|  |  |  |  |
| --- | --- | --- | --- |
| **Student Name :** | Lucky Pathan S. | | |
| **Enrollment No** | 210303105790 | **Branch :** | CSE |
| **Title of Journal Paper** | Socket Programming in Python (Guide) | | |
| **Authors** | Nathan Jennings | | |
| **Journal / Conference :** | Python Guide / web-dev | | |
| **Volume / Issue** | 1 | **Pages :** | 68 |

**Section – 01**

**Abstract / Introduction**

**Abstract:-**

**(1) What is the objective of the Paper?**

Purpose of the research paper is to make a :

* A simple socket server and client
* An improved version that handles multiple connections simultaneously
* A server-client application that functions like a full-fledged socket application, complete with its own custom header and content

**(2) What are the main results mentioned in the abstract?**

* Looked at the low-level socket API in Python’s socket module and saw how it can be used to create client-server applications
* Built a client and server that can handle multiple connections using a selectors object
* Created your own custom class and used it as an application-layer protocol to exchange messages and data between endpoints

**(3)** **What rational is given by the authors, attributing importance to the research problem?**

* The primary socket API functions and methods.
* To create a socket object using socket.socket(), specifying the socket type as socket.SOCK\_STREAM. The default protocol that’s used is the Transmission Control Protocol (TCP). This is a good default and probably what you want for its Is reliable and Has in-order data delivery.
* When using the loopback interface (IPv4 address 127.0.0.1 or IPv6 address ::1), data never leaves the host or touches the external network. In the diagram above, the loopback interface is contained inside the host. This represents the internal nature of the loopback interface and shows that connections and data that transit it are local to the host. This is why you’ll also hear the loopback interface and IP address 127.0.0.1 or ::1 referred to as “localhost.”

**Section – 02**

**Methodology**

**(1) Describe the methodology is used by author(s) to address the research problem?**

In this report, author teaches us the Python interface is a straightforward transliteration of the Unix system call and library interface for sockets to Python’s object-oriented style: the socket() function returns a socket object whose methods implement the various socket system calls. Parameter types are somewhat higher-level than in the C interface: as with read() and write() operations on Python files, buffer allocation on receive operations is automatic, and buffer length is implicit on send operations.

**(2) In what way the methodology used by the authors is relevant to the methodology you proposed to adopt?**

Based on research we use the Multi-Connection Client and Server feature of the socket API, which will help us to connect the users in a LAN network.

**Section – 03**

**Results and Conclusions**

**(1) List the results obtained by the authors.**

Explanation of a python package Argparse,

* Socket API Overview
* TCP Sockets
* Echo Client and Server
* Communication Breakdown
* Handling Multiple Connections
* Multi-Connection Client and Server
* Application Client and Server

**(2) What are the conclusions drawn by the authors from the study.**

Here we have covered a lot of ground in this tutorial! Networking and sockets are large subjects. New to networking or sockets. There are a lot of pieces to become familiar with in order to understand how everything works together. However, just like Python, it start to make more sense as you get to know the individual pieces and spend more time with them.

**Critical Analysis**

We get to learn of the low-level socket API in Python’s socket module and saw how it can be used to create client-server applications. Built a client and server that can handle multiple connections using a selectors object. Created your own custom class and used it as an application-layer protocol to exchange messages and data between endpoints. With this report we can create a socket server for the users to connect to the single network(LAN) for the log generation and the host data share of the bugs found.