NOVEL NETWORK DELAY BASED SIDE CHANNEL ATTACK

By

Mah Noor Javed

Sania Atif

Submitted to Faculty of Department of Computer Software Engineering National University of Sciences and Technology, Islamabad in partial fulfilment for the requirements of a B.E Degree in Computer Software Engineering, June 2019

In the name of Allah, the Most Beneficent, the Most Merciful

ABSTRACT

RTT Based Website Tracking- Novel Network Delay Based Side Channel Attack

Leakage of information through side channels has become number one security threat to the web traffic that are mainly. Presently Side channel attack mainly focusses on packet length, timing of packet and internet object length. Though , we discovered that web traffic that are encrypted can also leak data through delay of the network between a client and the websites he or she visits. Inspired through this observation, we performed a side channel attack to calculate the probability that the client visited the particular website. The attacker can make use of the techniques using pattern identification to identify websites by calculating mean and variance of the RTT of each packet between a user and websites.

The idea of the project Novel Network Delay Based Side channel Attack is to develop a web based application to predict which website the user is accessing through squid proxy server .This document is meant to outline the features and requirements of project Novel Network Delay Based Side channel Attack, to give the right direction to the developers and a software validation document for the client.

A web based application has been developed for the Military, government organization, federal agencies, and multinational companies followed by following modules normally developed in a virtual environment, the attacker will run an exploit on the victim's router , after the attack the attacker will be able to monitor the web traffic of the client going to the web server via squid in a can proxy. Different statistical analysis will be perform on the traffic and the attacker will be able to calculate the probability that which website the user is accessing based on the round trip time of each packet.

CERTIFICATE FOR CORRECTNESS AND APPROVAL

It is certified that work contained in the thesis RTT Based Website Tracking- Novel Network Delay Based Side Channel Attack carried out by Mah Noor Javed and Sania Atif under supervision of A/P Waseem Iqbal for partial fulfillment of Degree of Computer Software Engineering is correct and approved.

Approved by
Waseem Iqbal
Assistant Professor
Department of IS
MCS, NUST

Dated: 1 May, 2019

DECLARATION

No portion of the work presented in this dissertation has been submitted in support of another award or qualification either at this institution or elsewhere.

DEDICATION

To our parents, without whose support and cooperation, a work of this magnitude would not have been possible. To our supervisor, A/P Waseem Iqbal who has given us great support and valuable suggestions throughout the implementation process.

ACKNOWLEDGEMENTS

There is no success without the will of Allah Almighty. We are grateful to Allah, who has given us guidance , strength and enabled us to accomplish this task. Whatever we have achieved, we owe it to Him, in totality. We are also grateful to our parents and family and well- wishers for their admirable support and their critical reviews. We would like to thank our supervisor A/P Waseem Iqbal for his continuous guidance and motivation throughout the course of our project, without his help, we would not have been able to accomplish anything.

Table of Contents

Table of Contents

List of Figures

List of Tables

Chapter 1 Introduction

1.1. Overview

Leakage of information through side channels has become number one security threat to the web traffic that are mainly. Presently Side channel attack mainly focusses on packet length, timing of packet and internet object length. Though , we discovered that web traffic that are encrypted can also leak data through delay of the network between a client and the websites he or she visits. Inspired through this observation, we performed a side channel attack to calculate the probability that the client visited the particular website. The attacker can make use of the techniques using pattern identification to identify websites by calculating mean and variance of the RTT of each packet between a user and websites.

1.2. Problem Statement

Nowadays a very big problem that people are facing especially in large organization is to keep the track of all employee's web activities. Cybercrime has created a major threat to those who use the internet. Major cybercrimes are done online. And it is important for any agency or an organization to keep an eye on all their employees and must have complete knowledge of which websites they are accessing .

1.3. Approach

A network is setup in a virtually controlled environment where an attacker will perform a side channel attack on the client system who is using the internet via squid in a can proxy. The attacker will exploit the access point with which the client is connected. The attacker will be able to monitor the client's web traffic through a common access point, and will determine the website the client is accessing using through statistical analysis.

1.4. Scope

The intended scope of the project is to perform Side channel analysis on the client's system from where a website is accessing over the network .The side channel attack will allow the attacker to estimate which website is being accessed from the client's system. This can be used by Military, government organization, federal agencies, and multinational companies. We will also state the countermeasures in order to avoid such attacks and highlight the security risk.

1.5. Objectives

The objective of this project is to give the probability that an attacker correctly recognizes websites accessed by a victim using the round trip time of each packet.

During the course of this project, all the aspects of software engineering are covered i.e. survey and feasibility analysis, requirement gathering, architectural and detailed design, implementation and testing along with documentation (SRS, SDS, Test Document, Final Report and User manual). Students are also expected to develop extensive knowledge and technical skills in the following fields:

1. Basic knowledge of networking.
2. Python programming language.
3. Things related to security.

1.6. Deliverables

	Tasks	**Deliverables**
	Literature Review	Literature Survey
	Requirements Specification	Software Requirements Specification document (SRS)
	Detailed Design	Software Design Specification document (SDS)
	Implementation	Project demonstration
	Testing	Evaluation plan and test document
	Training	Deployment plan

	Deployment	Complete application with necessary documentation

Table 1-1 Deliverables

1.7. Overview of the Document

This document shows the complete working process of our application novel network delay based side channel attack. It starts off with the literature review which shows past work done in similar field, requirement analysis of the system, system architecture which highlights the modules of the software and represents the system in the form of component diagram, Use Case Diagram, Sequence Diagram and general design of the system. Then it will move on to discuss the detailed Description of all the components involved. Further the dependencies of the system and its relationship with other products and the capacity of it to be reused will be discussed. At the end test cases and any future work proposal has been presented.

1.8. Document Conventions

Heading are prioritized in a numbered fashion, the highest priority heading having a single digit and subsequent headings having more numbers, according to their level. Font used is Times New Roman. All the main headings are of size 18 and bold. All the second level sub-headings are of size 16 and bold. All the further sub-headings are of size 14 and bold. All references in this document are provided where necessary, however where not present, the meaning is self-explanatory. All ambiguous terms have been clarified in the glossary at the end of this document.

1.9. Intended Audience

This document is designed for:

1. **Developers:** (Project Group)

   so that it will be sure that they are growing the proper venture that fulfills the requirements furnished in this report.
2. **Testers:** (Project Group, Supervisor)

   To have a correct function and features to respond according to the requirement.
3. **Users:**

with the intention to get acquainted with the idea of the task and how to use/reply in failure situations and endorse other functions that would make it even more purposeful.

4. **Documentation writers:** (Project Group)

   To know what features and in what way they have to explain. What technologies are required, how the system will respond in each user's action, what possible system failures may happen and what are the solutions to all those failures etc.

5. **Project Supervisor:** (A/P Waseem Iqbal)

   This document will be used by the project supervisor to check whether all the requirements have been understood and in the end whether the requirements have been implemented properly and completely.

6. **Project Evaluators:** (CSE Dept. MCS)

   In order to know the scope of the project and evaluate the project throughout the development for grading.

Chapter 2 Literature Review

Detailed description of projects previously carried out in this context will be discussed in this section.

The main concept of side channel attack in this project is to observe the activity of the person who is under an attack and determine which website is being access over the internet there are different method through which we can determine which website is accessed, like the attack can be done by observing the size of the packet or calculating the round trip time of the packet from the client's system . Following are the remarkable name of those who did tremendous work in detecting the website being accessed by the victim.

Zhu et al used common data for the similitude estimation. Levine et al used a cross relationship system. Research demonstrated that the attacker could deduce delicate data from scrambled system traffic by looking at examples of bundle size and timing. For instance, Song et al used the packet inter arrival timing in SSHv1 associations with keystroke patterns and at last break the passwords. Sun et al. explored the sizes of HTML objects transmitted over SSL associations and could recognize the pages dependent on the number and size of items in each scrambled HTTP connection. Liberatore and Levine analyzed parcel size of HTTP traffic transmitted over the steady connections of SSH that could measurably recognize the webserver pages.

Chapter 3 Software Req. Specification (SRS)

3.1. Introduction

The introduction of the SRS (Software Requirement Specification) highlights an entire SRS with purpose, scope, acronyms,references, , definitions, abbreviations. The objective of this document is to give detailed information of Novel Network delay based side channel attack by defining in detail the problem statement. The detailed requirements of the Novel Network delay based side channel attack are provided in this document.

3.1.1. Purpose

This document includes the software requirements specifications of the Novel Network delay based side channel attack release version 1.0. The purpose of this document is to present a detailed description of the side channel attack. It will explain the purpose and features of the system, the interfaces of the system, what the system will do, the constraints under which it must operate and how the system will interact with external system. This document is intended for the both the stakeholders and the developer of the system.

3.1.2. Project Vision

A web based application has been developed for the Military, government organization, federal agencies, and multinational companies.

For	Military, government organization, federal agencies, and multinational companies to keep the track of all the activities of their employees.
What	A web based application to calculate the probability that a client is accessing that particular website using side channel analysis.

The	Novel Network Delay based side channel attack.
Is	A Web based application.
That	Perform statistical analysis on the round trip time of each packet and predict the website visited by the client.

Table 3-1Project Vision

3.2. Overall Description

3.2.1. Product Perspective

Novel Network Delay based side channel attack will help organization and agencies to monitor the web activity of their employee without having the physical contact with their system using side channel attack. The attacker will be able to identify which website the client is accessing through the common access point.

3.2.2. Product Features:

Novel Network Delay based side channel attack helps the agencies and organization to predict the website the client is accessing via squid in a can proxy through common access point. Main features of the application are listed below.

1. Login
2. Signup
3. Search number of interfaces available on the access point.
4. Select interface on which attack is to be perform.
5. Click on start capturing live packet to monitor the live ingoing and outgoing traffic of the client.
6. Perform statistical analysis on the captured traffic and calcuate mean and standard variance.
7. View result to show which website is being accessed by the client.

3.2.3. User Classes and Characteristics

Following are user classes and their brief description.

Client (Occasional user)

Client is the one who will under the observation.

Attacker(Regular user)

The attacker will be one who will carried out the attack on client's system.

Developers

The developers will use this at the developing time and at the time of any defect occurred in the product during maintenance.

Documentation Writers

The document can serve as a future reference for other versions of the SRS.

3.2.4. Operating environment

The product shall be operating in an android environment. It shall be compatible with version 4.3 (Jelly Bean) and all the higher versions of android.

Hardware

1. Computer: To run the application.
2. Network Infrastructure: To connect to the internet.
3. Local Network(router, access point)

Software

1. Squid in a can
2. Dhcpd
3. Docker
4. Tcp dump
5. Tcp trace
6. Python
7. Numpy
8. Scikit learn

3.2.5. Design and Implementation Constraints

Constraints of the system are given below:

1. The attack is not possible until the client and attacker are on the same access point.
2. The implementation is being done in a controlled environment .
3. In order to calculate the probability of a specific website we need to passively monitor the client's web traffic for at least a month prior to the attack.
4. In our controlled environment we have limited number of servers.
5. Slight changes will have to be made in the system if a different proxy server is being used from the client side is being used in our controlled environment. This is because the level of encryption is dependent on the proxy being used .
6. The final results will be an approximation / probabilities rather than exact values.

3.2.6. User Documentation

A user manual will be provided to the attacker who wishes to perform an attack in which separate instructions will be given. It will include the details of the system's working. Help documents will also be a part of the system.

The project report will also be available for the attacker which will highlight the system features, working and procedures.

3.2.7. Assumptions and Dependencies

1. Overall performance of the product will depend on the hardware infrastructure and network speed.
2. User must know the language and User Interface for the better performance of the product.
3. Limitations of the application must be kept in mind by the user.
4. User must be aware of the basic knowledge of networking and the local network.

3.3. External Interfaces Requirements

3.3.1. User Interfaces

1. User interface will be displayed on the desktop screen.
2. Interface will be user friendly and the standard English-US will be used.

3.3.2. Hardware Interfaces

1. This product requires functional PCs in order to work properly.
2. The client and the attacker should be connected to the same access point.
3. A proxy server should be used while accessing the website.
4. The computer should have a web browser with at least the minimum hardware

requirements.

3.3.3. Software Interfaces

1. Application should be able to run on a Windows based platform using Microsoft Windows XP or newer versions.
2. The proxy server in the controlled environment should be running on a linux.
3. A squid in a can proxy is being used in this case.

3.3.4. Communications Interfaces

1. The system will capture squid tunneled traffic.
2. The attacker and a client should be on the same local network and same access point.

3.4. System Features

This section describes in detail the system features of the Novel network delay based side channel attack.

3.4.1. LogIn

Description and priority

This feature allows the attacker to gain the access to use the application
Its priority will be high as without this feature the application will not allow the user to access the admin privilege.

Stimulus/Response Sequence
Data flow:

Basic Data Flow:

1. The attacker will logs in the application by entering the username and a password.
2. If the password and username matches he will gain the access to use the application.

Alternate Data Flows:
Alternative Data Flow :

1. If the user is signing for the first time ,he will be asked to create a new account.

Functional requirements

REQ-1: The system shall allow the attacker to login if password and a username matches.

3.4.2 Select an Interface:

Description

Access point have number of interfaces, an attacker will be allowed to select any interface to monitor the traffic.

Stimulus/Response Sequence
Data flow:

Basic Data Flow:

1. The system will display the interfaces available on the network
2. The attacker is now able to select any interface to monitor the traffic.
3. Once the attacker selects a particular interface the system will then start capturing packets.

Functional requirements

REQ-2: • The system shall display the user the number of interface available on the network. **Maintenance of Database**

Description

Database is a repository that will store the details of captured packets.

Stimulus/Response Sequences

Data flow:

Basic Data Flow:

1. The attacker will selects the interface to be monitored.
2. The system will start capturing the packets.
3. Along with the packet capturing the system will also start storing those details of captured packets in the database.
4. The capturing of packets and storing it In a database will be done in parallel.

Functional Requirements

REQ-3: • The system shall capture the packets and store them in database simultaneously.

3.4.3 Analyzing the Packets:

Description

Analyzing the captured packets and extracting the RTT.

Stimulus/Response Sequences

Data flow:

Basic Data Flow:

1. Once the database is completed ,the attacker will extract the RTT for each sites

Functional Requirements

REQ-4: The system shall allow the attacker to observe the captured packets.
REQ-5: The system shall allow the attacker to extract the RTT for each individual sites.

3.4.4 Statistical Analysis:

This feature allows the attacker to perform various mathematical calculations on the data gathered from the captured traffic.

Description

We have multiple mathematical theorems related to sample mean and mean variance . the attacker can apply those on the captured traffic in order to analyze it.

Stimulus/Response Sequences

Data flow:

Basic Data Flow:

1. User captures the traffic.
2. The user navigates to select which analysis he would like to perform on the data.
3. The user performs the analysis
4. The system visualizes the analysis on the screen

Functional Requirements

REQ-6: The system shall allow the user to perform complex mathematical calculations on the network traffic on the press of a button.

3.4.5 Statistical Analysis:

This feature allows the user to view the results of the analysis performed.

Description

After the statistical analysis the system will display the results that will be used by the attacker to determine the probability which website is being accessed.

Stimulus/Response Sequences

Data flow:

Basic Data Flow:

1. The system will display the result in visual manner.

Functional Requirements

REQ-7: The system shall allow the attacker to view the results of analysis.

REQ-7: The system shall allow the attacker to make conclusion based on the analysis.

3.5. Other Nonfunctional Requirements

3.5.1. Safety Requirements

The system must be fast and responsive to user actions. However while working with large
data the system may lead Application to become unresponsive

3.5.2. Security Requirements

The system shall be secure. The user account must be protected with username and password
. The identity of the attacker will be preserved .

3.5.3. Performance Requirement

The system should be fast in terms of performance. The system shall respond to the user request in less than 400 ms depending upon the congestion in network.

3.5.4. Software Quality Attributes

Reliability

1. The system should promote reliability to the user.
2. The system will run stably with all the features mentioned above available and executing perfectly.
3. It should be completely tested and debugged..
4. All exceptions should be well handled.

User Friendliness/Simplicity

The system should have a graphical user interface with user friendly options.

Chapter 4 Design and Development

4.1. Introduction

This design document covers all our functional requirements and demonstrates how they interrelate with each other abstractly. The low level design also illustrates as to how we have been implementing and how we are going to implement all of these requirements. This low level design for the time being does not address any non-functional requirements that our system has and that have been mentioned in the SRS Document.

4.2. Purpose

The motive of this software design document is to deliver a portrayal of the layout of our machine suitable sufficient to permit for software program improvement to proceed with an know-how of what is to be constructed and the way it is projected to be advanced. This software layout report offers statistics crucial to get an outline of the information for the software and the device to be built. The motive of this file is to present a layout view and particular description of Network delay based side channel attack . it's going to give an explanation for the cause, features, interfaces, what the device will do, its complete strategies in detail, the limitations under which it have to perform and the way the gadget will react to inputs and what is going to be its outputs. This report is supposed for both the stakeholders and the system developers.

4.3. Project Scope

The project scope is to perform Side channel analysis on the client's system from where a website is accessing over the network .The side channel attack will allow the attacker to estimate which website is being accessed from the client's system. This can be used by Military, government organization, federal agencies, and multinational companies. We will also state the countermeasures in order to avoid such attacks and highlight the security risk

4.4. System Architecture Description

In this section, the overall architecture of the system is discussed, including the introduction of various components and subsystems. It is mainly supported by System Architecture diagram which shows an insider's perspective of the system by describing the high level software components that perform the major functions to make the system operational.

4.4.1. Structure and Relationships

This section ponders upon the interrelationships and dependencies among various components. It is mainly described by a diagram which is further augmented by explanatory text. UML Class diagram also helps us understanding the system structure.

System Block Diagram

The diagram(s) show the higher level description of the application(s), generic working of the application(s) and interaction with the user.

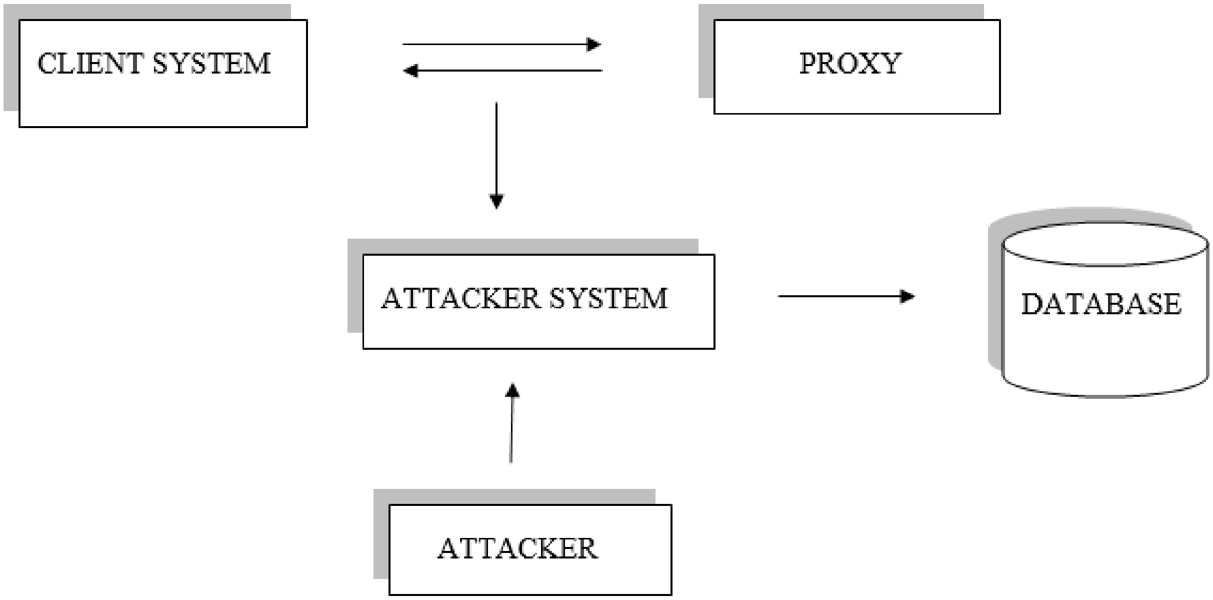

Figure 4-1 System Block Diagram

Explanation of System Block Diagram
The system will be architected mainly in four fundamental modules “Client system”, “Proxy”, “Attacker”, “Attacker system”, and the “Database”. Abstract diagram highlights the overall system, from system being accessed by the user till the database processing.. The sub modules of the Abstract diagram are further elaborated below.

Users
The victim is the one who will under the observation. The victim will be using the client system. As it is obvious from the above diagram that client is connected to the internet through the proxy.

Proxy
A SOCKS based OpenSSH proxy is being used in this case. The proxy server will send request to the internet on the behalf of the client.

Attacker

Attacker will be the one who is monitoring the ingoing and outgoing traffic of the client system through encrypted proxy tunnel. The attacker will then apply statistic on the observation to detect which website is being accessed by the client remotely.

Attacker system

Attacker system is the tool which is being used by the attacker for monitoring and observing the client system's ingoing and outgoing traffic.

Database

A database stores all the traffic data of a client system which will categorize further on the basis of the attacker need.

User View (Use case diagram)

Figure 4-3 shows course of events that take place when an actor (user and other allowed interactions) interact with the system. It shows the main functionality of the application available for a normal user and how it interacts with those.

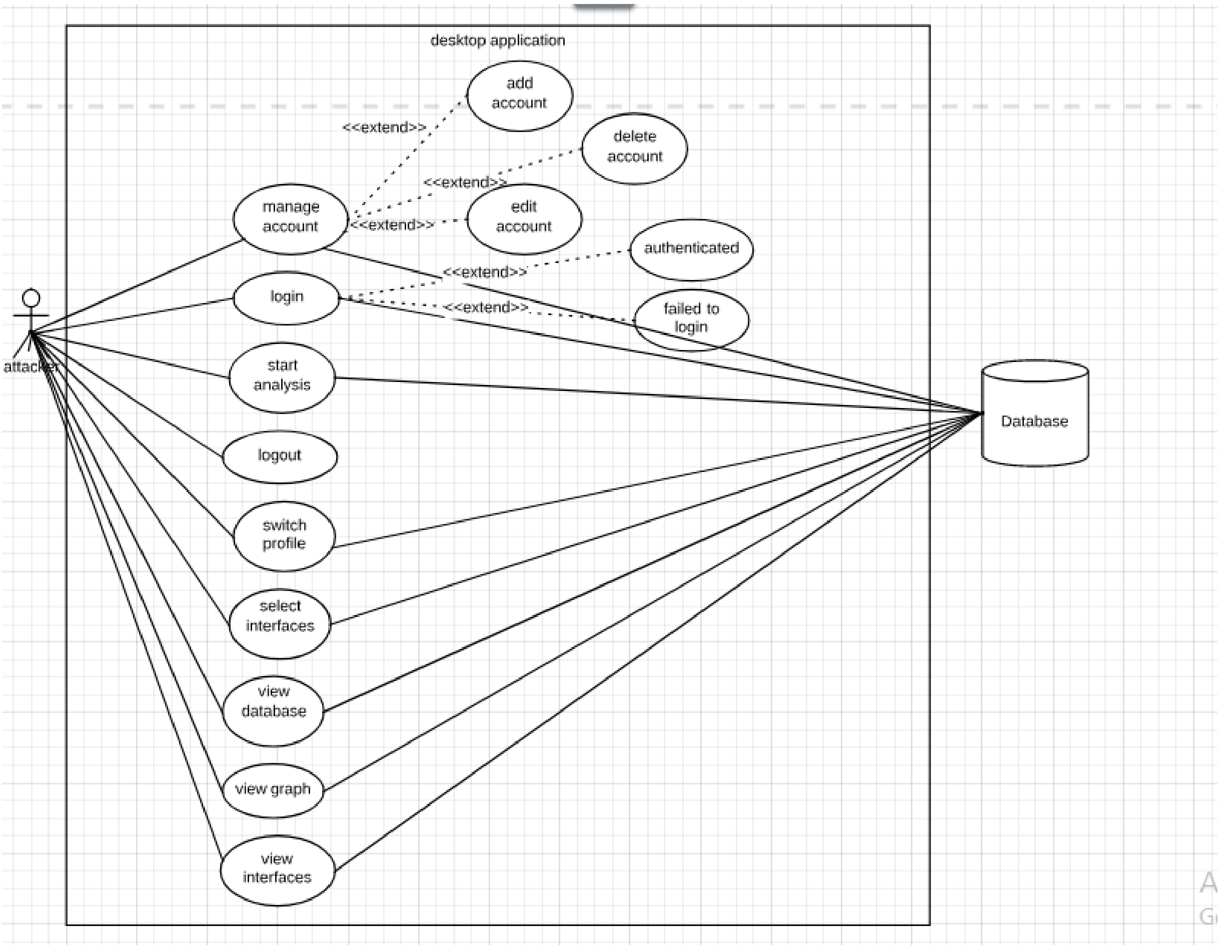

Figure 4-2 System Use Case Diagram

Actors:

Primary Actors:

1. User(Attacker)

Secondary Actors:

None

Use Cases:

1. Login
2. Manage Accounts
3. View interfaces

4. Select interfaces
5. View graph
6. Start analysis
7. Switch profile
8. View database
9. Log out

Use Case Description:

Use cases shown in the figure above are described below.

Use Case 1

Use Case ID:	1		
Use Case Name:	Manage Accounts		
Actors:	Attacker		
Created by:	Mahnoor	Last Updated by:	Mahnoor
Date Created:	7/01/2019	Date Last Updated:	7/01/2018
Description:	Attacker has to log in to the system and create accounts to view available interfaces. This will include creating, editing and deleting accounts.		
Preconditions:	Attacker has to log in.		
Post conditions:	Changes in account must be updated in the database.		
Normal Flow (Primary Scenario):	1. The Attacker will enter, modify or delete the accounts to view interfaces. 2. Changes will be updated in the database.		
Alternative Flows:	1. An error is encountered during the modification of database. 2. Proper functionality of the database will be checked.		

Table 4-1 Use Case 1

Use Case 2

Use Case ID:	2		
Use Case Name:	Login		
Actors:	Attacker		
Created by:	Mahnoor	Last Updated by:	Mahnoor
Date Created:	7/01/2019	Date Last	7/01/2019

		Updated:	
Description:	Attacker tries to login to the system.		
Preconditions:	Attacker has to open the login page first.		
Post conditions:	If the use case was successful, the attacker is now logged into the application . If not, the state of the application remains unchanged.		
Normal Flow (Primary Scenario):	1. The application requests that the attacker enter his or her name and password. 2. The attacker enters his or her name and password. 3. The attacker validates the entered name and password and logs the attacker into the application.		
Alternative Flows:	If inside the Basic flow, the attacker enters an invalid password, the application will show an error message. The attacker can pick out either to go back to the beginning of the basic flow or cancel the login, at which point the use case ends.		

Table 4-2 Use Case 2

Use Case 3

Use Case ID:	3		
Use Case Name:	View interfaces		
Actors:	Attacker		
Created by:	Mahnoor	Last Updated by:	Mahnoor
Date Created:	7/01/2018	Date Last Updated:	7/01/2018
Description:	After successful login in the application attacker will be allowed to view the available interfaces.		
Preconditions:	Attacker has to log in		
Post conditions:	-		
Normal Flow (Primary Scenario):	1. The attacker will be allowed to view the interfaces. 2. list of interfaces will be displayed on screen.		
Alternative Flows:	1. An error is encountered during the available interface .		

Table 4-3 Use Case 3

Use Case 4

Use Case ID:	4		
Use Case Name:	Select interfaces		
Actors:	Attacker		
Created by:	Mahnoor	Last Updated by:	Mahnoor
Date Created:	7/01/2018	Date Last Updated:	7/01/2018
Description:	After all interfaces has been displayed the attacker will be allowed to select the desire interface of choice.		
Preconditions:	Attacker has to log in.		
Post conditions:	1. The outgoing and ingoing traffic of the selected interface will start display on the screen. 2. Traffic will be store in the database.		
Normal Flow (Primary Scenario):	1. The attacker will login to the profile. 2.Traffic will start displaying on the screen.		
Alternative Flows:	-		

Table 4-4 Use Case 4

Use Case 5

Use Case ID:	5		
Use Case Name:	Start analyzing		
Actors:	Attacker		
Created by:	Mahnoor	Last Updated by:	Mahnoor
Date Created:	7/01/2018	Date Last Updated:	7/01/2018

Description:	After selection of a desired interface the attacker will be allowed to start analyzing the outgoing and ingoing traffic from that interface.
Preconditions:	Attacker has to log in.
Post conditions:	Analyzed data i.e outgoing and ingoing traffic of a selected interface will be then store in the database for further use.
Normal Flow (Primary Scenario):	1. The attacker will login to the profile. 2. The application will start analyzing the traffic.
Alternative Flows:	-

Table 4-5 Use Case 5

Use Case 6

Use Case ID:	6		
Use Case Name:	View Database		
Actors:	attacker		
Created by:	Mahnoor	Last Updated by:	Mahnoor
Date Created:	7/01/2018	Date Last Updated:	7/01/2018
Description:	Attacker can view the database at any time ,which includes all the analyzed traffic of different interfaces.		
Preconditions:	1. Attacker has to log in.		
Post conditions:	Database of interface traffic will be displayed.		
Normal Flow (Primary Scenario):	1. the attacker will login to the profile and select the database to view. 2. database will be displayed.		
Alternative Flows:	-		

Table 4-6 Use Case 6

Use Case 7

Use Case ID:	7		
Use Case Name:	View graph		
Actors:	Attacker		
Created by:	Mahnoor	Last Updated by:	Mahnoor
Date Created:	7/01/2018	Date Last Updated:	7/01/2018
Description:	Attacker can view the graph of analyzed traffic.		
Preconditions:	1. Attacker has to log in.		
Post conditions:	Graph of traffic of specific interface will be displayed.		
Normal Flow (Primary Scenario):	1. The attacker will login to the profile 2. The attacker will view the graph of selected interface.		
Alternative Flows:	-		

Table 4-7 Use Case 7

Use Case 8

Use Case ID:	8		
Use Case Name:	Log Out		
Actors:	Attacker		
Created by:	Mahnoor	Last Updated by:	Mahnoor
Date Created:	7/01/2018	Date Last Updated:	7/01/2018
Description:	Attacker attempts to log out of the system.		
Preconditions:	Attacker has to be logged in first.		
Post conditions:	The attacker will be logged out and sent to the login page.		
Normal Flow	1. User clicks the Sign Out button.		

(Primary Scenario):	2. The attacker is signed out and sent to the Login Screen.
Alternative Flows:	-

Table 4-8 Use Case 8

Sequence Diagrams

Following sequence diagrams show the sequence of activities performed in application.

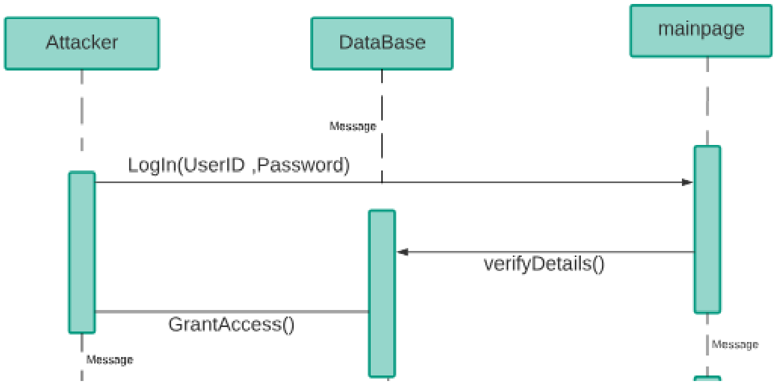

Figure 4-3 Sequence Diagram

Figure 4-4 shows the sequence of events as a attacker logs into his or her profile to view the homepage (interface)

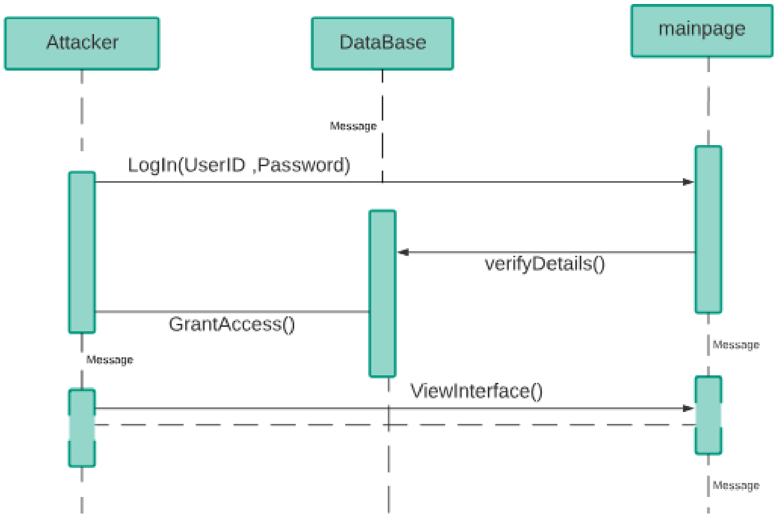

Figure 4-4 Sequence Diagram

Chapter 1 Figure 4-5 shows the procedure of viewing interfaces

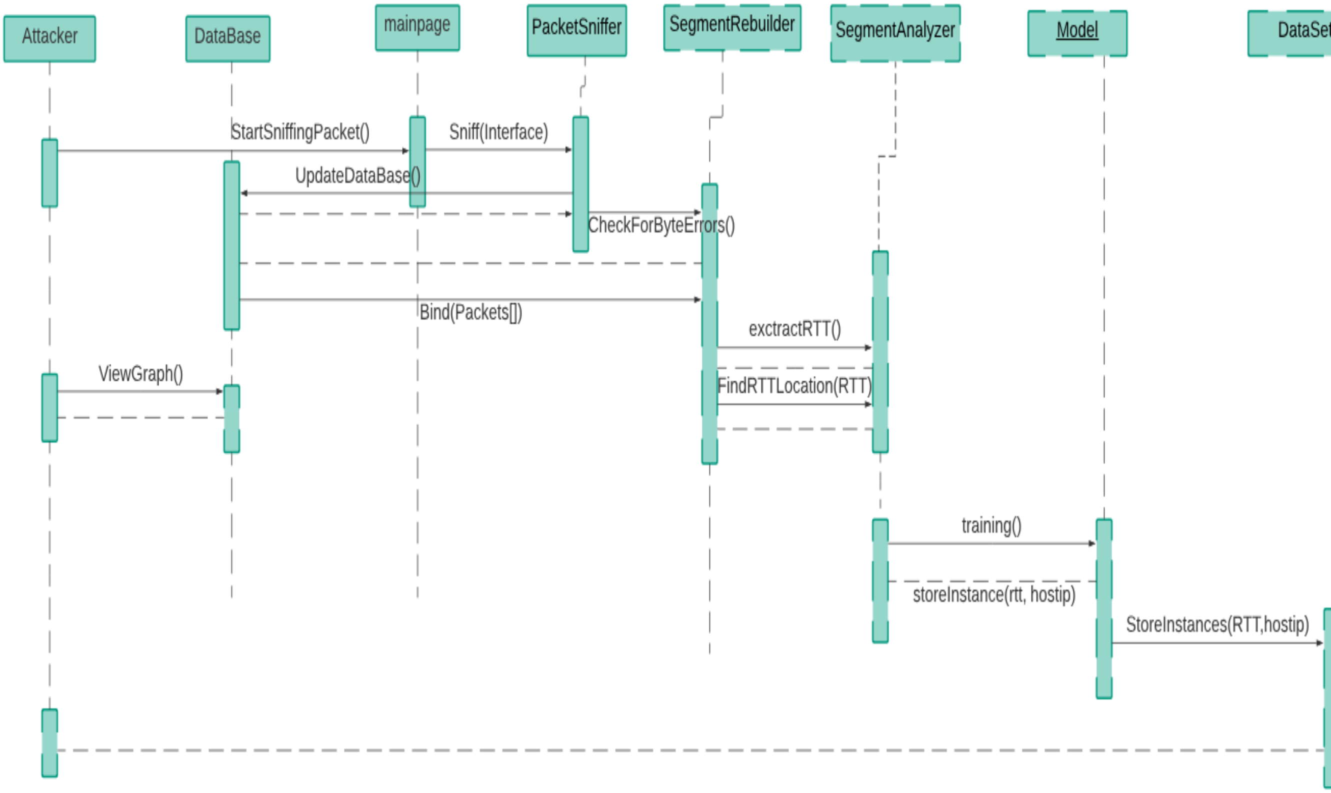

Figure 4-5 Sequence Diagram

Figure 4-6 shows how attacker gathers the data from a client system.

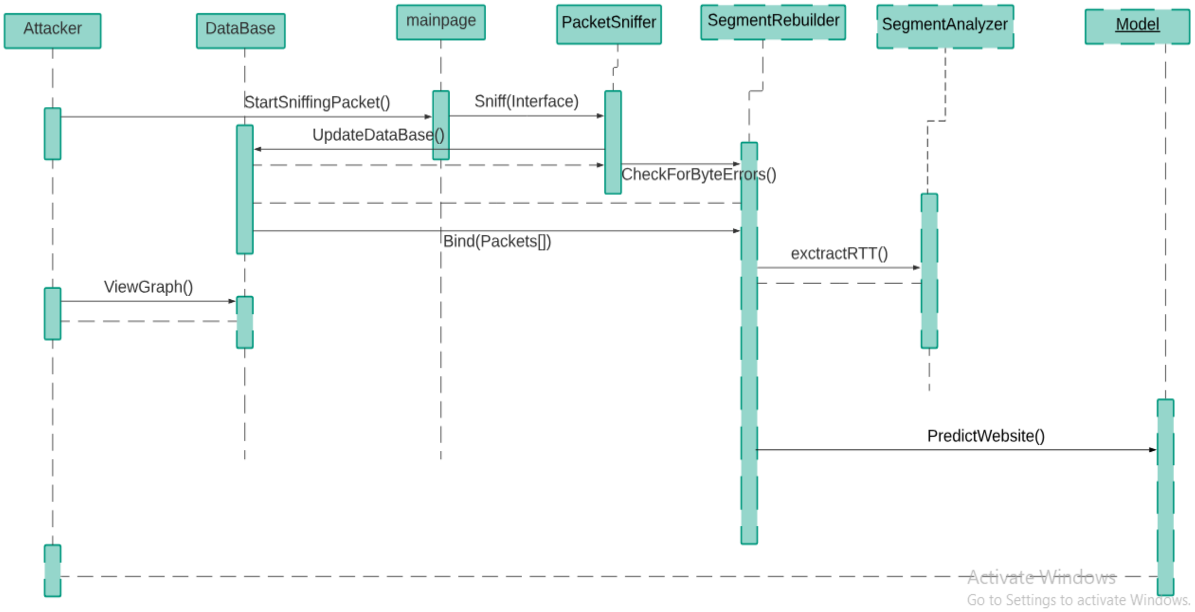

Figure 4-6 Sequence Diagram

Figure 4-7 shows how an attacker performs the real time attack on client system.

Implementation View (Class Diagram)

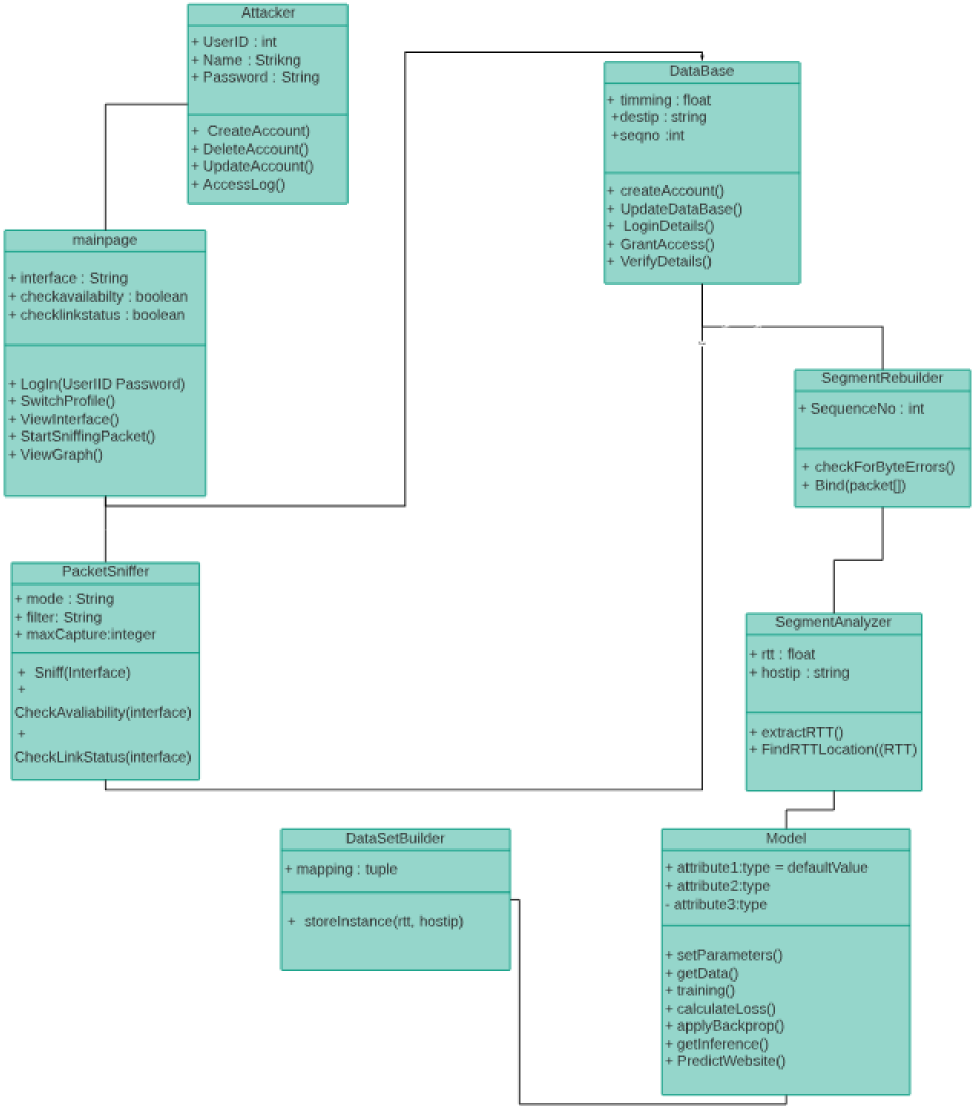

Figure 4-7 Class Diagram

<u>**Class's description**</u>

Class Name	Description
System	System class contains the origin for the function application has to perform. It is the main class which will be acting as a gateway to all the other classes
User	User class contains the the functions related to creating deleting and modifying the user account.
Interface handler	Interface handler contains all the option available for the attacker .it contains the main functions that the application will perform for example viwing the interface, selecting interface, data capture, storing to database and statistical analysis.
Data capturing	This class will perform the function of the application i.e data capturing of the interface and will store the data in the database by callig function of the other class.
database	Database will store all the data for further use.
Statistical analysis	In the statistical analysis class the function will do different stats in the collected data and deduce different result .
display	Display class will display the data in the screen i.e output

Table 4-9 Class Description

Dynamic View (Activity Diagram)

In activity diagram, the dynamic view of the system is shown. All the activities are shown.

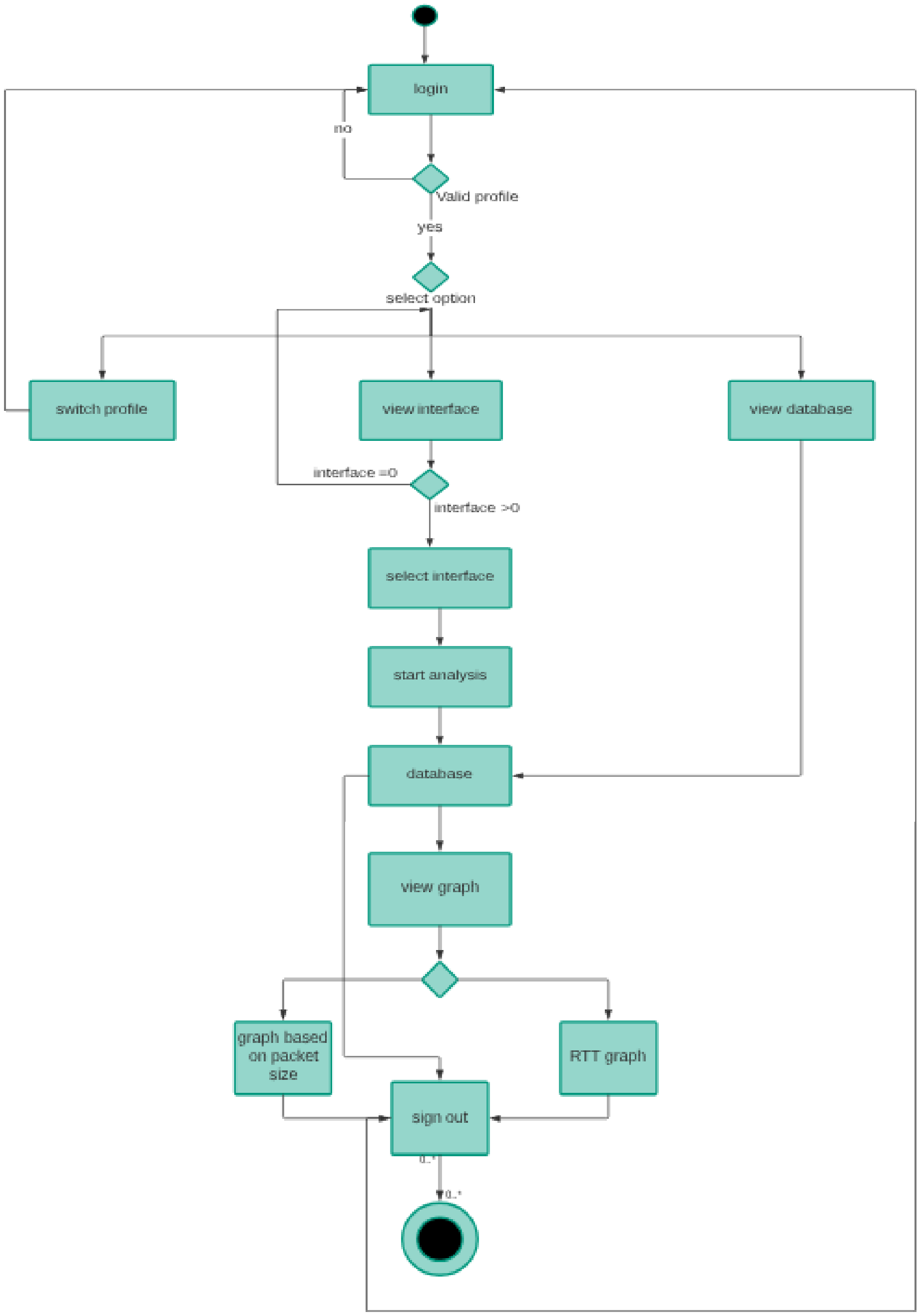

Figure 4-8 Activity Diagram

4.4.2 Detailed Description of Components

This section describes in detail all components novel network delay based side chaneel attack

Network packet sniffer

Identification	**Network Packet Sniffer**
Type	Utility
Purpose	• To gather raw network packets from the network stream going between proxy and victim client.
Function	• Sniff(clientID) • checkAvaliability() • checkLinkStatus()
Subordinates	• Victim to client proxy connection link
Dependencies	Network Utility programs to sniff eg: OpenWRT Any other Linux based router framework can be chosen such as DD-WRT however openWRT is chosen because of vast set of available tools and its purely open source nature
Interfaces	• To check availability of connection, network information • To extract sniffed packet
Resources	OpenWRT compatible router
Processing	• Analyze network link status • Check ongoing traffic in each link • Identify desired victim's traffic • Gather all ongoing packets on that particular link • Inspect each set of packets for errors
Data	Packet Link Info Sniffed Packet

Table 4-10 Network packet sniffer

Segment rebuilder

Identification	**Segment Rebuilder**
Type	Class
Purpose	• Analyze gathered raw network packets to build a transport layer segment to achieve the goal of extracting timing info of the end to end round trip • To bind together each packet body following the network layer protocols to rebuild the transport layer segment, this must be done vigilantly since a single error in a single byte will not allow the transport layer segment to be rebuilt properly because of the checksums
Function	• Bind(packet[]) • checkForByteErrors()
Subordinates	Sniffing Utilities
Dependencies	OpenWRT packet sniffer such as TCP Dump
Interfaces	• Takes Packets • Outputs Segment
Resources	TCP Dump
Processing	• Packet dismantling • Packet rebuilding
Data	Packet

Table 4-11 segment rebuilder

Segment analyzer

Identification	**Segment Analyzer**
Type	Algorithm
Purpose	• To analyze the segment and find the Round trip time of the segment, this will then be used to create the data set
Function	• FindRTTLocation() • extractRTT()
Subordinates	None
Dependencies	Segment Rebuilder
Interfaces	• Takes segment as input • Gives an RTT as outpt • For training we will also take hostip as output to use in the data set
Resources	Memory
Processing	• Identify rebuilt segment • Extract RTT • Store RTT, and optionally hostip
Data	Segment

Table 4-12 segment analyzer

Data set

Identification	**Data set**
Type	Component
Purpose	- To store the processed segments as tuples in the format: time → host - This data set will be used to train the model so that it can predict timings when used
Function	- storeInstance(rtt, hostip)
Subordinates	Segment Analyzer
Dependencies	MySQL
Interfaces	- It takes a tuple (rtt, hostip) as input - This data can later be used
Resources	Storage space
Processing	- It gets the data from the segment analyzer and takes 2 parameters as input (rtt, hostip) and saves them in an organized database
Data	Tuples

Table 4-13 Data set

Model

Identification	**Model**
Type	Component
Purpose	• To predict mapping between timing and hostip we will train an NN on a labeled dataset based on the supervised learning technique of machine learning
Function	• setParameters() • getData() • training() • calculateLoss() • applyBackprop() • getInference()
Subordinates	Database
Dependencies	Tensorflow
Interfaces	• It takes the labeled data as a tuple(rtt, hostip) • The trained model will get RTT as input and predict the hostIP
Resources	Labeled Dataset
Processing	• It gets the labeled data from the dataset and then try to optimize its weights and other parameters on the basis of the dataset provided.
Data	tuple(rtt, hostip)

Table 4-14 Model

4.5. Reuse and Relationships to other Products

Existing traffic examination strategies can be to a great extent classified into two types: side channels and covert channels. In a side channel assault, an attacker records traffic inactively and identifies the likeness between server's outbound traffic and customer's inbound traffic to connect the correspondence relationship.

The main concept of side channel attack in this project is to observe the activity of the person who is under an attack and determine which website is being acceses over the internet .there are different method through which we can determine which website is accessed, like the attack can be done by observing the size of the packet or calculating the round trip time of the packet from the client's system . Following are the remarkable name of those who did tremendous work in detecting the website being accessed by the victim.

Zhu et al used common data for the similitude estimation. Levine et al used a cross relationship system. Research demonstrated that the attacker could deduce delicate data from scrambled system traffic by looking at examples of bundle size and timing. For instance, Song et al used the packet inter arrival timing in SSHv1 associations with keystroke patterns and at last break the passwords. Sun et al. explored the sizes of HTML objects transmitted over SSL associations and could recognize the pages dependent on the number and size of items in each scrambled HTTP connection. Liberatore and Levine analyzed parcel size of HTTP traffic transmitted over the steady connections of SSH that could measurably recognize the webserver pages.

4.6. Design Decisions and Tradeoffs

This Project cannot perform the experiment in live environment since we are guessing probability of which website is accessed is being guessed so this experiment will perform in a controlled environment.

Chapter 5 Project Test and Evaluation

5.1. Introduction

The purpose of this document is to elicit all material that is essential to plan and control the test efforts for the development of this project. It specifies the test plan for application of Network Novel Delay based Side Chanel attack during the development phase and provides rationale behind necessity of these tests. This document provides an overview of the tests that were implemented, the items that were targeted by the tests, along with the testing approach that was deployed. This testing is being done according to the elicited requirements in Software Requirements Specification Document of Network Novel Delay based Side Chanel attack.

5.2. Test Items

The test items selected for testing include the following

1. Account Management
2. Tcp dump
3. RTT
4. Attack
5. Retrieving data.
6. Conversion of packets
7. calculations
8. capturing of packets
9. predict host

5.3. Features to Be Tested

The features of our web based application include the functionality mentioned in our design document. Following features are to be tested keeping in view the test items and system features aforementioned

1. Login
2. Signup
3. Manage accounts

4. Changing password
5. Dumping tcp packets
6. Extract RTT
7. Run exploit.
8. Retrieve dump or captured packets
9. Import packets
10. Load CSV
11. Calculating mean
12. Calculating standard deviation
13. Capturing live run time packets,
14. Predicting host.
15. logout

5.4. Test Approach

The system is working in modules so the testing phase will be initiated by testing each module separately i.e. unit testing, and then step by step integrating modules to test them with each other i.e. integration testing, followed by the testing of complete application as a whole.

5.5. Item Pass/Fail Criteria

Details of the test cases are specified in section Test Deliverables. Following the principles outlined below, a test item would be judged as pass or fail.

1. Items will pass the test if the actual output of each of the test case is same as the desired output of the system.
2. Any transfer of data between any modules is updated in the database.

.

5.6. Suspension Criteria and Resumption Requirements

1. The construct incorporates many critical defects which seriously limit checking out progress.
2. Software/ Hardware problem.
3. Assigned sources aren't available while needed to be examined. however unforeseen

Resumption Requirements

1. Resumption will handiest occur when the troubles that prompted the suspension have been resolved

5.7. Test Deliverables

Testing tasks

1. Development of test cases
2. Execution of tests based on the developed test cases
3. Report defects from the executed test cases, if any
4. Provision of complete test report
5. Incorporate changes later in the stage of the project development

Test cases

Following are the Test Cases:

5.7.1. Application startup testing

Test Case Name	Application Startup Testing
Test Case ID	1
Description	This feature sends the user to the login screen of the application when he/she open the application.
Testing Technique Used	Black Box Testing
Preconditions	The computer is on and is connected to the internet.
Input Values	None
Valid Inputs	None

Steps	1. Clickon the application icon.
Expected Output	The user will be sent to the login screen of the application.
Actual Output	Successful opening of the application
Status	PASS

Table 5-1 Test Case

5.7.2 Login feature testing

Test Case Name	Login Feature Testing
Test Case ID	2
Description	This feature asks the user to enter his/her credentials for login. This test case is aimed to check that feature works according to user requirement.
Testing Technique Used	Black Box Testing
Preconditions	System is running and connected to database. User has opened the login screen.
Input Values	1. Username 2. Password
Valid Inputs	1. Valid and authorized username 2. Valid and authorized password
Steps	1. Enter username. 2. Enter password. 3. Click “Login”.
Expected Output	The user credentials will be passed to the server for verification. The valid users will be directed to main page after login.
Actual Output	Successful login. User is directed to the main page of the application..
Status	PASS

Table 5-2 Test Case

5.7.3. Sign up

Test Case Name	Signup
Test Case ID	3
Description	This test case checks that the new user is entering the correct information in the data for creating an account in the application.
Testing Technique Used	Black Box Testing
Preconditions	The user is on the start up page of the application.
Input Values	First name , Last name, email,, phone number, password.
Valid Inputs	Alphanumeric values for the fields stated above
Steps	1. Click on “Sign Up”. 2. Fill the following information.
Expected Output	The user will have a personal account.
Actual Output	The user will have a personal account.
Status	PASS

Table 5-3 Test Case

5.7.4. Manage accounts

Test Case Name	Manage account
Test Case ID	4
Description	This feature allows the user to add or remove account from the application’s database.
Testing Technique Used	Black Box Testing
Preconditions	The computer is on and is connected to the internet.and user has a privilege to manage the account
Input Values	None
Valid Inputs	None

Steps	1. Click on the “manage account.
Expected Output	The user will be able to add or remove account.
Actual Output	The user will be able to add or remove account.
Status	PASS

Table 5-4 Test Case

5.7.5 Changing password

Test Case Name	Changing password
Test Case ID	5
Description	This feature allows the user to change the password of their account after answering some security questions.
Testing Technique Used	Black Box Testing
Preconditions	The computer is on and is connected to the internet.and user has a privilege to change the password.
Input Values	None
Valid Inputs	None
Steps	1. Click on the “change password”.
Expected Output	The user will be able to change password.
Actual Output	The user will be able to change password.
Status	PASS

Table 5-5 Test Case

5.7.6 TCP dump

Test Case Name	Tcp Dump
Test Case ID	6
Description	This test case checks if the traffic over the network is being captured.
Testing Technique Used	Black Box Testing
Preconditions	Attacker is logged into the account, and system is connected to

	same router of which the traffic is capturing.
Input Values	None
Valid Inputs	None
Steps	1. Click on the “Tcp Dump”. 2. Click on “start dumping”
Expected Output	Traffic will be captured from the selected router.
Actual Output	Traffic gets captured from the selected router.
Status	PASS

Table 5-6 Test Case

5.7.7. extract RTT

Test Case Name	Extract RTT
Test Case ID	7
Description	This test case checks that RTT is correctly extracted from the captured packets.
Testing Technique Used	Black Box Testing
Preconditions	Attacker is logged into the account, and system is connected to the router from which the packets is being captured.
Input Values	Click on the extract RTT button
Valid Inputs	Mouse Click
Steps	1. Click on “Extract RTT”.
Expected Output	RTT will be extracted from the captured packets.
Actual Output	RTT will be extracted from the captured packets.
Status	PASS

Table 5-7 Test Case

5.7.8 Run exploit

Test Case Name	Run Exploit
Test Case ID	8
Description	This test case checks that the system is connected to targeted router on which the attack is being done.
Testing Technique Used	Black Box
Preconditions	Staff is logged into the account, and system is connected to the router from which the packets is being captured.
Input Values	Click on the run exploit button
Valid Inputs	Mouse click on run exploit button.
Steps	1. Click on “live capturing”. 2. Click on “run Exploit”.
Expected Output	The attack will then initiated on the targeted router.
Actual Output	The attack will then initiated on the targeted router
Status	PASS

Table 5-8 Test Case

5.7.9 Retrieve dump packets

Test Case Name	Retrieve dump packets
Test Case ID	9
Description	This test case checks the packet is successfully captured from the router and dumped in to a file.
Testing Technique Used	Black Box
Preconditions	Staff is logged into the account, and system is connected to the router from which the packets is being captured.
Input Values	Mouse Click
Valid Inputs	Mouse Click
Steps	1. Click on “live capturing” 2. Click on “run Exploit”

	3. Click on “retrieve packets”
Expected Output	The dump packets captured from the targeted router is retrieved.
Actual Output	The dump packets captured from the targeted router is retrieved.
Status	PASS

Table 5-9 Test Case

5.7.10. Convert packets into CSV

Test Case Name	Convert packets into CSV
Test Case ID	10
Description	This test case checks that converting the captured packets in to csv format is working successfully or not
Testing Technique Used	Black Box
Preconditions	Staff is logged into the account, and system is connected to the router from which the packets is being captured.
Input Values	None
Valid Inputs	None
Steps	1. Click on”convert to CSV”.
Expected Output	The button “convert to CSV” will take the user on to the screen which will allow the attacker to choose given options for converting. .
Actual Output	The attacker will be given the options to convert the file into CSV formt.
Status	PASS

Table 5-10 Test Case

5.7.11. Import packets

Test Case Name	Import Packets
Test Case ID	11
Description	This test case checks that the packets that is being captured from the client system is available in the pcap file for the conversion into CSV format.
Testing Technique Used	Black Box
Preconditions	Staff is logged into the account, and system is connected to the router from which the packets is being captured.
Input Values	None
Valid Inputs	None
Steps	1. Click on “convert to CSV” 2. Click on ”Import packets”.
Expected Output	The button “Import packets” will allow the user to import packet i.e pcap file and load it in the wireshark.
Actual Output	The attacker will successfully import the pcap file of captured packets in the wireshark.
Status	PASS

Table 5-11 Test Case

5.7.12. Load CSV

Test Case Name	Load CSV

Test Case ID	12
Description	This test case checks that the captured file is successfully converted into CSV format using wireshark and load CSV allow the packets in the csv format to display on screen.
Testing Technique Used	Black Box
Preconditions	Staff is logged into the account, and system is connected to the router from which the packets is being captured, and the attacker is successfully loaded the pcap file in the CSV format.
Input Values	None
Valid Inputs	None
Steps	1. Click on "Load CSV".
Expected Output	The button "Load CSV" will successfully display the converted packets on the screen.
Actual Output	The converted packets will be display on the screen.
Status	PASS

Table 5-12 Test Case

5.7.13. Calculating mean

Test Case Name	Calculating mean
Test Case ID	13
Description	This feature sends the user to calculate the mean of each website using minimum and maximum RTT each website.
Testing Technique Used	Black Box Testing
Preconditions	The computer is on and is connected to the internet and the user has already captured the packets of the victim's traffic.

Input Values	None
Valid Inputs	None
Steps	1. Clickon the “calculate mean”
Expected Output	The user will be able to calculate mean of each website.
Actual Output	the user will be able to calculate mean of each website.
Status	PASS

Table 5-13 Test Case

5.7.14. Calculating standard deviation

Test Case Name	Calculating standard deviation
Test Case ID	14
Description	This feature sends the user to calculate the standard deviation of each website using mean RTT each website.
Testing Technique Used	Black Box Testing
Preconditions	The computer is on and is connected to the internet and the user has already captured the packets of the victim’s traffic.
Input Values	None
Valid Inputs	None
Steps	1. Clickon the “calculate standard deviation ”
Expected Output	The user will be able to calculate standard deviation of each website.
Actual Output	the user will be able to calculate standard deviation of each website.
Status	PASS

Table 5-14 Test Case

5.7.15. Run time capturing

Test Case Name	Run time capturing
Test Case ID	15
Description	This test case checks the number of clients connected on the same router and and the attacker will start capturing the packets of the targeted client.
Testing Technique Used	Black Box
Preconditions	Staff is logged into the account, and system is connected to the router from which the packets is being captured.
Input Values	None
Valid Inputs	None
Steps	Click on the “start capturing live packets”.
Expected Output	The system will start capturing the live packets of the targeted client.
Actual Output	The system will start capturing the live packets of the targeted client.
Status	PASS0

Table 5-15 Test Case

5.7.16. Predict host

Test Case Name	Predict Host
Test Case ID	16
Description	This test case will predict the host by comparing mean of the captured packet from the dump.
Testing Technique Used	Black Box
Preconditions	Attacker is logged into the account, and system is connected to the router from which the packets is being captured.

Input Values	None
Valid Inputs	None
Steps	1. Click on “predict host”.
Expected Output	The system will predict the host using of run time capture packet and packets from the dump.
Actual Output	System predict the host.
Status	PASS

Table 5-16 Test Case

5.7.17. logout

Test Case Name	Logout
Test Case ID	17
Description	This test case checks if the logout button successfully sign out the user’s account from the application.
Testing Technique Used	Black Box
Preconditions	Faculty is logged in the account .
Input Values	None
Valid Inputs	None
Steps	1. Select “Log out
Expected Output	The system will successfully logout the user from the user’s account and will take the user to the main page of the application.
Actual Output	The system will successfully logout the user from the user’s account and will take the user to the main page of the application.
Status	PASS

Table 5-17 Test Case

5.8. Responsibilities, Staffing and Training Needs

5.8.1. Responsibilities:

All integration testing tasks and component testing comes under the responsibility of all the developers.

5.8.2. Staffing and Training Needs:

The testing of the project basic knowledge of testing strategies and techniques is needed. Developers should have a knowledge of techniques such as black box testing and integration testing.

Each other's work will be tested by the developers and will be actively participating in the development and testing of the project simultaneously.

5.9. Risk and Contingencies

Efforts have been made to remove all and every chance of failure but there are certain unpredictable factors such as network issues, corrupt input data, or system failure that may lead to some issues. Error handling will be applied more deeply to cover all these issues but unforeseen circumstances may happen.

5.9.1. Schedule Risk:

The project might get behind schedule so in order to complete the project in time we will be needing to increase the hours/day that the project is being worked on.

5.9.2. Operational Risks:

To meet the goals of the project scheduling daily meetings and regular deadlines will eliminate the operational risks as well as proper communication will be provided within a group.

5.9.3. Technical risks:

Keeping the once defined requirements constant will eliminate the technical risks.

5.9.4 Programmatic Risks:

In case of a programmatic risk in order to stay inside the constraints of the project.

the scope of the project will be limited .

Chapter 6 Future Work

This projected will be extended in many ways using latest security features and technologies some of the future work are given below:

1. The network setup will be improve by allowing the network to monitor more then one client on the same access point.
2. The improved version of the project will be able to operate on the access point using different type of proxies.
3. The improved network will be able to observe the traffic of a client on different access point.

Chapter 7 Conclusion

By utilizing the modern technology features, an experiment has been conducted in a virtually controlled environment , where a network is setup comprises of an attacker client proxy server and an access point. The attacker monitored the traffic of the client and through different statistical analysis and calculation of mean and standard variance the attacker predicted the probability of which website is being accessed by the client through a squid in a can proxy. Here the side channel attack is being done by observing round trip time of each packets send between a client and server via squid in a can proxy.

Appendices

Appendix A: Glossary

1. **Activity diagrams -** are graphical representations of workflows of stepwise activities and actions with support for choice, iteration and concurrency. In the Unified Modeling Language, activity diagrams are intended to model both computational and organizational processes (i.e. workflows).
2. **Class diagram** - In the Unified Modeling Language (UML) is a type of static structure diagram that describes the structure of a system by showing the system's classes, their attributes, operations (or methods), and the relationships among objects.
3. **SDS** – In the context of software, Design Specification is usually a design document that describes all data, architectural, interface and component-level design for the software. A design specification provides explicit information about the requirements for a product and how the product is to be put together.
4. **Sequence diagram -** is an interaction diagram that shows how processes operate with one another and in what order. It is a construct of a Message Sequence Chart. A sequence diagram shows object interactions arranged in time sequence. It depicts the objects and classes involved in the scenario and the sequence of messages exchanged between the objects needed to carry out the functionality of the scenario.
5. **SRS** – A software requirements specification is a description of the software system to be developed. It lays out functional and non-functional requirements, and may include a set of use cases that describe user interactions that the software must provide.
6. **Use case diagram -** At its simplest is a representation of a user's interaction with the system and depicts the specifications of a use case. A use case diagram can portray the different types of users of a system and the case and will often be accompanied by other types of diagrams as well.
7. **RTT**: Round Trip Time.

Bibliography

Bibliography

H. Roberts, E. Zuckerman, J. York, R. Faris, and J. Palfrey, "2010 circumvention tool usage report," http://cyber.law.harvard.edu/sites/cyber.law.harvard.edu/files/2010 Circumvention Tool Usage Report.pdf, 2010.

Q. Sun, D. R. Simon, Y.-M. Wang, W. Russel, V. N. Padmanabhan, and L. Qiu, "Statistical identification of encrypted web browsing traffic," in Proceedings of the IEEE Symposium on Security and Privacy (S&P), May 2002.

G. D. Bissias, M. Liberatore, D. Jensen, and B. N. Levine, "Privacy vulnerabilities in encrypted http streams," in Proceedings of the 5th Privacy Enhancing Technologies Workshop (PET), May 2005.

M. Liberatore and B. N. Levine, "Inferring the source of encrypted http connections," in Proceedings of the ACM conference on Computer and Communication Security (CCS), October 2006.

C. V. Wright, S. E. Coull, and F. Monrose, "Traffic morphing: An efficient defense against statistical traffic analysis," in Proceedings of the Network and Distributed Security Symposium (NDSS), February 2009.

X. Luo, P. Zhou, E. W. W. Chan, W. Lee, R. K. C. Chang, and R. Perdisci, "Httpos: Sealing information leaks with browser-side obfuscation of encrypted flows," in Proceedings of the 18th Annual Network and Distributed System Security Symposium (NDSS), February 2011.

Y. Zhu, X. Fu, B. Graham, R. Bettati, and W. Zhao, "On flow correlation attacks and countermeasures in mix networks," in Proceedings of Workshop on Privacy Enhancing Technologies (PET), May 2004.

B. N. Levine, M. K. Reiter, C. Wang, and M. Wright, "Timing attacks in low-latency mix-based systems," in Proceedings of Financial Cryptography (FC), February 2004.

ORIGINALITY REPORT

17%	5%	4%	13%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	Zhen Ling, Junzhou Luo, Yang Zhang, Ming Yang, Xinwen Fu, Wei Yu. "A novel network delay based side-channel attack: Modeling and defense", 2012 Proceedings IEEE INFOCOM, 2012 Publication	4%
2	Submitted to Colorado Technical University Online Student Paper	2%
3	Submitted to CSU, San Jose State University Student Paper	1%
4	Submitted to RDI Distance Learning Student Paper	1%
5	Submitted to South University Student Paper	1%
6	www.slideshare.net Internet Source	1%

8	Submitted to CSU, Fullerton Student Paper	1%
9	archive.org Internet Source	<1%
10	www.cs.utexas.edu Internet Source	<1%
11	www.geeksforgeeks.org Internet Source	<1%
12	Submitted to Higher Education Commission Pakistan Student Paper	<1%
13	Submitted to University of Zululand Student Paper	<1%
14	Submitted to University of Bradford Student Paper	<1%
15	manualzz.com Internet Source	<1%
16	Submitted to Mahidol University Student Paper	<1%
17	Submitted to Universiti Tunku Abdul Rahman Student Paper	<1%

18	Submitted to University of Technology, Sydney Student Paper	<1%
19	fr.slideshare.net Internet Source	<1%
20	Submitted to London School of Commerce Student Paper	<1%
21	magizbox.com Internet Source	<1%
22	Submitted to Institute of Technology, Nirma University Student Paper	<1%
23	Submitted to Florida Virtual School Student Paper	<1%
24	Submitted to University of Northumbria at Newcastle Student Paper	<1%
25	Submitted to University of Lugano Student Paper	<1%
26	Submitted to University of East London Student Paper	<1%
27	Submitted to City College Coventry Student Paper	<1%

28	en.wikipedia.org Internet Source	<1%
29	Submitted to Swinburne University of Technology	<1%
30	Submitted to Liverpool John Moores University Student Paper	<1%
31	stackoverflow.com Internet Source	<1%
32	Submitted to Cranfield University Student Paper	<1%
33	hep-proj-grid-fabric.web.cern.ch Internet Source	<1%
34	Submitted to University of KwaZulu-Natal Student Paper	<1%
35	Submitted to Misr International University Student Paper	<1%
36	ibimapublishing.com Internet Source	<1%
37	Submitted to Anglia Ruskin University Student Paper	<1%

38	Submitted to Visvesvaraya Technological University Student Paper	<1%
39	Submitted to Kuwait University Student Paper	<1%
40	Submitted to Thapar University, Patiala Student Paper	<1%
41	www.ijtra.com Internet Source	<1%
42	Zhen Ling, Junzhou Luo, Kui Wu, Xinwen Fu. "Protocol-level hidden server discovery", 2013 Proceedings IEEE INFOCOM, 2013 Publication	<1%
43	Submitted to University of Bedfordshire Student Paper	<1%
44	www.w3.org Internet Source	<1%
45	Submitted to University of London External System Student Paper	<1%
46	phobos.ramapo.edu Internet Source	<1%

47	Submitted to UT, Dallas Student Paper	<1%
48	Submitted to University of Sunderland Student Paper	<1%
49	Submitted to Auckland University of Technology Student Paper	<1%
50	Submitted to Universiti Sains Islam Malaysia Student Paper	<1%
51	Submitted to Multimedia University Student Paper	<1%
52	Submitted to Majan College Student Paper	<1%
53	Submitted to Texas A & M University, Kingville Student Paper	<1%
54	Submitted to Universiti Malaysia Sarawak Student Paper	<1%