MAJOR PROJECT REPORT

On

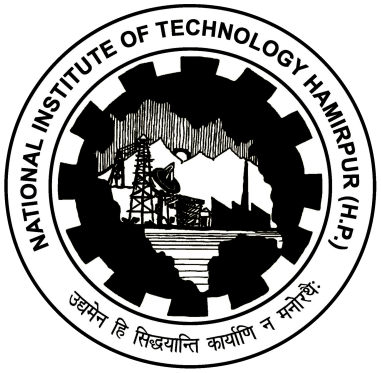
AGENT USING CLIENT SERVER MODULE

Submitted By:

Rishabh Chaudhary - 14MI508
Akhilesh Kumar - 14MI526
Rahul Kumar - 14MI538
Gourav Sisodiya - 14MI560

Under supervision of:

Dr. T.P. Sharma
Associate Professor
National Institute of Technology, Hamirpur



Computer Science and Engineering Department

National Institute Of Technology, Hamirpur (H.P.)
India - 177005

December 2017

DECLARATION

We, hereby, declare that this written submission represents our ideas in our own words and where others ideas or words have been included, We have adequately cited and referenced the original sources. We also declare that We have adhered to all principles of academic honesty and integrity and have not misrepresented or fabricated or falsified any idea/data/fact/source in our submission. We understand that any violation of the above will be the cause of disciplinary action by the Institute and can also evoke penal action from the sources which have thus not been properly cited or from whom proper permission has not been taken when needed.

Rishabh Chaudhary	**Akhilesh Kumar**	**Rahul Kumar**	**Gourav Sisodiya**
(14MI508)	(14MI526)	(14MI538)	(14MI560)

Date:
13/12/2017

ACKNOWLEDGMENT

*We take immense pleasure in thanking **National Institute of Technology, Hamirpur** for providing us the great chance for learning and development. Therefore, we consider ourselves very lucky being a part of this project.*

We consider ourselves privileged to express gratitude and respect towards all those who guided us through this project. It is with hearty gratitude that we acknowledge their contributions to our project.

*We would like to express our sincere gratitude and heartfelt thanks to **"Dr. T. P. Sharma, Associate Professor Computer Science and Engineering Department of National Institute of Technology, Hamirpur"** for his unflinching support and guidance, valuable suggestions and expert advice. His words of wisdom and expertise in subject matter were of immense help throughout the duration of this project.He has been a true mentor and without his support and guidance, the completion of this project would not have been possible.*

*We also take the opportunity to thank all the faculty of **Department of Computer Science and Engineering, National Institute of Technology, Hamirpur** for helping us by providing necessary knowledge base and resources.*

We perceive this opportunity as a big milestone in our career development. We will strive to use gained skills and knowledge in the best possible way, and We will continue to work on their improvement. Hope to continue cooperation with all of you in the future.

TABLE OF CONTENTS **Page No.**

FIGURE INDEX

ABSTRACT

System administrators have to maintain the computers by updating, upgrading them and installing the required softwares regularly. This job is in efficient and time consuming.

The aim of this project is to make an agent integrated with client and server which will help the admin to update/upgrade or install softwares/packages on every computer connected in a network by just issuing the agent with the list of tasks to perform on each computer, from his computer.

This software allows the admin to broadcast a file securely to all other computers connected on that network, which contains the series of commands/tasks to perform. The recipient computer will then extract the commands from the file received and then execute them one by one.

This project is developed by using Python and in this project concepts of Sockets, TCP/IP, UDP, Unicasting, Multicasting and Broadcasting are used.

***Chapter 1** Provides the brief introduction on The project "Agent using client server module".*

***Chapter 2** Will provide the brief about the technologies used.*

***Chapter 3** Will help getting started for the project.*

***Chapter 4** Describes the requirements of the project.*

***Chapter 5** Analyses the existing system and provides an insight about how the proposed system is better than the existing system.*

***Chapter 6** In this chapter we will do the analysis of the project on the basis of feasibility standards.*

***Chapter 7** In this chapter we will study about the design specifications of the project.*

***Chapter 8** Describes about how the system is being implemented and maintained.*

Chapter 9 *Describes the testing of the system. System testing is the testing of complete and fully integrated software product.*

Chapter 10 *In this chapter we will read about the different terminologies being used.*

Chapter 11 *Study and understanding the conclusion of the project.*

Chapter 12 *Includes Screenshots of the project.*

Chapter 13 *Summary of the project developed.*

Chapter 1. Introduction

1.1 Overview

This project named “Agent using client server Interaction”, is an open source project. It will allow the system administrator to perform certain tasks on the client systems connected on a network with just firing a command, which will then perform the set of desired tasks on all the systems. This set of desired tasks can be anything ranging from installing a particular software on systems to downloading and executing some available script.

This software can easily adapt to large network of computers and works efficiently for small networks.

1.2 Scope

This software can be used in any computers connected to a single network like a computer lab. The only constraint is that they all must run Linux based OS with python and Thefuck (package) installed.

The project uses the package Thefuck for error handling and error correction. It tries to match a rule for the previous command, creates a new command using the matched rule and runs the corrected command. Moreover, we can also create our own rule for the package and in that way it’ll become more effective and efficient according to our needs.

Also the error correction package we are using is open source and hence is under rapid development phase and also its available for all available major operating systems.

1.3 Objective

Usually the admin has to install or update the softwares in each and every computers in the lab. This task is inefficient and time consuming. We have developed this software to reduce the time consumed by almost 95% and do the required task efficiently will success rate greater than 95%. By using our program admin can issue a single or multiple commands which will automatically get executed in all the systems in the lab. The admin has to write those commands in a single file and then provide that file as input to our server.

Also, in case of network failure we can easily re issue the tasks to the client nodes or some specific nodes.

Chapter 2. Technologies used

2.1 Python :

Python is a high level programming language that lets you work quickly and integrate systems more effectively.

Python is an interpreted, interactive, object-oriented programming language that combines remarkable power with very clear syntax. The Python Library Reference documents built-in and standard types, constants, functions and modules.

Finally, the Python Reference Manual describes the syntax and semantics of the core language in (perhaps too) much detail.

Python's basic power can be extended with your own modules written in C or C++. On most systems such modules may be dynamically loaded. Python is also adaptable as an extension language for existing applications.[1]

2.2 Socket programming :

Socket programming is is a way of connecting two nodes or systems on a network to communicate with each other. One socket(node) listens on a particular port at an IP, while other socket reaches out to the other to form a connection. Server forms the listener socket while client reaches out to the server.[2]

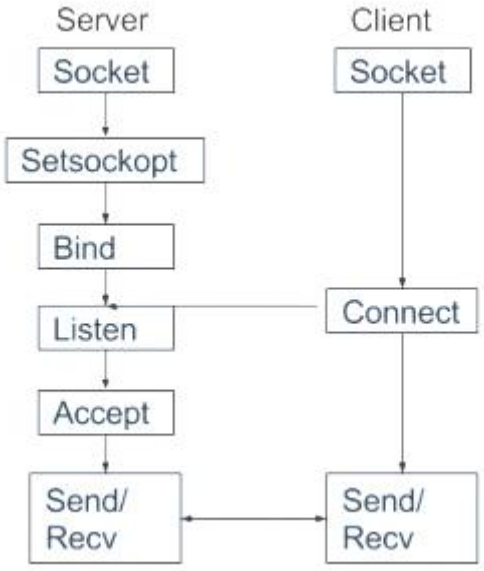


Figure 2.1 Socket Programming

Chapter 3. Getting Started

3.1 Introduction

This project is an open source project. It will allow the system administrator to perform certain tasks on the client systems connected on a network with just firing a command, which will then perform the set of desired tasks on all the systems.

This software can easily adapt to large network of computers and works efficiently for small networks.

3.2 Purpose

Usually the admin has to install or update the softwares in each and every computers in the lab. This task is inefficient and time consuming. We have developed this software to reduce the time consumed by almost 95% and do the required task efficiently will success rate greater than 95%. By using our program admin can issue a single or multiple commands which will automatically get executed in all the systems in the lab. The admin has to write those commands in a single file and then provide that file as input to our server. The file which is being given as input to server is a simple text file with one command/instruction per line.

This file contains the IP of the server on the last line, so in case its needed in case of major failure then the client can send the log file back to the server for analysis.

3.3 Scope

This software can be used in any computers connected to a single network like a computer lab. The only constraint is that they all must run Linux based OS with python and Thefuck (package) installed.

The project uses the package Thefuck for error handling and error correction. It tries to match a rule for the previous command, creates a new command using the matched rule and runs the corrected command. Moreover, we can also create our own rule for the package and in that way it'll become more effective and efficient according to our needs.

Also the error correction package we are using is open source and hence is under rapid development phase and also its available for all available major operating systems.

Chapter 4. Requirements

4.1 User interface

Each part of the user interface intends to be as user friendly as possible. Also its designed keeping it to be extensible in mind, therefore it has a terminal interface, as that interface is the most extensible interface possible and its also easy to use, keeping in mind the skills of network administrator. Also our project has an instant mode which just broadcasts the payload without asking for any other flags when enabled.

4.2 Functional requirements

- Less time consuming.
- Easy to implement and handle.
- Information stored in logs, in case of errors, logs can be referred.
- Data redundancy can be avoided to a great extent.
- Very less manual work.
- Fast and efficient.
- Works on all Linux environments.
- User friendly.
- Hassle free and chaos free working.
- Can be extended for monitoring systems.
- Secure transmission of data.

4.3 Non-functional requirements

- ***Security*** : The server shall only be accessible to authorized administrators.
- ***Availability*** : The system should be available whenever code needs to be executed and only restricted by the down time of the server on which the system runs.
- ***Maintainability*** : The authorized server takes care of the code execution. In case of a failure, a re-initialization of the server tasks will be done.
- ***Openness*** : This project can be easily be configured to perform some extended/advanced tasks/operations and that also without decreasing performance.
- ***Scalability*** : The project can easily handle more no. Of clients/nodes without increasing performance overhead.

Chapter 5. System Analysis

5.1 Existing System

In the existing system, the installations of any software on the systems in the lab is done manually which is very time consuming for the students as well as for the lab assistants. Also it is in efficient. The installation process on every system is very chaotic and students or lab assistants have to manually install the software which is not a very feasible thing to do.

In existing system, the process of the manual installation of softwares is time consuming and chaotic thus a new system is required which can do the required tasks with these properties:

- Easily
- Securely
- Efficiently

5.2 Proposed System

The proposed system is having many advantages over the existing system. It requires less overhead and is very efficient. The proposed system deals with the installation process efficiently because everything is done online from a dedicated server and server can run commands on remote hosts and execute the code on the hosts itself, thus making the new proposed system more convenient and user friendly.

Some of the advantages of new system over existing system:

- Less time consuming.
- More user friendly.
- More convenient.
- More transparent as compared to existing system.
- Hassle free and chaos free.
- Easy to implement and handle.
- Information stored in logs, in case of errors, logs can be referred.
- Data redundancy can be avoided to a great extent.
- Very less manual work.

Chapter 6. Feasibility Study

6.1 Technical feasibility

Technical feasibility assesses the current resources (such as hardware and software) and technology, which are required to accomplish user requirements in the software within the allocated time and budget.

The technical feasibility in the proposed system deals with the technology used in the system. It deals with the hardware and software used in the system whether they are of latest technology or not. It happens that after a system is prepared a new technology arises and the user wants the system based on that technology. This software uses Linux and Python 3.x along with package Thefuck (which is used for error correction).

6.2 Economical feasibility

This system doesn't requires any extra hardware and it is run on available computers of network administrator.

It is necessary to consider the benefits that can be achieved by developing the software so we can say that it is also economic feasible. Also this project is satisfying all the requirements for it to be economical feasible eg.:

- To analyze whether the software will meet organizational requirements.
- To determine whether the software can be implemented using the current technology and within the specified budget and schedule.
- To determine whether the software can be integrated with other existing software.

6.3 Operational feasibility

Operational feasibility assesses the extent to which the required software performs a series of steps to solve business problems and user requirements. The project has been developed in such a way that it becomes very easy even for a person to operate it. This software is very user friendly and does not require much technical person to operate . The user should only have knowledge of basics of Linux and it's commands and python (basics). Thus the project is even operationally feasible.

Chapter 7. System Design

7.1 Server

A server is a computer program that processes requests, provides services and deliver data to other computer programs either on same or other computers.The system in which server program runs is also referred to as a server. That machine may be only a dedicated server or can be used for other purposes too. But in our project server is the one which issues the commands and instructions to the clients.

Running the Server : *python3 server.py file.txt*

Output of the script : *Sending file....*
File sent

7.2 Client

A client is a computer program that, depends on sending a request to another computer program (that may or may not be located on another computer). But in our project the client is the one who is waiting for the instruction of server to execute/perform some desired task.

Running the client : *python3 client.py*

Output of the script : *File Downloaded*
No. of commands in file are : 1
List of commands and server IP :
who
10.42.0.1
Line 1: who
rishabh pts/0 2017-12-14 12:25 (linux)
[0]

7.3 Agent

An agent is an entity which takes the payload (data) to desired destinations. So, that the clients can perform the tasks issued by the server.

Chapter 8. Implementation

8.1 Introduction

Implementation is the stage in the project where the theoretical design is turned into a working system and is giving confidence on the new system for the users that it will work efficiently and effectively. It involves careful planning, investigation of the current system and its constraints on implementation, design of methods to achieve the change over, an evaluation of changed over methods. Apart from planning major task of preparing the implementation are education and training of users, the implementation process begins with preparing a plan for the implementation of the system. According to this plan, the activities are to be carried out, discussions made regarding the equipment and resources and the additional equipment has to be acquired to implement the new system. In network backup system no additional resources are needed.

Implementation is the final and the most important phase. The most critical stage in achieving a successful new system is giving the users confidence that the new system will work and be effective. The system can be implemented only after thorough testing is done and if it is found to be working according to the specification. This method also offers the greatest security since the old system can take over if the errors are found or inability to handle certain type of transactions while using the new system.

8.2 User training

After the system is implemented successfully, training of the user is one of the most important sub tasks of the developer. For this purpose user manuals are prepared and handled over to the user to operate the developed system. Thus the users are trained to operate the developed system. Both the hardware and software securities are made to run the developed systems successfully in future. In order to put new application system into use, the following activities were taken care of:

- Preparation of user and system documentation.
- Conducting user training with demo and hands on.
- Test run for some period to ensure smooth switching over the system.

The users are trained to use the newly developed functions. User manuals describing the procedures for using the functions listed on menu are circulated to all the users. It is confirmed that the system is implemented up to users need and expectations.

8.3 Security and maintenance

Maintenance involves the software industry captive, typing up system resources .It means restoring something to its original condition. Maintenance follows conversion to the extent that changes are necessary to maintain satisfactory operations relative to changes in the user's environment. Maintenance often includes minor enhancements or corrections to problems that surface in the system's operation. Maintenance is also done based on fixing the problems reported, changing the interface with other software or hardware enhancing the software.

Any system developed should be secured and protected against possible hazards. Security measures are provided to prevent unauthorized access of the database at various levels. An uninterrupted power supply should be so that the power failure or voltage fluctuations will not erase the data in the files. Password protection and simple procedures to prevent the unauthorized access are provided to the users .The system allows the user to enter the system only through proper username and password.

Chapter 9. System testing

9.1 Introduction

System testing is the stage of implementation, which is aimed at ensuring that the system works accurately and efficiently before live operation commences. Testing is the process of executing the program with the intent of finding errors and missing operations and also a complete verification to determine whether the objectives are met and the user requirements are satisfied.

The ultimate aim is quality assurance. Tests are carried out and the results are compared with the expected document. In the case of erroneous results, debugging is done. Using detailed testing strategies a test plan is carried out on each module. The various tests performed are unit testing, integration testing and user acceptance testing.

9.2 Unit testing

The software units in a system are modules and routines that are assembled and integrated to perform a specific function. Unit testing focuses first on modules, independently of one another, to locate errors. This enables, to detect errors in coding and logic that are contained within each module. This testing includes entering data and ascertaining if the value matches to the type and size supported by java. The various controls are tested to ensure that each performs its action as required.

9.3 Integration testing

Data can be lost across any interface, one module can have an adverse effect on another, sub functions when combined, may not produce the desired major functions. Integration testing is a systematic testing to discover errors associated within the interface. The objective is to take unit tested modules and build a program structure. Here the Server module and Client module options are integrated and tested. This testing provides the assurance that the application is well integrated functional unit with smooth transition of data.

9.4 User acceptance testing

User acceptance of a system is the key factor for the success of any system. The system under consideration is tested for user acceptance by constantly keeping in touch with the system users at time of developing and making changes whenever required.

Chapter 10. Terminologies used

10.1 Server

A server is a computer program that processes requests, provides services and deliver data to other computer programs either on same or other computers.The system in which server program runs is also referred to as a server. That machine may be only a dedicated server or can be used for other purposes too. But in our project server is the one issuing commands and tasks to clients using an agent.

10.2 Client

A client is a computer program that, depends on sending a request to another computer program (that may or may not be located on another computer). But in our project the client is the one who is waiting for the instruction of server to execute/perform some desired task.

10.3 Unicast

The term Unicast is used to describe communication where information is sent from a point to another point. In case of unicast, there is just one receiver, and one sender. It can also be understood as unidirectional/bidirectional point to point communication.

10.4 Multicast

Multicast is the term used to describe communication when some information is sent from one or more points/nodes to a finite set of other points/nodes. In Multicast, there may be one or more senders, and the information is distributed to a set of receivers.

10.5 Broadcast

The term Broadcast is used to describe communication where information is sent from one point/node to all points/nodes. In broadcast there is just one sender, but the information is sent to all connected receivers/nodes.

10.6 Socket

A network socket is an internal endpoint for sending or receiving data at a single node in a computer network. We can also see it as a combination of IP address and port of a particular node.

10.7 Port

A port is an endpoint of communication in an operating system.

10.8 TCP/IP

Transmission Control Protocol/Internet Protocol is a reliable protocol used for communication. In this protocol it is guaranteed that the packets will reach their destination. It is not as fast as UDP but is surely more reliable without any doubt.

10.9 UDP

User Datagram Protocol(UDP) is an unreliable protocol used for communication. There is no guarantee that packets will be delivered. But it is faster than its reliable counterpart TCP (Transmission Control Protocol).

Chapter 11. Conclusion

We conclude that our software can be used to install, update, upgrade or remove software in the computers connected in a network. To do this the admin need to create a file containing the commands to update/upgrade or install/remove the software and then using our software she need to multicast/broadcast it from the server, then the data will reach its desired destination through an agent. The client computers then extracts the commands from the file then executes them one by one.

This whole process of transmission is being secured using PGP encryption.
In this software the server will digitally sign the file being sent to be multicast/broadcast. On reception of the file the recipient will decrypt it with the help of the public key provided to it prior to the execution of the software.

Our software is also very easy to use. User has to only multicast/broadcast the file containing the command through our software, then our software will take care of the rest. The user must have a basic knowledge of Linux commands and how to run a python program that's all and if there is any error in those commands, then our software will automatically correct it and execute the correct command.

Chapter 12. Screenshots

Figure 12.1 Broadcasting from server

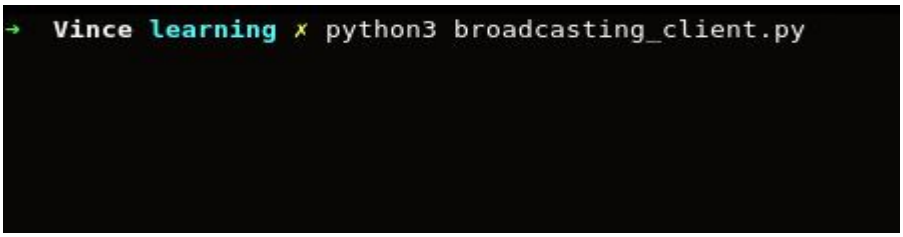


Figure 12.2 Client on reception mode

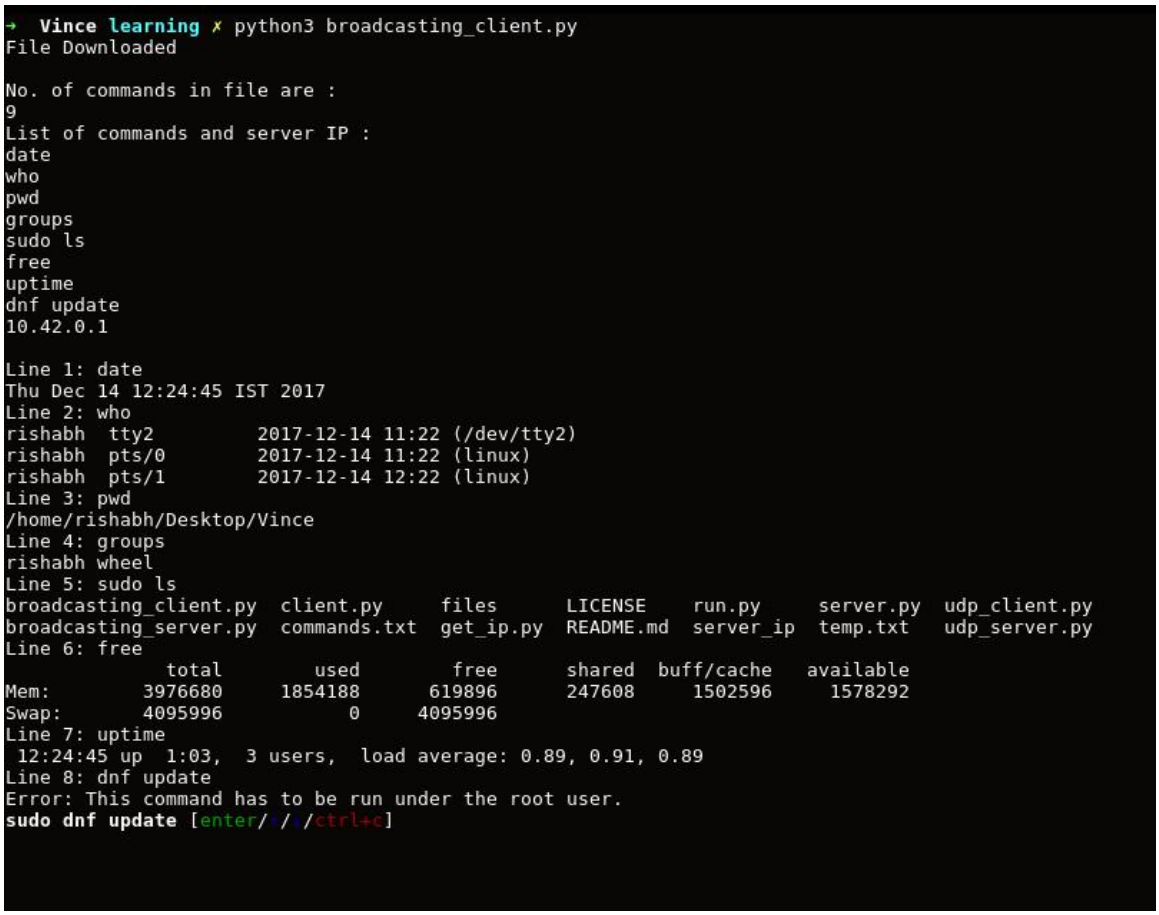


Figure 12.3 Synchronous execution with error detection

Figure 12.4 Error correction

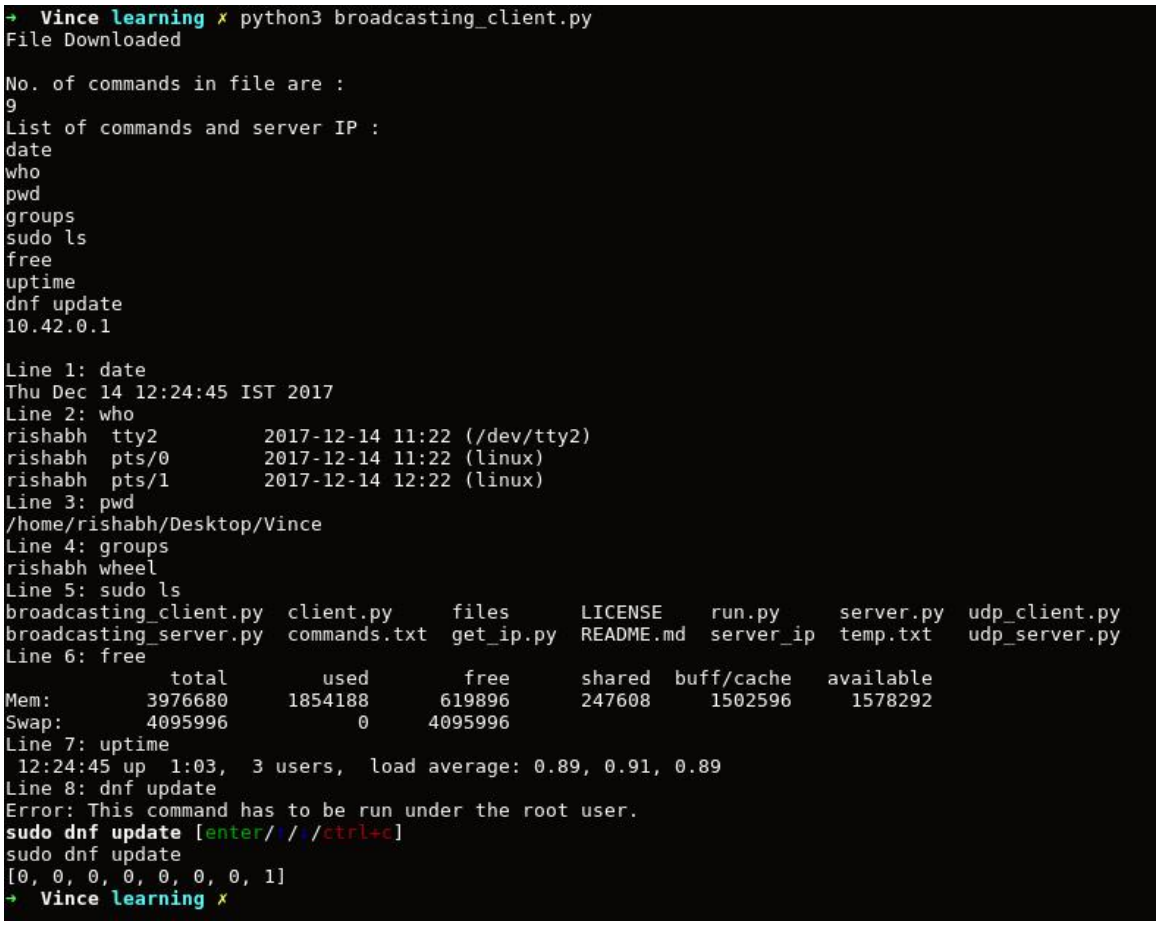


Figure 12.5 Final execution with error correction

Chapter 13. Summary

The expanded functionality of today's software requires an appropriate approach towards software development. The project is open source project.
It can be used to perform those tasks easily which were really tiring and time consuming as well we were following an inefficient way to do those tasks.

This software can easily adapt to large network of computers and works efficiently for small networks. This project is satisfying all the requirements for it to be economical feasible. The project has been developed in such a way that it becomes very easy even for a person to operate it. Our software can be used to install, update, upgrade or remove software in the computers connected in a network.

Also this whole process of transmission is being secured using PGP encryption.
This software is also very easy to use and manage and do not require any additional resources to be allocated to it.

The project, is developed using Python programming language and is based on the requirement specification of the user and the analysis of the existing system, with flexibility for future enhancement.

References

[1] https://docs.python.org

[2] http://www.geeksforgeeks.org/socket-programming-cc/

[3] George Coulouris, Jean Dollimore, Tim Kindberg, Gordon Blair "DISTRIBUTED SYSTEMS Concepts and Design, Fifth Edition" 2012, Pearson

[4] William Stallings, "Cryptography and Network Security Principles and Practice, Sixth Edition", 2014, Pearson

[5] Kohl, J.; Neuman, B.; and Ts'o, T. "The Evolution of the Kerberos Authentication Service." in Brazier, F., and Johansen, D. Distributed Open Systems. Los Alamitos, CA: IEEE Computer Society Press, 1994.

[6] Tung, B. Kerberos: A Network Authentication System. Reading, MA: Addison-Wesley, 1999

[7] Balazinska, M., Balakrishnan, H. and Karger, D. (2002). INS/Twine: A scalable peer-to-peer architecture for intentional resource discovery. In Proceedings of the International Conference on Pervasive Computing, Zurich, Switzerland, August, pp. 195–210.

[8] Baldoni, R., Beraldi, R., Quema, V., Querzoni, L. And Tucci-Piergiovanni, S. (2007). TERA: Topic-based event routing for peer-to-peer architectures. In Proceedings of the 2007 Inaugural International Conference on Distributed Event-Based Systems. Toronto, Ontario, Canada, June, pp. 2–13.

[9] Baset, S.A. and Schulzrinne, H.G. (2006). An analysis of the Skype peer-to-peer Internet telephony protocol. In Proceedings of the 25 th IEEE International Conference on Computer Communications (INFOCOM'06), pp. 1–11.