Deployment and End User Manual

Introduction

This is the end-user's guide on configuring a Python environment and executing code. This tutorial is meant to assist you in quickly and effectively learning Python programming. This guide will show you how to execute the Python code on your computer, regardless of your level of programming experience and a basic idea about our chat application and how it works.

Python Environment Setup

Anaconda installation

A variety of helpful packages and tools for scientific computing, data analysis, and machine learning are included in the Python distribution Anaconda. The steps to install Anaconda are as follows:

- Download the Anaconda distribution for your operating system by visiting the Anaconda website at https://www.anaconda.com.
- **Launch Installer:** Launch the Anaconda installer after the download is finished. To install Anaconda on your computer, adhere to the on-screen directions. During installation, be careful to select the option to add Anaconda to your system PATH.
- **Verify Installation:** To ensure that Anaconda has been installed correctly, after installation, open a terminal or command line and execute conda --version. The version number of the conda should appear on the screen.

Configuring an Environment for Conda

An isolated environment with a particular set of packages and their dependencies is called a conda environment. You can use it to handle various package sets for various projects. To create a conda environment suitable for CovertComm, follow these steps:

- **Creating a Conda Environment**: To create a new conda environment called py311 in our case, open a terminal or command prompt and run the following command:

```
conda create --name py311 python=3.11
```

- **Activate the Environment:** You must activate the environment after it has been established. To activate the py311 environment, type the following command:

```
conda activate py311
```

- **Installation of packages:** Once the environment is up and running, conda can be used to install other packages. To run our project, the following packages are to be installed to ensure successful execution of the code:

```
pip install scapy, PyQt5, prompt_toolkit, cryptography
```

- **Running the code in the environment:** Once the packages are successfully installed and everything is setup, we can then run the python code in any OS. There should be two or more machines which can be connected i.e. from the local system to the VM or vice versa or one system to the other via the below mentioned command, depending on the frontend you need to activate:

```
python chat_frontend_[cli|tkinter|pyqt].py <network_adapter>
<target_ip> <listen_port> <password>
```

chat_frontend_[cli|tkinter|pyqt].py: python file. Choose cli for the CLI frontend, tkinter for the Tkinter frontend or pyqt for the PyQt5 one.
<network_adapter>: Network interface name, e.g. **eth0** or **ens33**

<target_ip>: IP address of the receiving system, e.g. **192.168.56.11**
<listen_port>: Connection port to connect to and receive from, e.g. **3334**
<password>: Password, which can be anything of your own interest, as long as it is between 8 and 20 characters and is alphanumeric.

How does CovertComm application work?

In the section above, we saw how to setup the environment and how to execute the code. Now we will see a basic guide on how to use our chat application and which is which.

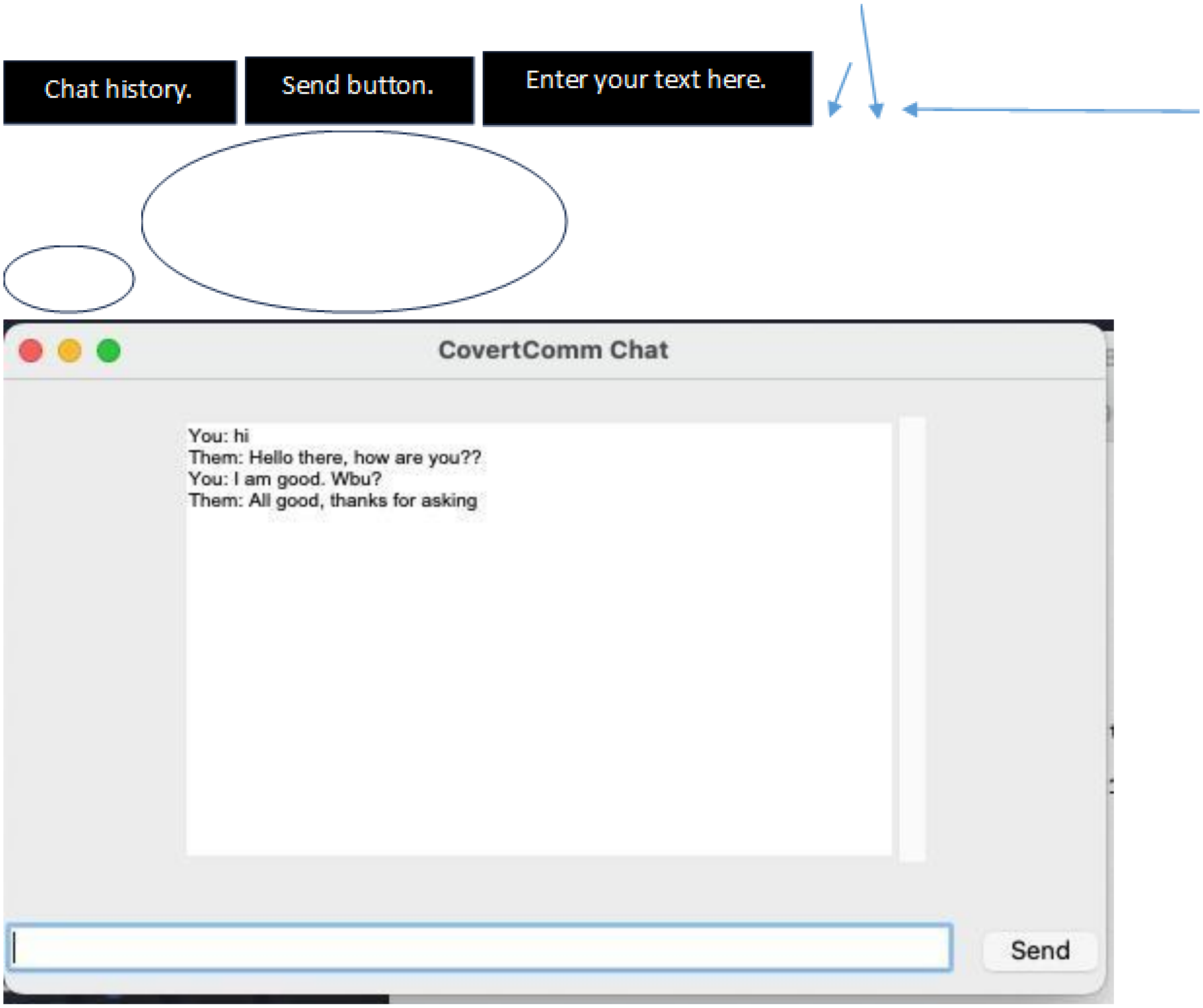
Interface Overview

Chat Window: A chat window will appear when you execute the code, allowing you to send messages.

Sending Messages:

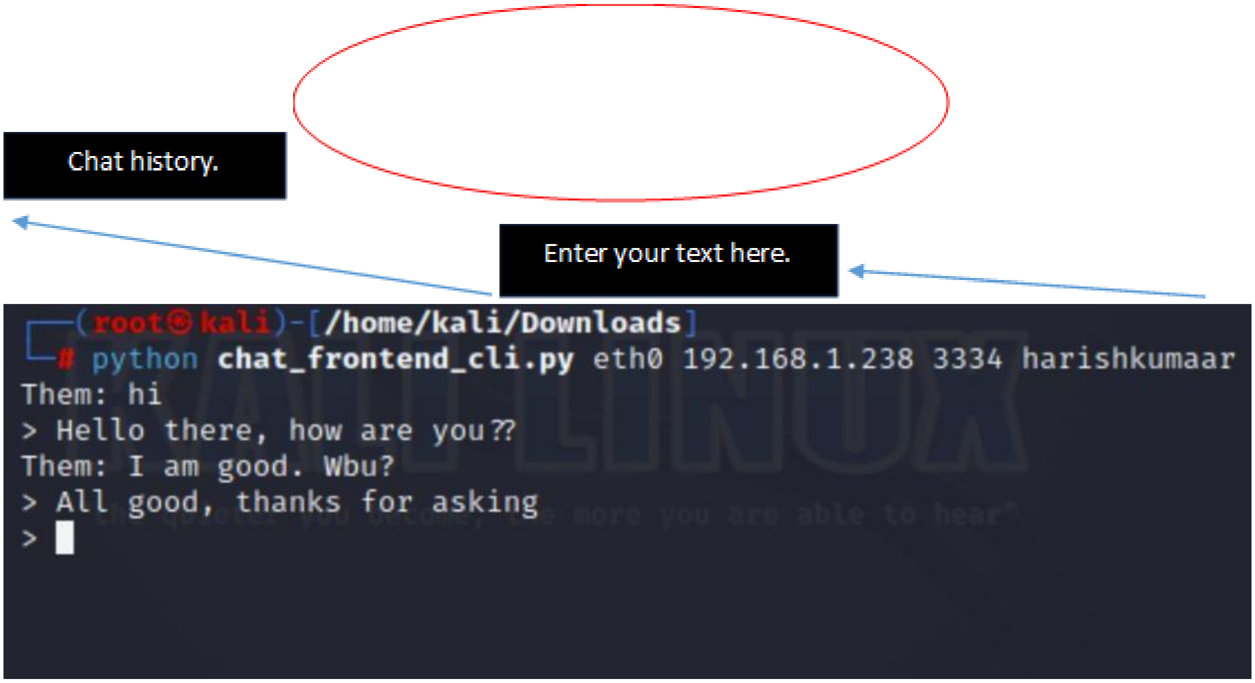
Tap on the text field located at the bottom of the chat window to send a message.

After typing your message, hit the "Send" button.



You: hi
Them: Hello there, how are you??
You: I am good. Wbu?
Them: All good, thanks for asking

Figure 24: Tkinter frontend interface



```
┌──(root㉿kali)-[/home/kali/Downloads]
└─# python chat_frontend_cli.py eth0 192.168.1.238 3334 harishkumaar
Them: hi
> Hello there, how are you??
Them: I am good. Wbu?
> All good, thanks for asking
> █
```

Figure 25: CLI frontend interface

As explained above, once the code is executed, you will be redirected to the user interface, which is either PyQt, Tkinter, or CLI depending on which code is executed. Once the connection is established and you type a message in the box highlighted in the images above. Once you have typed the desired message, click the send button for PyQt or tkinter interface, or just hit enter for the CLI interface. After that, your message will be sent to the desired receiver, and you can see the chat history in the interface as shown in the images above.

Conclusion

In summary, this Deployment and End User Manual presents a detailed guide for establishing a Python environment and operating CovertComm. It encompasses Python and Anaconda installation, conda environment configuration, and script execution instructions essential for the chat application's functionality. Whether users are novices or seasoned individuals, this manual furnishes clear, step-by-step guidance for a seamless setup process.

By adhering to the outlined instructions, users can proficiently install Anaconda, generate a CovertComm-compatible conda environment, and procure the requisite packages for code execution. Additionally, the manual offers insights into the chat application's mechanics, including interface navigation, message transmission, and chat history review.