliques are of equal size, so that there is a space for new nodes to join .It also limit the rates at which a new node joins the network. To enter in a network, the node first generates private/public key pair. This pair of keys prevents other malicious nodes by reusing the keys of other nodes. When the clique size hinders communication or drops below k, the clique is divided to two equally spaced cliques as shown in the figure.



Herbivores manage the clique sizes between k and 3k. Round protocol in herbivore ensures anonymity, enough bandwidth and no tampering over the network.

Herbivore is resilient to many different attacks e.g., Sybil Attacks in when the malicious nodes tries to join the cliques to slow down the rate of the data over the Network, herbivore introduces rate limiting node entry into the network to solve this kind of problem. It also prevents other attacks like DOS, by eliminating the nodes which are transmitting too slowly over the network to regain its bandwidth efficiency. Herbivore provides anonymity but not privacy but it is also the case with other protocols. The interesting thing to know is, when the number of users or hosts is increased in the network, the bandwidth and latency ratios are not infected by it [4].

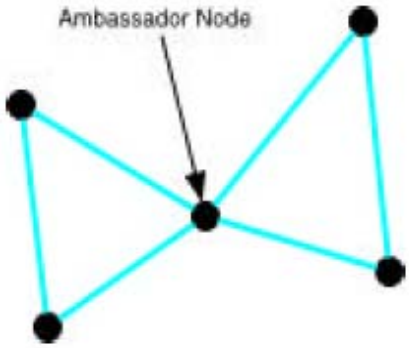
5.3 Clique Net

Clique Net protocol is using the basic of DC-Net mechanism to achieve anonymity and privacy.CliqueNet have a strong anonymity and challenges that no attacker can determine the sender of the data packets over the network even while wiretapping at any location in the network. It is peer to peer, scalable protocol just like Herbivore and guarantees strong anonymity, privacy and anticensorship.

In privacy ,the users release their identity to other participants of their choice but needs to hide from other users which are unauthorized to reveal that information e.g. whistleblowers,patients,witnesses etc.
Some transactions over the internet require strong Anonymity, in which a user hides its identity from all their parties in the network especially from the eavesdropper. The examples in this case are, casting a ballot in a voting booth and engaging in financial transactions.

DC-Nets provide anonymity but scale poorly with the increasing number of participants. The aggregate bandwidth of the DC-Net is O ($1/N^2$). In addition to strong anonymity and scalability, clique net has robustness feature in which it provides proofs to identify malicious nodes and excludes them from the network, if there is any.

Like Herbivore, clique net achieves scalability by automatically dividing the network into smaller dc-nets called **cliques**. Each clique has 3 to 5 nodes. **Single** client can be a member of other cliques, which joins them together in a network. That client is called an Ambassador node, as shown in the figure and is responsible for forwarding of messages from one clique to acting as a router.



So every node can communicate to every other node, no matter which clique it belongs to .The transfer of data occur in four phases of a round[6]. Each clique member reserves a portion of the other phases in the reservation phase. Then each client commits anonymously to the sent messages in the commitment

phase. If these two phases are in working, transmit authorization phase confirms that the nodes are ready for the data transmission. At the end, the data is transmitted by order of reservation in the Transmission phase. Everyone sends something to send in that round. Dummy Message is sent, if no meaningful message has to be sent [5].

7. Comparative Study

See the table in Appendix A

8. Evaluation

Evaluation Criteria

Different mechanisms are analyzed according to criteria such as overhead, latency, complexity, bandwidth required, and degree of anonymity.

See the table in Appendix B

9. Conclusion

Based on the evaluations and comparison, we have made a conclusion that both Mixes and DC-Nets are been using in the real world with the different mechanisms, but Mixes have the greater tendency to work in most of the network applications independently. Analyzing the three mechanisms we found that Chaum Mixes will be used effectively while in the field of web applications e.g. web surfing crowd mixes are the useful anonymous system to be used.

In DC-Nets ,we analyzed by different perspectives that Clique-Net can be used in the network applications more efficiently because along other paremeters,it is also provide privacy to the anonymous communication which makes it a bit superior from other protocols .

10. References

1. www.microsoft.com/security/glossary.mspx (Anonymity)
2. http://www.everything2.com/index.pl?node_id=1064494 (DC problem scenario)
3. www.cs.umd.edu/projects/**p5**/**p5**.pdf
4. www.cs.cornell.edu/People/egs/papers/**herbivore**-tr.pdf
5. www.cs.cornell.edu/people/egs/papers/**cliquenet**-iptp.pdf
6. www.cs.umd.edu/~bobby/anoncomm/thesis.pdf (p5 and cliquenet)
7. Design and Analysis of an anonymous Communications Channel for the Free Haven Project By Michael J. Freedman Anonymity Lecture notes, by Tzachy Reinman, computer science and Engineering school the Hebrew university Jerusalem.
8. Crowds home page http://www.research.att.com/projects/crowds/
9. The Onion Routing home page. http://www.onion-router.net/
10. M. Reed, P. Syverson, and D. Goldschlag. " Anonymous Connections and Onion Routing", IEEE Journal on Selected Areas in Communications, vol. 16, no. 4, May 1998, pp. 482-494.
11. M. K. Reiter and A. D. Rubin. "Crowds: Anonymity for Web Transactions", ACM Transactions on Information and System

13 Anonymity Lecture Notes, Tzachy Reinman, Computer Science and Engineering School, the Hebrew University, Jerusalem.

14 Claudia D¶³az and Bart Preneel K.U.Leuven Dept. Electrical Engineering-ESAT/COSIC Kasteelpark Arenberg 10, B-3001 Leuven-Heverlee, Belgium claudia.diaz@esat.kuleuven.ac.be, bart.preneel@esat.kuleuven.ac.be http://www.esat.kuleuven.ac.be/cosic/