Remotely Administrative Trojan for Mobile Phones (RAT)

By

Capt Sohaib Rasheed Bhatti

Capt Jamal ud din Wahab

Supervised by:

Col Asim Dilawar Bakhshi

Maj Wajahat Sultan

Submitted to the faculty of Department of Computer Software Engineering,
Military College of Signals, National University of Sciences and Technology, Islamabad,
in partial fulfillment for the requirements of B.E Degree in Software Engineering.
25 June 2022

In the name of ALLAH, the Most benevolent, the Most Courteous

CERTIFICATE OF CORRECTNESS AND APPROVAL

This is to officially state that the thesis work contained in this report

"Remotely Administrative Trojan for Mobile Phones"

is carried out by

Capt Sohaib Rasheed Bhatti

Capt Jamal ud din Wahab

under my supervision and that in my judgement, it is fully ample, in scope and excellence, for the degree of Bachelor of Software Engineering in Military College of Signals, National University of Sciences and Technology (NUST), Islamabad.

Approved by
Supervisor

Col Asim Dilawar Bakhshi
Maj Wajahat Sultan

Date: _____

DECLARATION OF ORIGINALITY

We hereby declare that no portion of work presented in this thesis has been submitted in support of another award or qualification in either this institute or anywhere else.

ACKNOWLEDGEMENTS

Allah Subhan'Wa'Tala is the sole guidance in all domains.
Our Parents, colleagues and most of all supervisors Col Asim Dilawar Bakhshi, Maj Wajahat Sultan Without your guidance it would never have been possible for us to complete our Final Year Project.

Plagiarism Certificate (Turnitin Report)

This thesis has 8% similarity index. Turnitin report endorsed by Supervisor is attached.

_____

Capt Sohaib Rasheed Bhatti

_____

Capt Jamal Ud Din

_____

_____

Signature of Supervisor

ORIGINALITY REPORT

8%	2%	2%	7%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	Submitted to Higher Education Commission Pakistan Student Paper	6%
2	Submitted to Nexford University Student Paper	1%
3	boris.unibe.ch Internet Source	1%
4	Shanshan Jiang, Kine Jakobsen, Jonas Bueie, Jingyue Li, Peter Halland Haro. "A Tertiary Review on Blockchain and Sustainability with Focus on Sustainable Development Goals", Institute of Electrical and Electronics Engineers (IEEE), 2022 Publication	<1%

ABSTRACT

Andro Spy is a very lightweight Android RAT (remote administration tool) to break into an Android-powered smartphone remotely. It gives you the power to establish control over android devices with an easy-to-use GUI and all the features you need to monitor them. It's the interface is really sleek and easy to use and even comes with some extra FUN features that not all the RATs offers.

The aim of this RAT is to follow the practice of stealthy, ongoing hacking seeking to accumulate data over time, as opposed to causing damage to information or systems, is known as an advanced persistent threat (APT**).** Remote Access Trojans are a powerful tool in this type of attack because they do not slow down a computer's performance or automatically begin deleting files once installed—and because they're so adaptable

.

Table of Contents

Chapter 1: Introduction 1

1.1 Overview 1

1.2 Problem Statement 2

1.3 Proposed Solution 3

1.4 Working Principle 4

1.4.1 Input Component 4

1.4.2 Application Component 5

1.4.3 Tools and Methodologies 6

1.4.3 Minimum Hardware Requirements 7

1.4.4 Software Requirements for Development Environment 8

1.4.5 Languages 9

1.4.6 Output Component 9

1.5 Objectives 10

1.5.1 General Objectives: 10

1.5.2 Academic Objectives: 10

1.6 Scope 10

1.7 Deliverables 11

1.8 Relevant Sustainable Development Goals 12

1.9 Structure of Thesis 12

Chapter 2: Literature Review 13

2.1 Industrial background 13

2.2 Existing solutions and their drawbacks 13

Chapter 3: RAT Techniques 15

3.1 RAT Techniques 16

3.1.1 Working OF RAT 16

3.1.2 System Architecture 17

3.1.3 Architectural Design 17

3.2 Decomposition Description 18

3.3 Design Rationale

Chapter 4 System Architecture 19

4.1 Architectural Design 19

4.2 Decomposition Description 24

4.3 Design Rationale 25

Chapter 5: DATA DESIGN 26

5.1 Data Description 26

Chapter 6 Human Interface Design 28

6.1 Overview of User Interface 29

6.2 Interface Images

Chapter 7: Conclusion 30

Chapter 8: Future Work 32

Chapter 1: Introduction

The introduction provides an overview of the entire FYP with purpose, scope, definitions, acronyms, abbreviations, references, and overview of the SDD. The aim of this document is to present a detailed description of the project Remotely Administrated Trojan which with the help of the output device. The detailed designed structure of the RAT is provided in this document.

The practice of stealthy, ongoing hacking seeking to accumulate data over time, as opposed to causing damage to information or systems, is known as an advanced persistent threat (APT). Remote Access Trojans are a powerful tool in this type of attack because they do not slow down a computer's performance or automatically begin deleting files once installed—and because they're so adaptable.

Unlike other types of viruses—such as keyloggers, which record everything someone types on the infected computer, or ransomware, which essentially holds a computer or files hostage until the hacker is paid off—Remote Access Trojans give hackers total administrative control over the infected system, so long as they remain undetected.

1.1 Overview

Remote administration means any method of controlling a computer or computers from a remote location. Any system with an Internet connection, TCP/IP or on a Local Area Network (LAN) can be remotely administered. Remote administration can be used for a number of activities and can span multiple servers. RATs are stealthily planted and help gain access of victim machines, through patches, updates, games, E-mail attachments,

or even in legitimate-looking binaries. RAT can be made into FUD that is fully undetectable so that the antivirus in the victim's computer cannot detect it. RATs can give the attacker access to the directories, webcams, keyboard etc. The attacker can find out what victim is typing on his/her computer using key-loggers. The attacker can also control the keyboard leaving victim unable to type on their own computer and can install many other malicious programs. These RATs can be injected via pen drives, hard drives or any other external storage devices. These devices are called "bash bunny". A Bash Bunny can install the RAT, backdoor and payload just by inserting the drive in victim's computers. The RAT creates processes to hide its activities and injects running tasks with malicious code which go unnoticed by the system. They can cause Distributive Denial of Service (DDoS) attacks, obtain sensitive information, and record the actions of the current session of the system such as screen preview, keystrokes. They redirect traffic to other systems for obtaining specific services. To prepare software that can be used by intelligence agencies and LEAs for monitoring and surveillance of terrorists, anti-state agents, and pre-selected targets

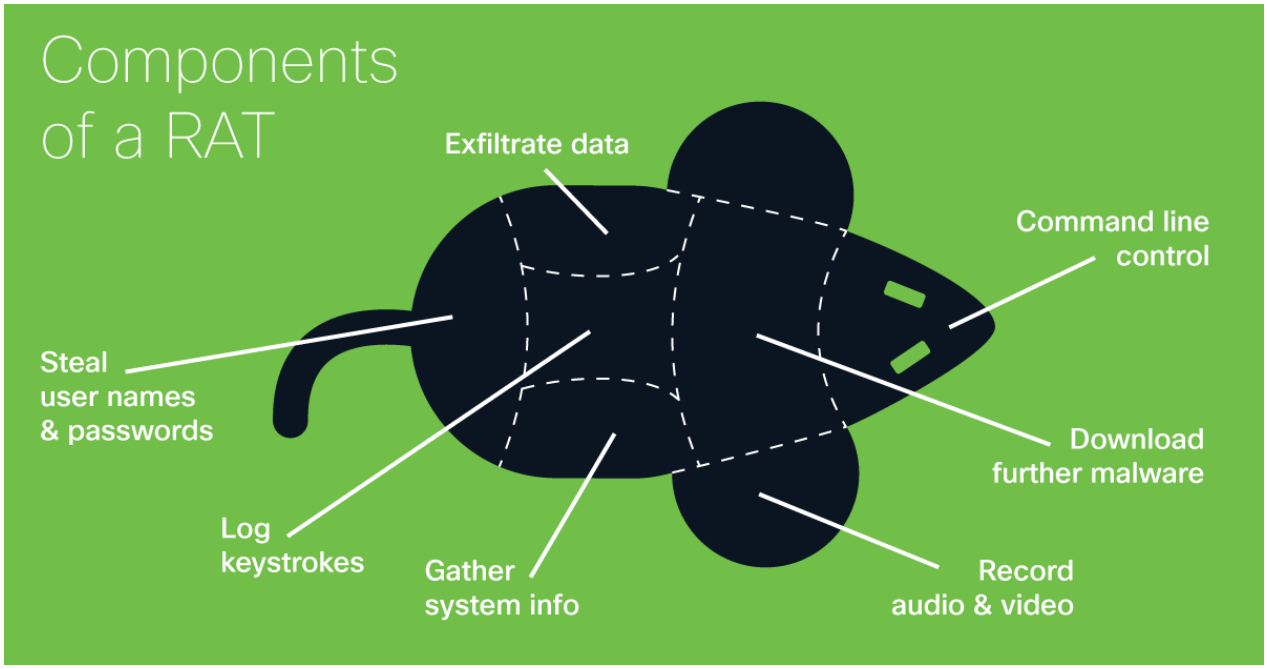
1.2 Problem Statement

In this modern era of communication cell phones have immensely changed the way people communicate today. A cell phone can be all a person need for interaction. From a cell phone, a person can make calls, send text messages, emails, send and also receive directions, buy things online, do online banking, listen to music and much more. Since one can do everything with one device, there is no longer a need to go around with multiple devices.

Due to easily access of mobile phones now a days, it has emerged as a serious threat against LEAs in monitoring and surveillance of terrorists, anti-state agents, and pre-selected targets.

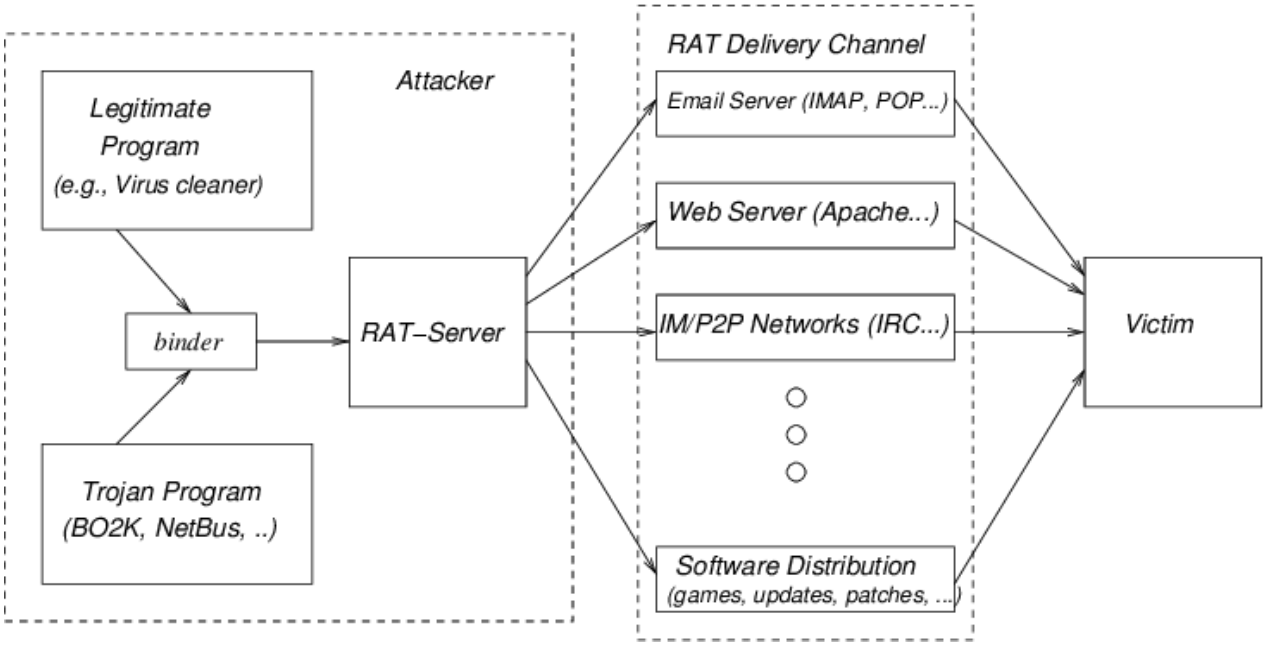
1.3 Proposed Solution

So far some work has been done on this Project. Creating brand-new Remote Access Trojans capable of avoiding detection is a time-intensive process, which means it's usually more worthwhile for hackers to use them against larger targets like governments, corporations, and financial institutions including surveillance and monitoring of the targets.



1.4 Working Principle

The RAT consists of two parts. The first is a server-side application based on C#, in our case, just our desktop or laptop, but this could be scaled up to some degree if needed. This acts as a control panel which we use to create and connect to the RAT. The second part is client-side, which is the infected Android application we'll use as a backdoor.



1.4.1 Input Component

The first is a server-side application based on C#, in our case, just our desktop or laptop, but this could be scaled up to some degree if needed. This acts as a control panel which we use to create and connect to the RAT.

1.4.2 Application Component

The second part is client-side, which is the infected Android application we'll use as a backdoor.

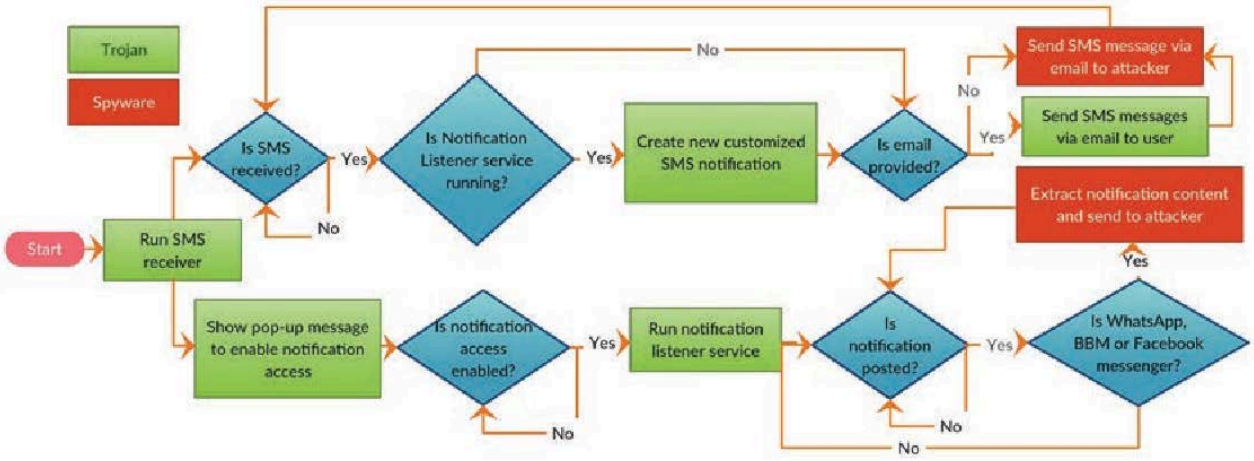
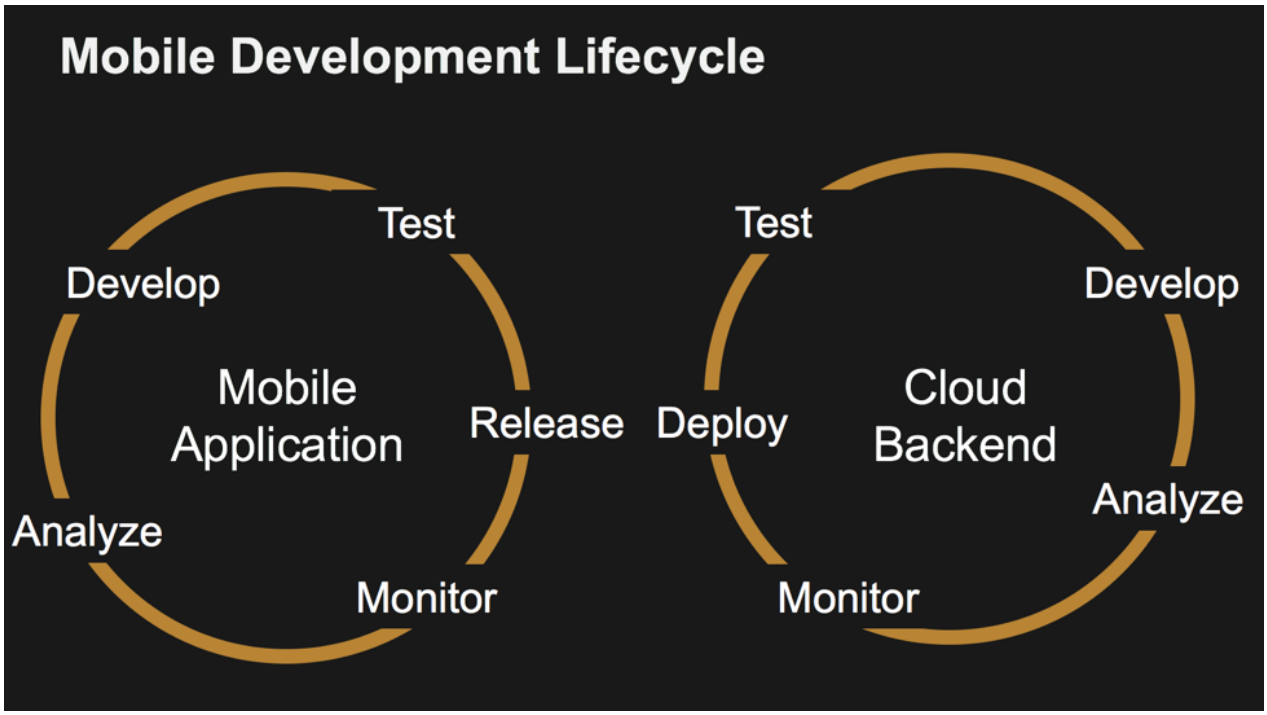


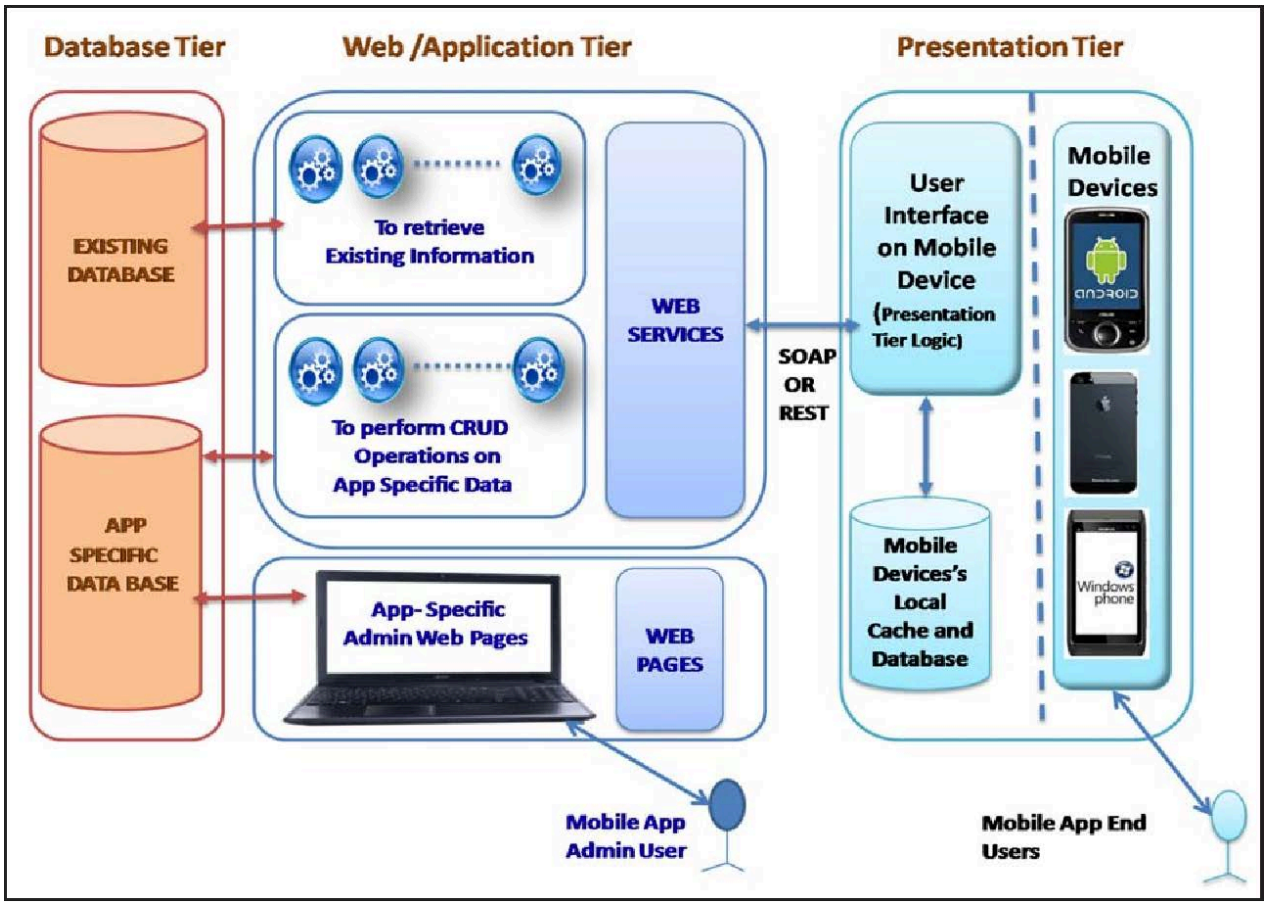
Fig. 1: Application flowchart



1.4.3 Tools and Methodologies

Followings are the requirements to start the development process of the proposed system.

The Android application runs as a service (not an activity) that begins during startup. Therefore the user does not have to interact with the service (although there is an activity debug to configure the IP address and port connection).



1.4..3 Hardware Requirements for setting up Development Environment

1. Dual Core Processor
2. 8 GB RAM
3. 40 GB Hard Disk

1.4.4 Software Requirements for Development Environment

1. Visual basic

2. Encrypter

3. Binder

4. Antivirus evasion tool

5. Different testing platforms

6. SSH Tunnels

5.1.5 Special Skills Required:

1. Encoding

2. Encryption

3. Social engineering

4. Binding

5. Bypassing/ Evasion of Antiviruses and different security tools

6. Crypting a code

7. Making software FUD

1.4.5 Languages

5 C#

6 Java

7 Xamarin

1.4.6 Output Component

This Apk generated will be the output file which will be binded by any Legitimate APK and will be Crypted to evade the Antiviruses and firewalls

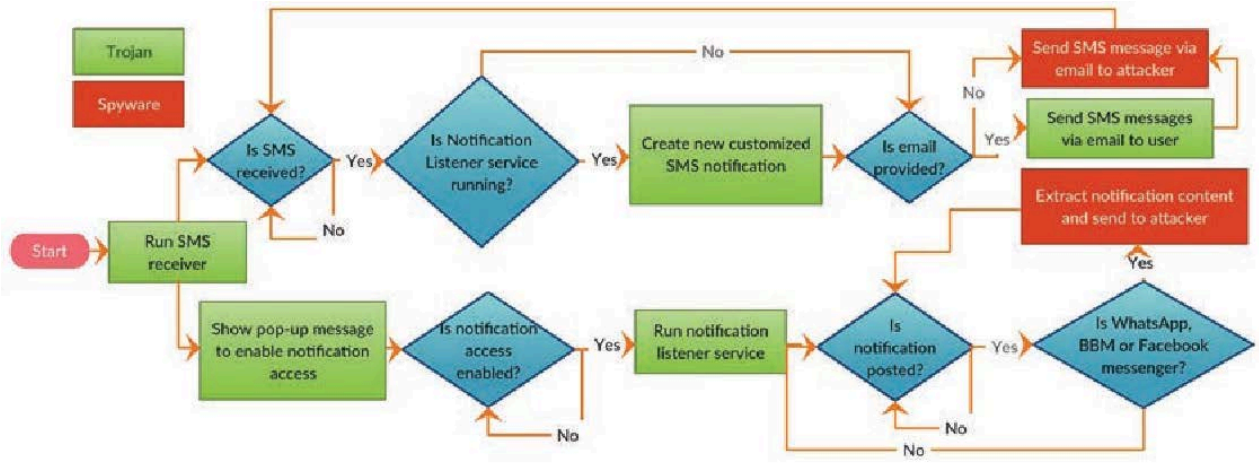


Fig. 1: Application flowchart

1.5 Objectives

1.5.1 General Objectives:

The aim of this RAT is to follow the practice of stealthy, ongoing hacking seeking to accumulate data over time, as opposed to causing damage to information or systems, is known as an advanced persistent threat (APT). Remote Access Trojans are a powerful tool in this type of attack because they do not slow down a computer's performance or automatically begin deleting files once installed—and because they're so adaptable

1.5.2 Academic Objectives:

- Development of a smart and intelligent RAT system
- To implement Machine Learning techniques and simulate the results
- To increase productivity by working in a team
- To design a project that contributes to the welfare of society

1.6 Scope

This project finds its scope is to follow the practice of stealthy, ongoing hacking seeking to accumulate data over time, as opposed to causing damage to information or systems, is known as an advanced persistent threat (APT**).** Remote Access Trojans are a powerful tool in this type of attack because they do not slow down a computer's performance or automatically begin deleting files once installed—and because they're so adaptable

1.7 Deliverables

1. Monitoring and surveillance of pre-selected targets
2. Monitoring and surveillance of terrorist and anti-state agents
3. Security purposes
4. Intelligence purposes
5. Surveillance purposes

1.8 Relevant Sustainable Development Goals

The practice of stealthy, ongoing hacking seeking to accumulate data over time, as opposed to causing damage to information or systems, is known as an advanced persistent threat (APT). Remote Access Trojans are a powerful tool in this type of attack because they do not slow down a computer's performance or automatically begin deleting files once installed—and because they're so adaptable.

Andro Spy is a very lightweight Android RAT (remote administration tool) to break into an Android-powered smartphone remotely. It gives you the power to establish control over android devices with an easy-to-use GUI and all the features you need to monitor them. It's the interface is really sleek and easy to use and even comes with some extra FUN features that not all the RATs offers.

1.9 Structure of Thesis

Chapter 2 contains the literature review and the background and analysis study this thesis is based upon.

Chapter 3 contains the RAT Techniques

Chapter 4 introduces detailed data design.

Chapter 5 introduces human interface design

Chapter 6 contains the conclusion of the project.

Chapter 7 highlights the future work needed to be done for the commercialization of this project.

Chapter 2: Literature Review

The main goal of this projects it to prepare software that can be used by intelligence agencies and LEAs for monitoring and surveillance of terrorists, anti-state agents, and pre-selected targets.

2.1 Industrial background

In this modern era of communication cell phones have immensely changed the way people communicate today. A cell phone can be all a person need for interaction. From a cell phone, a person can make calls, send text messages, emails, send and also receive directions, buy things online, do online banking, listen to music and much more. Since one can do everything with one device, there is no longer a need to go around with multiple devices.

Due to easily access of mobile phones now a days, it has emerged as a serious threat against LEAs in monitoring and surveillance of terrorists, anti-state agents, and pre-selected targets

2.2 Existing solutions and their drawbacks

Pakistan don't have any indigenous RAT to monitor the Activities of suspects. Due to easily access of mobile phones now a days, it has emerged as a serious threat against LEAs in monitoring and surveillance of terrorists, anti-state agents, and pre-selected targets

- A computer virus attaches itself to a program or file so it can spread from one computer to another, leaving infections as it travels. Much like human viruses, computer viruses can range in severity: Some viruses cause only mildly annoying effects while others can damage your hardware, software or files.

Almost all viruses are attached to an executable file, which means the virus may exist on your computer but it cannot infect your computer unless you run or open the malicious program. It is important to note that a virus cannot be spread without a human action, (such as running an infected program) to keep it going. People continue the spread of a computer virus, mostly unknowingly, by sharing infecting files or sending e-mails with viruses as attachments in the e-mail.

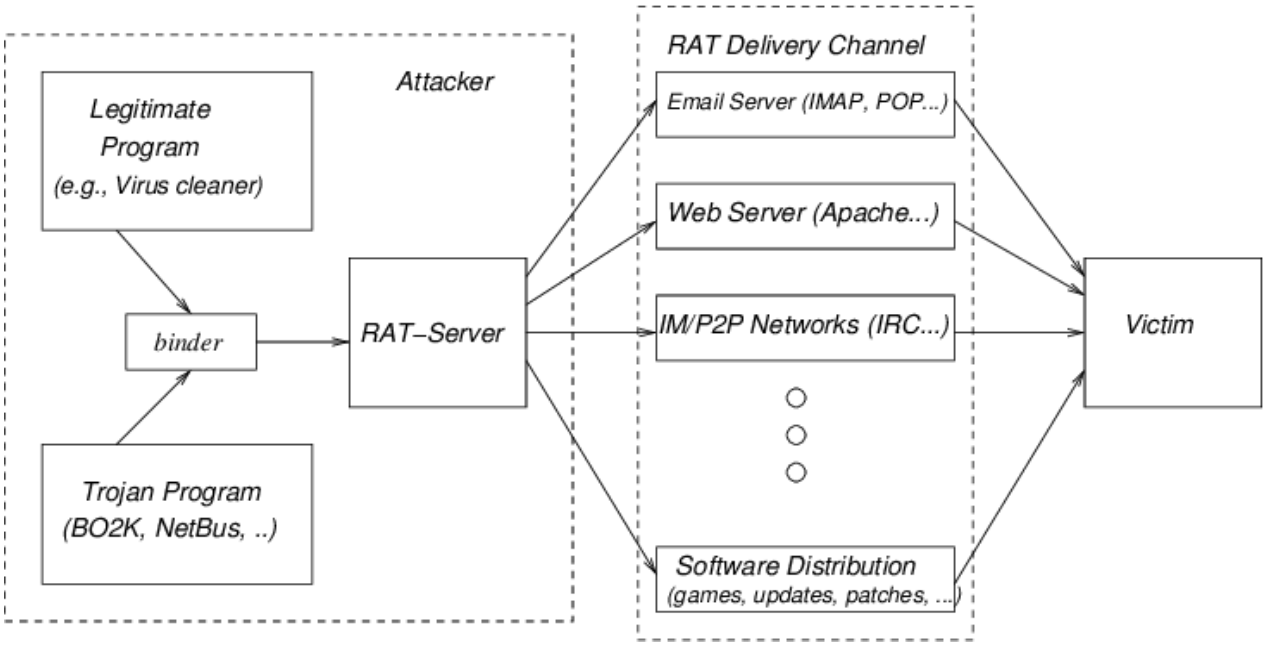
- A Trojan horse is not a virus. It is a destructive program that looks as a genuine application. Unlike viruses, Trojan horses do not replicate themselves but they can be just as destructive. Trojans also open a backdoor entry to your computer which gives malicious users/programs access to your system, allowing confidential and personal information to be theft.
- A key logger is a small piece of software that, when downloaded into your computer, will record every keystroke. The key logger will capture every keystroke on the keyboard, every username, password and credit card number, etc., exposing all of your data and personal information.
- A remote access Trojan (RAT) is a malware program that includes a back door for administrative control over the target computer. RATs are usually downloaded invisibly with a user-requested program -- such as a game -- or sent as an email attachment.

Chapter 3: RAT Techniques

3.1 RAT Techniques

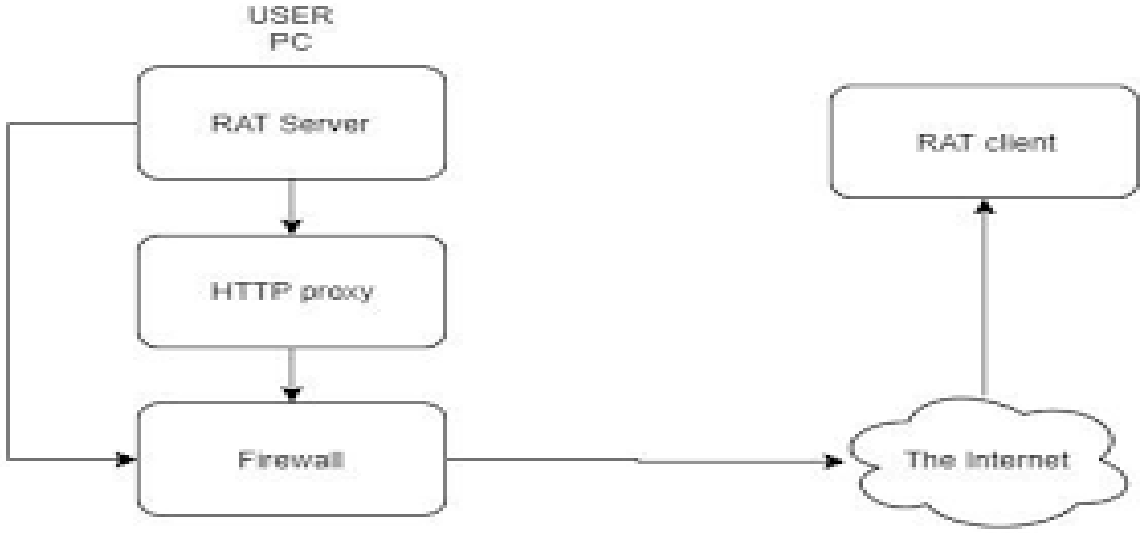
Remote Access Trojans can be installed in a number of methods or techniques, and will be similar to other malware infection vectors. Specially crafted email attachments, web-links, download packages, or .torrent files could be used as a mechanism for installation of the software. Targeted attacks by a motivated attacker may deceive desired targets into installing such software via social engineering tactics, or even via temporary physical access of the desired computer.

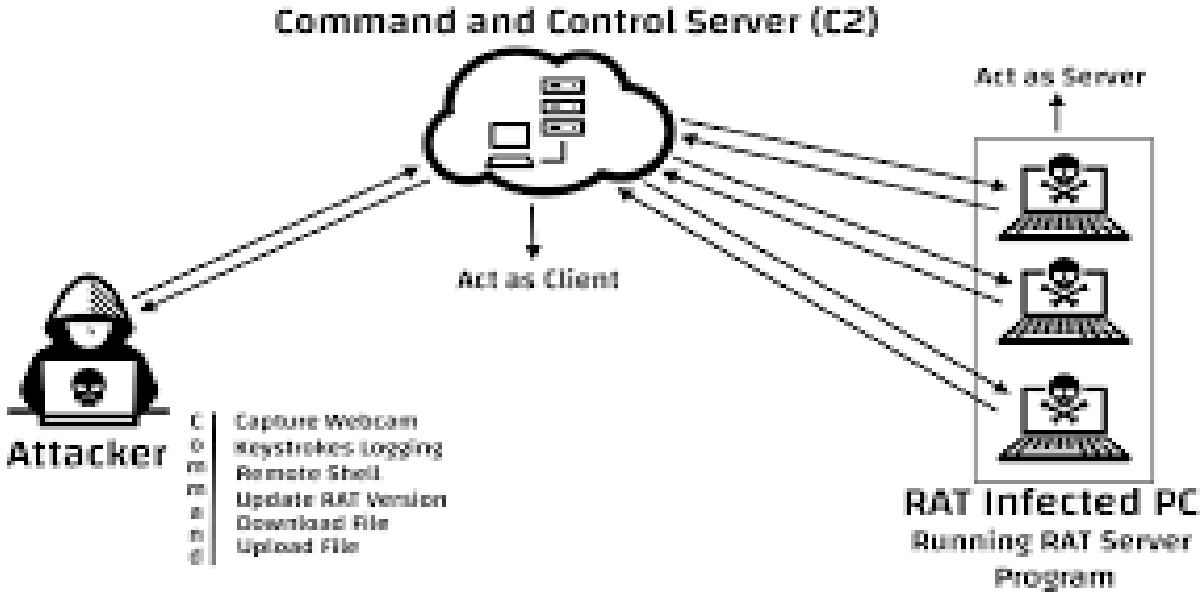
There are a large number of Remote Access Trojans. Some are more well-known than others. SubSeven, Back Orifice, ProRat, Turkojan, and Poison-Ivy are established programs. Others, such as CyberGate, DarkComet, Optix, Shark, and VorteX Rat have a smaller distribution and utilization. This is just a small number of known Remote Access Trojans, and a full list would be quite extensive, and would be continually growing.

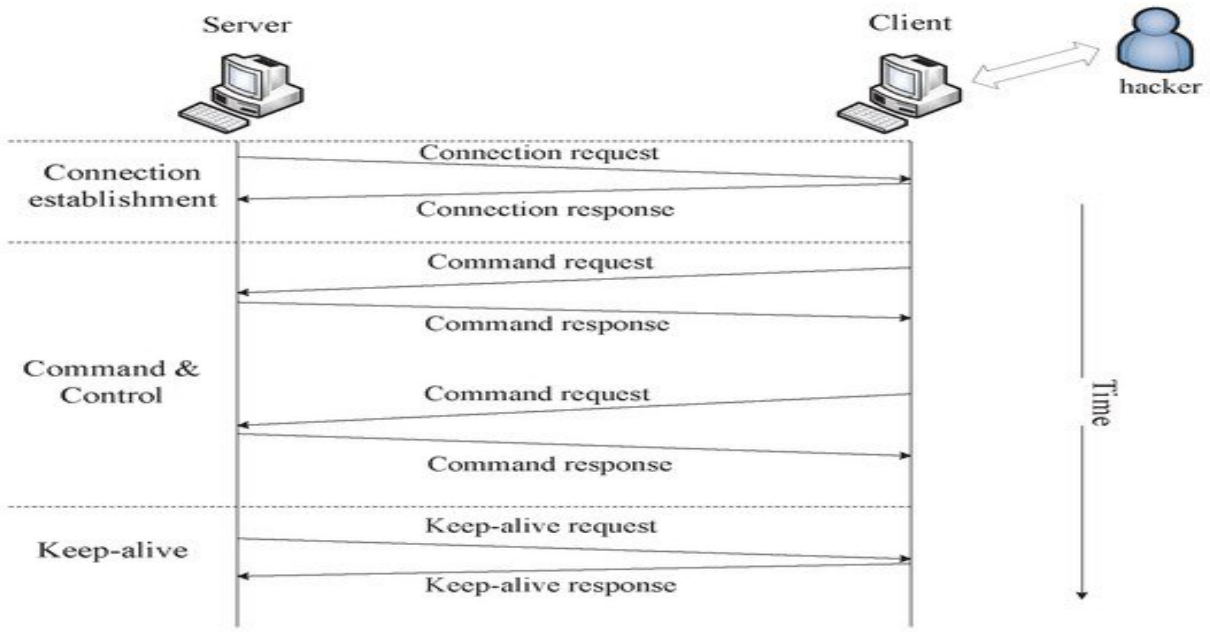


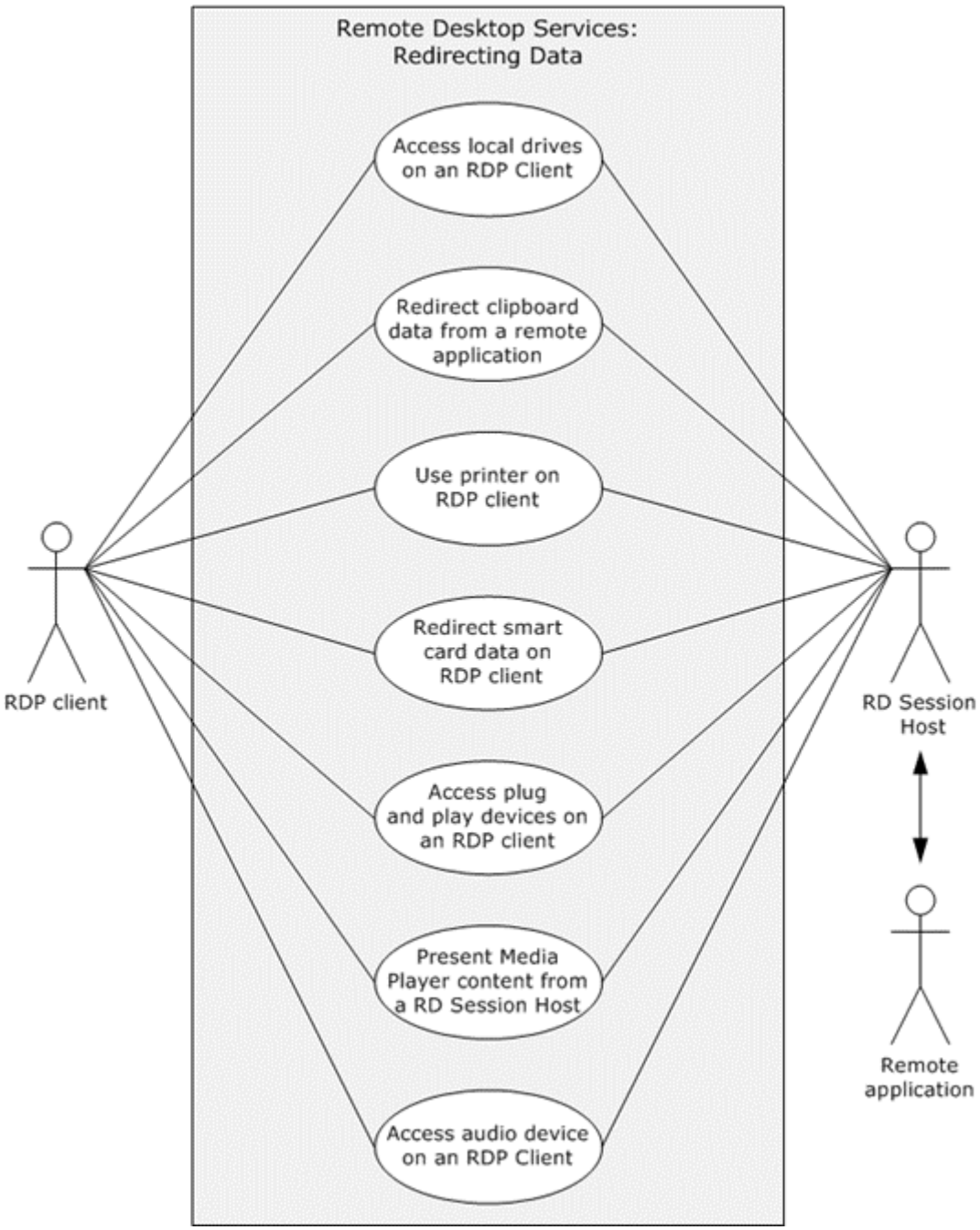
3.1.1 Working OF RAT

The RAT consists of two parts. The first is a server-side application based on C#, in our case, just our desktop or laptop, but this could be scaled up to some degree if needed. This acts as a control panel which we use to create and connect to the RAT. The second part is client-side, which is the infected Android application we'll use as a backdoor.





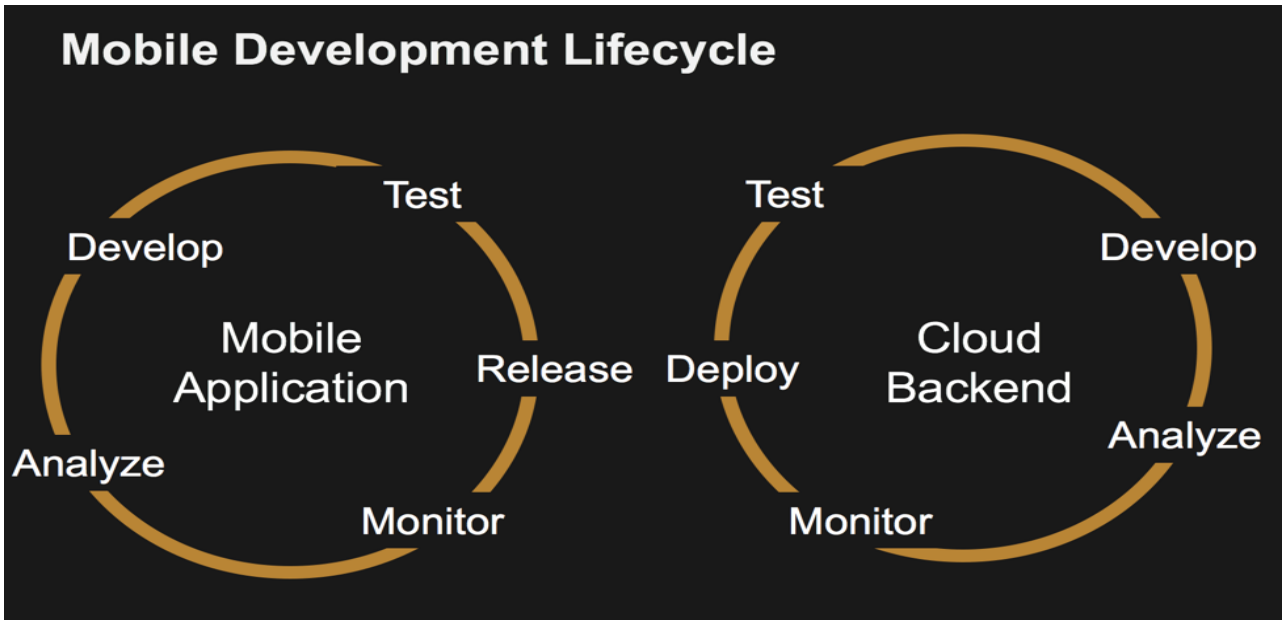

Server
Client
hacker
Connection establishment
Connection request
Connection response
Command & Control
Command request
Command response
Command request
Command response
Keep-alive
Keep-alive request
Keep-alive response
Time


Remote Desktop Services:
Redirecting Data
Access local drives on an RDP Client
Redirect clipboard data from a remote application
Use printer on RDP client
Redirect smart card data on RDP client
Access plug and play devices on an RDP client
Present Media Player content from a RD Session Host
Access audio device on an RDP Client
RDP client
RD Session Host
Remote application

4 SYSTEM ARCHITECTURE

4.1 Architectural Design

A Trojan generally has two parts Client and Server or Master and Slave. We can say Server is Slave and Client is Master. So a server side is installed on a remote host and the attacker manipulates it with client software.
Remotely administrated Trojan for mobile phones architecture is a set of structural elements along with their interfaces that compose the system. It includes some techniques which help one in developing an Android mobile Trojan. The app architecture gets formulated by taking all procedure that works on mobiles. This set of system helps to avoid customer problems.

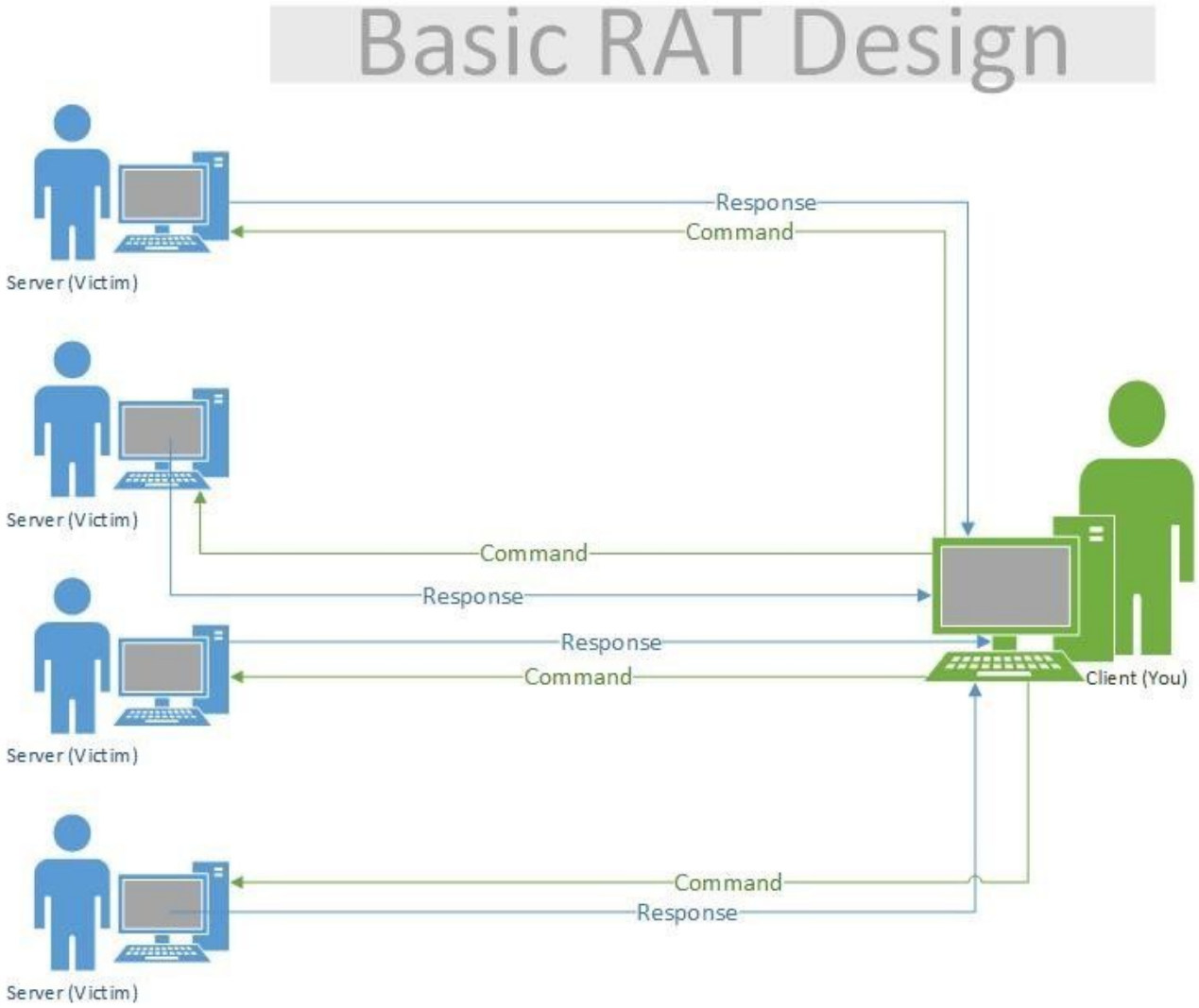


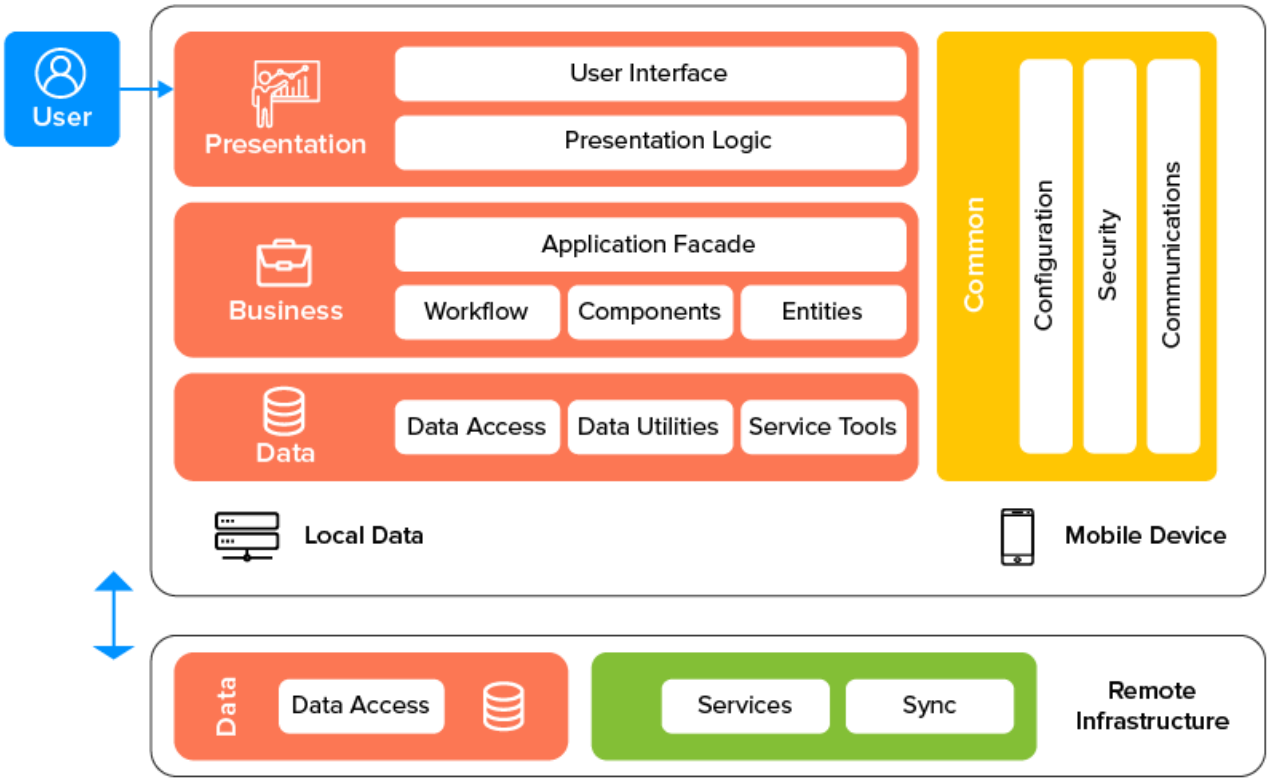
The RAT consists of two parts. The first is a server-side application based on C#, in our case, just our desktop or laptop, but this could be scaled up to some degree if needed. This acts as a control panel which we use to create and connect to the RAT. The second part is client-side, which is the infected Android application we'll use as a backdoor.

Code for Desktop Interface:

The RAT consists of two parts. The first is a server-side application based on C#, in

our case, just our desktop or laptop, but this could be scaled up to some degree if needed. This acts as a control panel which we use to create and connect to the RAT. The second part is client-side, which is the infected Android application we'll use as a backdoor.





The most popular multilayer architecture is the three-layer architecture. This three-layer architecture is important for designing or creating mobile app architecture. It refers to your component's internal architecture. The given are the main three important layers of mobile architecture design:

Presentation Layer

The presentation layer consists of two components. These two components include the User Interface and UI process. While discussing this layer, the primary focus is the end user's mobile application's presentation. The mobile application developers should look for a client type. So, it would be compliant with the infrastructure.

During the presentation layer stage, you need to decide on many important things. These include themes, fonts, colours and shadings. The developer should keep in mind the client's deployment restriction while mobile app architecture designs. Another important necessity of this layer is to select the correct data format. Then use robust data validation mechanisms.

Business Layer

The business layer is for the elements on the business front. This layer looks at how the app will present the business to the end-users. This layer includes business components, workflow, and entities. These layers' complexity is more complex than others. It concerns too many problems such as caching and logging. The exception management and security challenges also add to its complexity.

There are two parts of this layer to reduce the complexity. These include the service layer and domain model. The service layers are for common application function sets. While the domain model is for knowledge and expertise linked to specific problems.

Data Access Layer

The data access layers are to meet the application needs. They offer efficient and secure data transactions. For this purpose, a developer needs to design this layer. It combines different parts including data utilities, data access components, and service agents.

The selection of the correct data format is important. Also, having a strong validation technique is another factor that makes it important to design this layer. Mobile app developers should consider the maintenance of the data. This practice helps in keeping this layer changeable with the business requirements.

Factors to Consider During Mobile App Development Architecture

It is important to build a better application architecture as it can get you success. You need to keep the details of mistakes during mobile app development architecture. This practice can lead you to success. If you avoid the problems and don't consider the factors, your app can be a failure. The following includes the things you need to consider while developing the architecture:

Determining the Device Type

Determining the device type is another important factor. There are different categories of smartphones that you need to determine and consider. The operating system decides the type of smartphones. There are various android phones, iPhones and many others.

They are based on the operating system they use. Besides the category or type of device, you should consider many other things. Other things include screen size, resolution, and CPU characteristics. Furthermore, you should also consider storage capacity and availability of development tool framework.

Bandwidth Scenarios

Your target audience's bandwidth scenarios are important to consider. There can be times when there will be zero connectivity. While creating a mobile app the targets audience internet network is an important factor. If your app is slow on the users' internet, they would abandon the app. This state would not be good for your business.

You need to consider the account power consumption, design access mechanism, and speed. Furthermore, choose the best software protocols and hardware for your mobile app. There are many things you need to consider. You need to adjust all these things for a slow and intermittent internet connection.

User Interface

For a mobile application, a great user interface plays an important part. A great user interface makes it easy and comfortable for users to interact with the app. The application interface should be simple and creative. It should not confuse or mislead users. The more, the simpler and creative it would be, the evolving the connection between your app and users. You should use a creative interface according to the demand of your target audience.

The Right Navigation Method

Choosing the right navigation method is important and crucial. It is an important aspect of mobile app development. The navigation method should be according to the app requirements and customers' preferences. Choosing the best fit after analyzing different navigations methods can lead to success. Some of the popular navigation methods include:

Stacked Navigation Bar

In this navigation bar, there's the design of a fixed bar. You put their links to all other elements within your mobile app.

Tab Controller

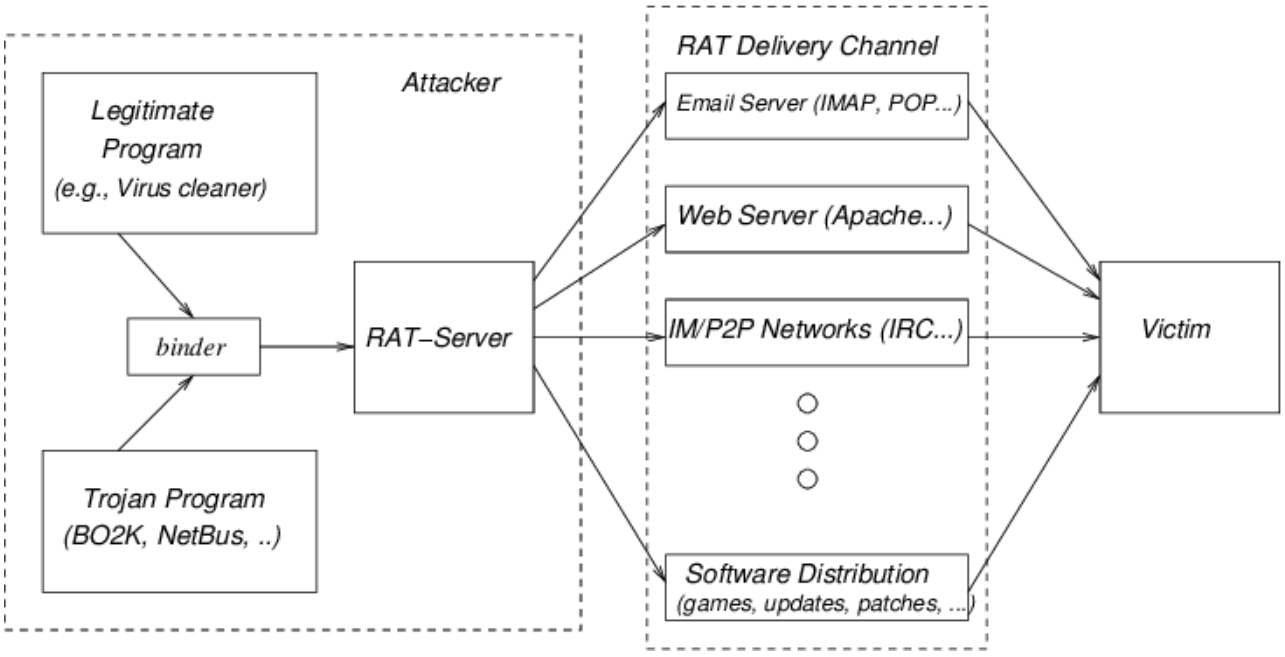
With the tab controller, one can switch between the groups of tabs with links

Real-Time Updates vs. Push Notifications

You should also decide whether you want to create a real-time update or push button. You can decide this factor by keeping your audience in mind. The real-time update can be expensive but is a compelling feature.

4.2 Decomposition Description

Trojan Binding and Crypting



4.3 Design Rationale

So we chose this design because of the following reasons:

1. **Performs specific task**
   Crypted systems performs some specific function or tasks and provides Security.
2. **Low Cost**
   The price of this indigenously created RAT is not so much as compared to open Market tools.
3. **Time Specific**
   It performs the tasks within a certain time frame.
4. **Low Data Bandwidth Consumption**
   RAT does not requires a more data speed and does not consumes
5. **High Efficiency**
   The efficiency level of generated APKs are so high.
6. **Minimal User interface**
   These systems require less user interface and easy to use.
7. **Less Human intervention**
   These systems require no human intervention or very less human intervention.

5 DATA DESIGN

5.1 Data Description

Andro Spy is a very lightweight Android RAT (remote administration tool) to break into an Android-powered smartphone remotely. It gives you the power to establish control over android devices with an easy-to-use GUI and all the features you need to monitor them. It's the interface is really sleek and easy to use and even comes with some extra FUN features that not all the RATs offer. Build a fresh custom backdoor APK or bind the payload with any existing APK such as a game or social media app.

1. Easy to use GUI interface.
2. Simple APK generator.
3. Powerful Files Explorer with all access priviliges.
4. Read and Write Messages remotely.
5. Make a phone call or record an active call.
6. Browse Call Logs.
7. Read/Write Contact List.
8. Remote Camera to capture Images & Videos from target device.
9. Listen to the live conversations through remote Mic, and record the audio from Mic.
10. Check Internet Browser History.
11. GPS Locator.
12. List of all the installed Applications.
13. Get phone's detailed info.

14. FULLY STEALTH MODE..!
15. Multi port support: Can work on any port.
16. Insertion point encoding.
17. Run multiple patches on a single device.
18. Transmit data securely from and to the device over the network.
19. Capable of controlling program configurations.
20. Notifications hidden from the phone's notification bar.
21. Name of the package can be changed to anything.

Before the RATs are installed they are customized that is the default TCP/UDP ports the listener/host IP, changing them to such as apk's or games or any software or to make it more believable they are attached with a genuine apk or game or software. The most efficient method of creating a RAT is to code it yourself via terminal and convert it into an executable. The most basic way of injecting a RAT is through E-mail, apk, games, software, or anything which is executable. For DDos the RATs are spread on many computers for this the easiest way for an attacker is to go on chat platforms and select from the active user at random and inject the RAT in their system. Once the RAT is injected in the device it can outlive reboots system, crashes evade Anti viruses. It edits registry and files and can be triggered during every reboot transparently.

6 HUMAN INTERFACE DESIGN

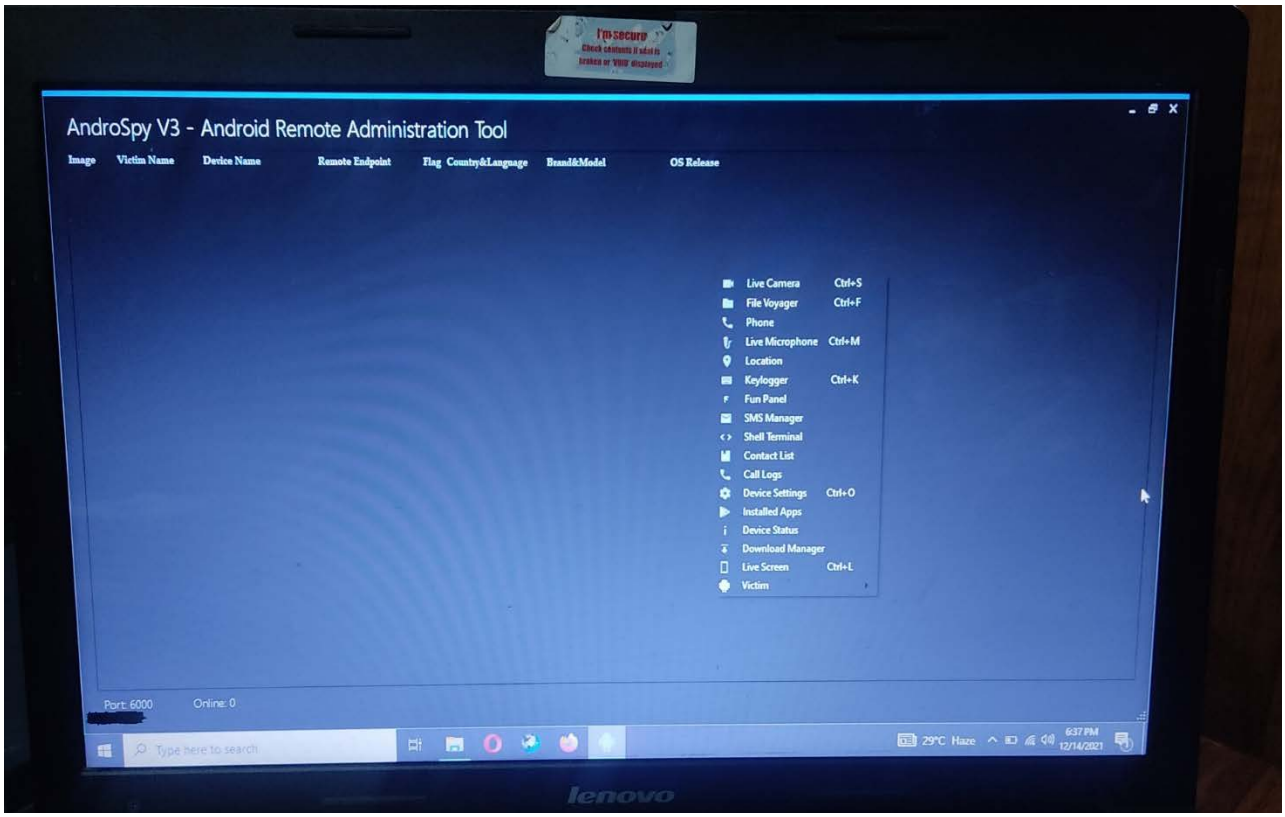
6.1 Overview of User Interface

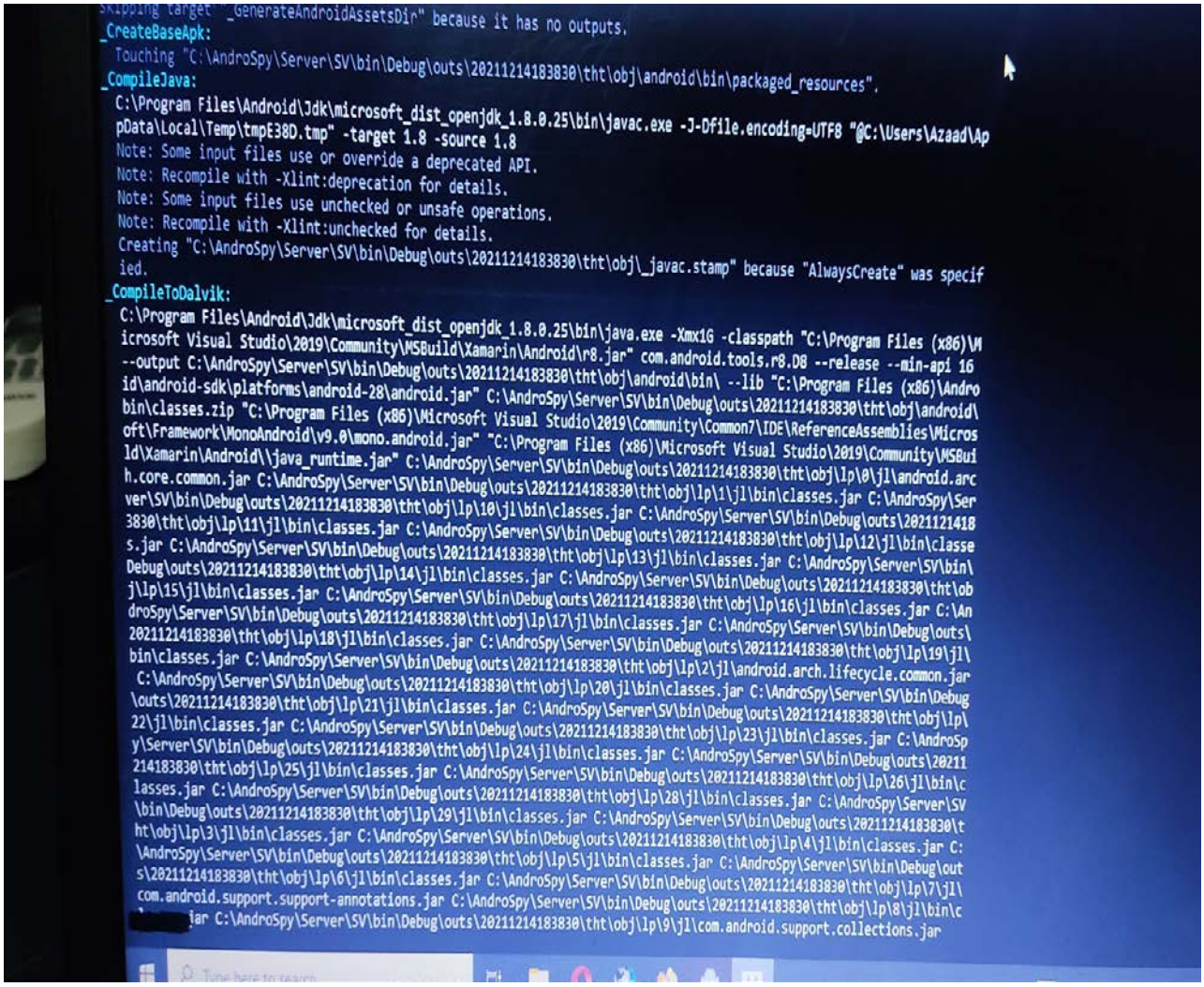
In this project, the huma user interface is Software based. Meaning that the point where the user will interact with our device is totally based on Software using a Desktop Interface. So, to easily understand the user interface design, following points will help a lot.

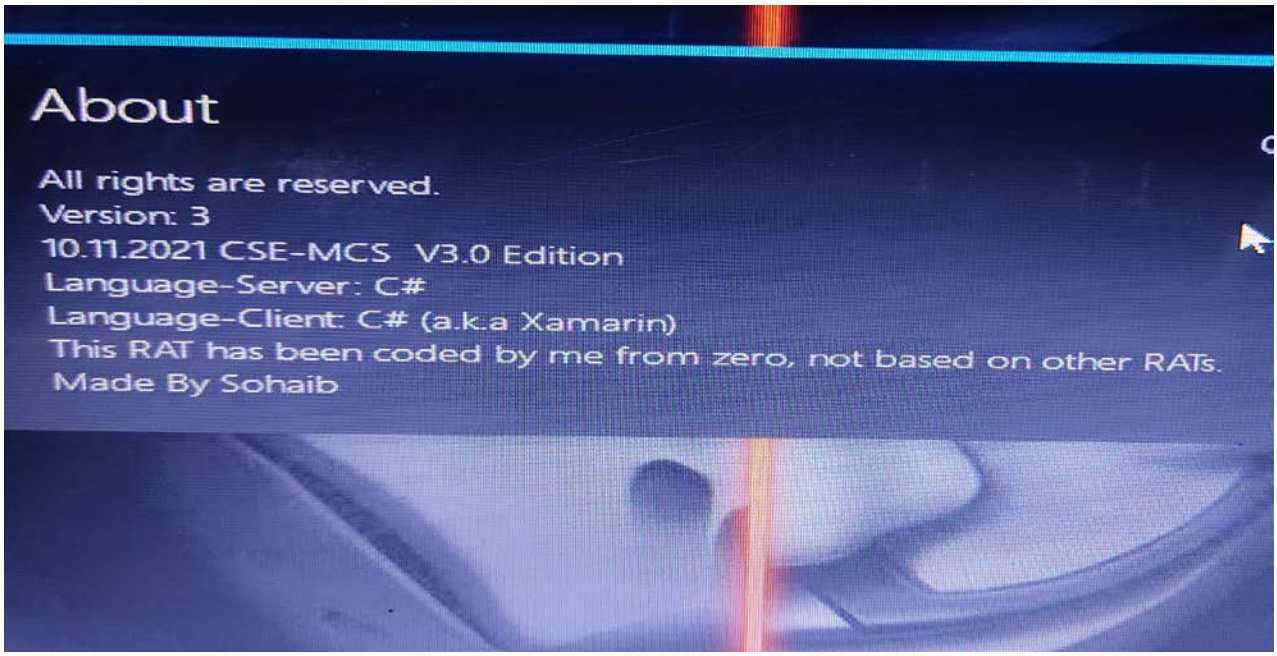
1. The Desktop Interface will be started on desktop.
2. UDP/TCP port will be given before executing
3. The APK will be generated and obsfucated.

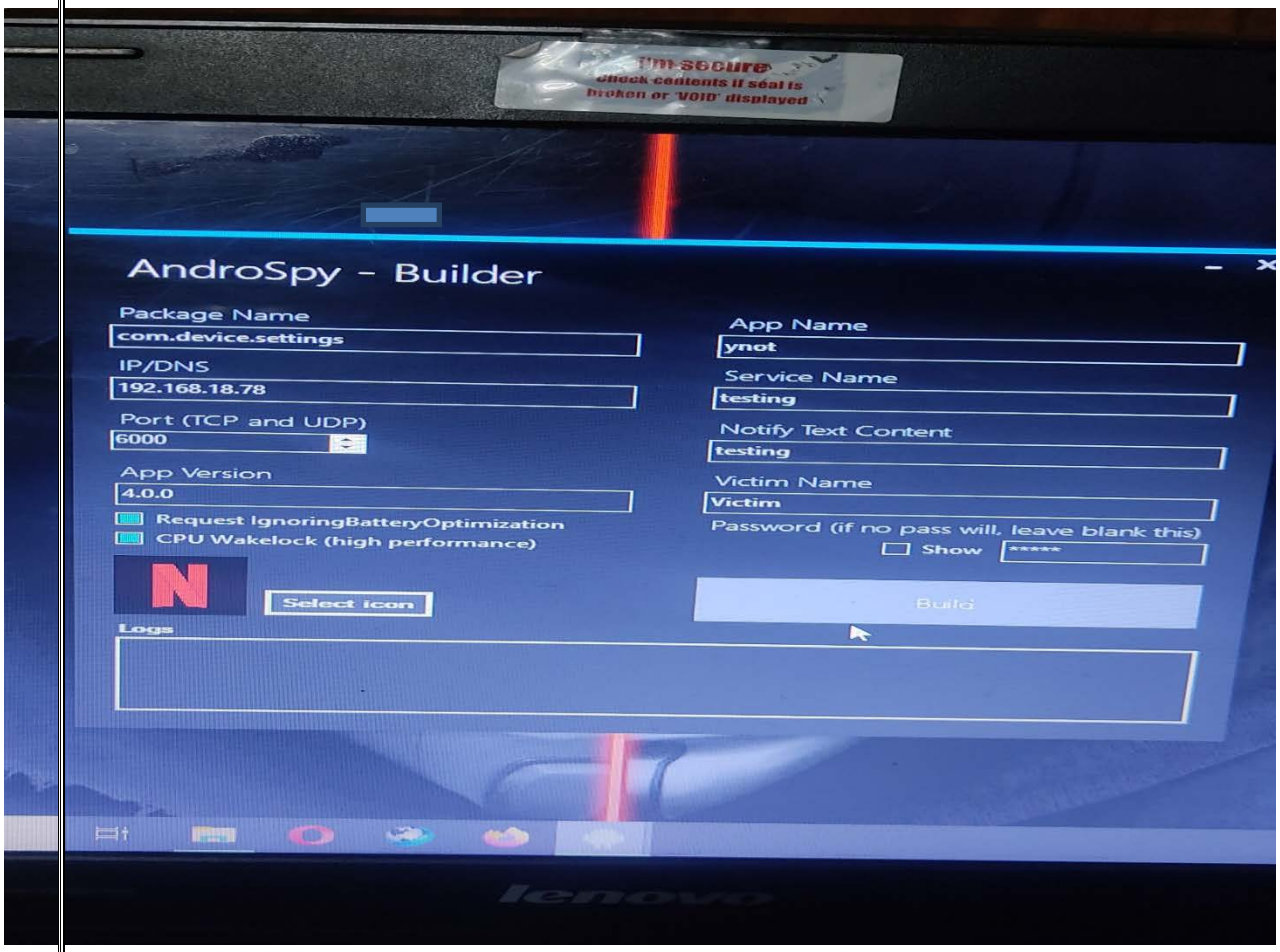
So, in this way, after Creating an APK our Trojan is ready to exploit any mobile device.

6.2 Interface Images




Skipping target "_GenerateAndroidAssetsDir" because it has no outputs.
_CreateBaseApk:
Touching "C:\AndroSpy\Server\SV\bin\Debug\outs\20211214183830\tht\obj\android\bin\packaged_resources".
_CompileJava:
C:\Program Files\Android\Jdk\microsoft_dist_openjdk_1.8.0.25\bin\javac.exe -J-Dfile.encoding=UTF8 "@C:\Users\Azaad\AppData\Local\Temp\tmpE38D.tmp" -target 1.8 -source 1.8
Note: Some input files use or override a deprecated API.
Note: Recompile with -Xlint:deprecation for details.
Note: Some input files use unchecked or unsafe operations.
Note: Recompile with -Xlint:unchecked for details.
Creating "C:\AndroSpy\Server\SV\bin\Debug\outs\20211214183830\tht\obj_javac.stamp" because "AlwaysCreate" was specified.
_CompileToDalvik:
C:\Program Files\Android\Jdk\microsoft_dist_openjdk_1.8.0.25\bin\java.exe -Xmx1G -classpath "C:\Program Files (x86)\Microsoft Visual Studio\2019\Community\MSBuild\Xamarin\Android\r8.jar" com.android.tools.r8.D8 --release --min-api 16 --output C:\AndroSpy\Server\SV\bin\Debug\outs\20211214183830\tht\obj\android\bin\ --lib "C:\Program Files (x86)\Android\android-sdk\platforms\android-28\android.jar" C:\AndroSpy\Server\SV\bin\Debug\outs\20211214183830\tht\obj\android\bin\classes.zip "C:\Program Files (x86)\Microsoft Visual Studio\2019\Community\Common7\IDE\ReferenceAssemblies\Microsoft\Framework\MonoAndroid\v9.0\mono.android.jar" "C:\Program Files (x86)\Microsoft Visual Studio\2019\Community\MSBuild\Xamarin\Android\\java_runtime.jar" C:\AndroSpy\Server\SV\bin\Debug\outs\20211214183830\tht\obj\lp\0\jl\android.arch.core.common.jar C:\AndroSpy\Server\SV\bin\Debug\outs\20211214183830\tht\obj\lp\1\jl\bin\classes.jar C:\AndroSpy\Server\SV\bin\Debug\outs\20211214183830\tht\obj\lp\10\jl\bin\classes.jar C:\AndroSpy\Server\SV\bin\Debug\outs\20211214183830\tht\obj\lp\11\jl\bin\classes.jar C:\AndroSpy\Server\SV\bin\Debug\outs\20211214183830\tht\obj\lp\12\jl\bin\classes.jar C:\AndroSpy\Server\SV\bin\Debug\outs\20211214183830\tht\obj\lp\13\jl\bin\classes.jar C:\AndroSpy\Server\SV\bin\Debug\outs\20211214183830\tht\obj\lp\14\jl\bin\classes.jar C:\AndroSpy\Server\SV\bin\Debug\outs\20211214183830\tht\obj\lp\15\jl\bin\classes.jar C:\AndroSpy\Server\SV\bin\Debug\outs\20211214183830\tht\obj\lp\16\jl\bin\classes.jar C:\AndroSpy\Server\SV\bin\Debug\outs\20211214183830\tht\obj\lp\17\jl\bin\classes.jar C:\AndroSpy\Server\SV\bin\Debug\outs\20211214183830\tht\obj\lp\18\jl\bin\classes.jar C:\AndroSpy\Server\SV\bin\Debug\outs\20211214183830\tht\obj\lp\19\jl\bin\classes.jar C:\AndroSpy\Server\SV\bin\Debug\outs\20211214183830\tht\obj\lp\2\jl\android.arch.lifecycle.common.jar C:\AndroSpy\Server\SV\bin\Debug\outs\20211214183830\tht\obj\lp\20\jl\bin\classes.jar C:\AndroSpy\Server\SV\bin\Debug\outs\20211214183830\tht\obj\lp\21\jl\bin\classes.jar C:\AndroSpy\Server\SV\bin\Debug\outs\20211214183830\tht\obj\lp\22\jl\bin\classes.jar C:\AndroSpy\Server\SV\bin\Debug\outs\20211214183830\tht\obj\lp\23\jl\bin\classes.jar C:\AndroSpy\Server\SV\bin\Debug\outs\20211214183830\tht\obj\lp\24\jl\bin\classes.jar C:\AndroSpy\Server\SV\bin\Debug\outs\20211214183830\tht\obj\lp\25\jl\bin\classes.jar C:\AndroSpy\Server\SV\bin\Debug\outs\20211214183830\tht\obj\lp\26\jl\bin\classes.jar C:\AndroSpy\Server\SV\bin\Debug\outs\20211214183830\tht\obj\lp\28\jl\bin\classes.jar C:\AndroSpy\Server\SV\bin\Debug\outs\20211214183830\tht\obj\lp\29\jl\bin\classes.jar C:\AndroSpy\Server\SV\bin\Debug\outs\20211214183830\tht\obj\lp\3\jl\bin\classes.jar C:\AndroSpy\Server\SV\bin\Debug\outs\20211214183830\tht\obj\lp\4\jl\bin\classes.jar C:\AndroSpy\Server\SV\bin\Debug\outs\20211214183830\tht\obj\lp\5\jl\bin\classes.jar C:\AndroSpy\Server\SV\bin\Debug\outs\20211214183830\tht\obj\lp\6\jl\bin\classes.jar C:\AndroSpy\Server\SV\bin\Debug\outs\20211214183830\tht\obj\lp\7\jl\com.android.support.support-annotations.jar C:\AndroSpy\Server\SV\bin\Debug\outs\20211214183830\tht\obj\lp\8\jl\bin\c
[redacted]iar C:\AndroSpy\Server\SV\bin\Debug\outs\20211214183830\tht\obj\lp\9\jl\com.android.support.collections.jar


About
All rights are reserved.
Version: 3
10.11.2021 CSE-MCS V3.0 Edition
Language-Server: C#
Language-Client: C# (a.k.a Xamarin)
This RAT has been coded by me from zero, not based on other RATs.
Made By Sohaib



Chapter 7: Conclusion

In this thesis, we have described a RAT. Andro Spy is a very lightweight Android RAT (remote administration tool) to break into an Android-powered

smartphone remotely. It gives you the power to establish control over android devices with an easy-to-use GUI and all the features you need to monitor them. It's the interface is really sleek and easy to use and even comes with some extra FUN features that not all the RATs offer. Build a fresh custom backdoor APK or bind the payload with any existing APK such as a game or social media app.

1. Easy to use GUI interface.
2. Simple APK generator.
3. Powerful Files Explorer with all access priviliges.
4. Read and Write Messages remotely.
5. Make a phone call or record an active call.
6. Browse Call Logs.
7. Read/Write Contact List.
8. Remote Camera to capture Images & Videos from target device.
9. Listen to the live conversations through remote Mic, and record the audio from Mic.
10. Check Internet Browser History.
11. GPS Locator.
12. List of all the installed Applications.
13. Get phone's detailed info.
14. FULLY STEALTH MODE..!
15. Multi port support: Can work on any port.
16. Insertion point encoding.
17. Run multiple patches on a single device.
18. Transmit data securely from and to the device over the network.
19. Capable of controlling program configurations.
20. Notifications hidden from the phone's notification bar.
21. Name of the package can be changed to anything.

Before the RATs are installed they are customized that is the default TCP/UDP ports the listener/host IP, changing them to such as apk's or games or any software or to make it more believable they are attached with a genuine apk or game or software. The most efficient method of creating a RAT is to code it yourself via terminal and convert it into an executable. The

most basic way of injecting a RAT is through E-mail, apk, games, software, or anything which is executable. For DDos the RATs are spread on many computers for this the easiest way for an attacker is to go on chat platforms and select from the active user at random and inject the RAT in their system. Once the RAT is injected in the device it can outlive reboots system, crashes evade Anti viruses. It edits registry and files and can be triggered during every reboot transparently.

Chapter 8: Future Work

Future milestones that need to be achieved to commercialize this project are the following**.**

To prepare software that can be used by intelligence agencies and LEAs for monitoring and surveillance of terrorists, anti-state agents, and pre-selected targets

The characteristics of the RAT are :

- Manipulate processes in task manager.
- Hinders mouse movement randomly.
- Files are deleted, moved, downloaded without permission.
- Infect system with viruses, malwares and worms
- Keyboard stops working.
- Anytime access to victim's computer is provided.
- Software that can be used by intelligence agencies and LEAs for monitoring and surveillance
- Monitoring and surveillance of pre-selected targets
- Monitoring and surveillance of terrorist and anti-state agents
- Security purposes
- Intelligence purposes
- Surveillance purposes

REFERENCES

1. S. Albawi, T. A. Mohammed, and S. Al-Zawi, "Understanding of a convolutional neural network," *Proc. 2017 Int. Conf. Eng. Technol. ICET 2017*, vol. 2018-Janua, no. April 2018, pp. 1–6, 2018.

2. P. Choudhary, R. Sharma, G. Singh, S. Das, and S. G. Dhengre, “A Survey Paper On Drowsiness Detection & Alarm System for Drivers,” *Int. Res. J. Eng. Technol.*, pp. 1433–1437, 2016.

3. Y. Ed-Doughmi, N. Idrissi, and Y. Hbali, “Real-time system for driver fatigue detection based on a recurrent neuronal network,” *J. Imaging*, vol. 6, no. 3, 2020.

4. A. Jalilifard and E. B. Pizzolato, “An efficient K-NN approach for automatic drowsiness detection using single-channel EEG recording,” *Proc. Annu. Int. Conf. IEEE Eng. Med. Biol. Soc. EMBS*, vol. 2016-Octob, pp. 820–824, 2016.

5. G. Chen and W. Wang, “Target recognition in infrared circumferential scanning system via deep convolutional neural networks,” *Sensors (Switzerland)*, vol. 20, no. 7, pp. 1–18, 2020.

6. J. Gwak, A. Hirao, and M. Shino, “An investigation of early detection of driver drowsiness using ensemble machine learning based on hybrid sensing,” *Appl. Sci.*, vol. 10, no. 8, 2020.

7. M. P. Singh, G. Srivastava, and P. Kumar, “Internet traffic classification using machine learning,” *Int. J. Database Theory Appl.*, vol. 9, no. 12, pp. 45–54, 2016.