

# Performance Analysis of HTTP and FTP Based on OPNET

Pan Wei

School of Computer Science  
Communication University of China  
weipan@cuc.edu.cn

Zhiguo Hong, Minyong Shi

School of Computer Science  
Communication University of China

**Abstract**—in this paper, the performance of HTTP protocol and FTP Protocol is analyzed by using OPNET-based simulation. Firstly, related scenarios are modeled with OPNET Simulator. Furthermore, two case studies are conducted by choosing average queuing delay and TCP delay as performance indices respectively. In the case of HTTP protocol's performance, the changing trend with the increase of the number of clients is reported. Additionally, in the case of comparison analysis, TCP delays of HTTP and FTP protocols are studied with the connection of 5 clients. On the basis of this work, performances of related network system with multiple traffics can be conducted.

**Keywords**—HTTP; FTP; OPNET

## I. INTRODUCTION

Hypertext Transfer Protocol HTTP is a simple request-response protocol [1], widely due to transfer information between Web servers and clients. It offers efficient way of transferring documents according to the related specifications. HTTP protocol is an application layer communication protocol for client browser or other program and the Web server [2], this protocol specifies what kind of message the client can send to the server, and what kind of response to the message. FTP (File Transfer Protocol) is one of the protocols in the TCP/IP protocol suite [3]. FTP protocol includes two components, one for the FTP server and the other for FTP client. FTP client can send commands to the server to download files, upload files, create or change the directory on the server. HTTP and FTP protocols are used to request/response pattern, and also work in TCP (Transmission Control Protocol) protocols. How to discover the differences between HTTP and FTP with various web applications is an important problem. Results on comparison will give instructive guide for both web servers and clients. In this paper, we use OPNET simulation platform to design different scenarios and make HTTP and FTP protocols-based comparative analysis to investigate the Web application's performance respectively.

The rest of this paper is organized as follows. In section II, it systematically introduces the characteristics and functions of HTTP and FTP. Section III describes the design of the simulation in detail. Section IV combined with the characteristics of HTTP and FTP. Section V concludes this paper.

## II. STANDARDS OF APPLICATION IN OPNET

OPNET software offers user-friendly GUI (graphical user interface), which provides convenient operations to build scenarios, set parameters of networks and test the performance with simulation[4]. It supports HTTP and FTP-based applications and users can establish various scenarios containing multiple servers and clients easily. Also it offers the modeling and analysis of two-tier, request/response -based web applications.

### A. HTTP Protocol

OPNET tool can simulate HTTP application as the browsing activities of Internet, i.e. one of the client nodes periodically contact Web server to retrieve web pages [5].

In OPNET, each page is modeled as a text (a basic HTML file) and a few inline objects (such as images and data files referenced objects) of a combination. HTTP uses TCP as the transport protocol, the running process is as follows: the client sends a HTTP request to a Web page; the server receives the request, and the corresponding page will be replied to the client. In this analysis process, if the Web page contains several inline objects, the client node will request for the follow-up objects from the Server. In OPNET, HTTP Configuration Table (HTTP configuration table) mainly includes four key attributes of describing HTTP application: HTTP Specification defines the configuration of Web browser; Page Interarrival Time (seconds) specifies the time between page requests in a row; Page Properties describes an OPNET said of a typical Web page; Server Selection decides the retrieving path by parsing an embedded Web link index page.

### B. FTP Protocol

OPNET standard application for FTP models the basic operations [6]. OPNET simplify the multiple FTP applications and simulate the two main data transmission of FTP operation: get and put. FTP put operations are used to upload a file to FTP server and FTP get operation are used to download a file from the FTP server to the client nodes. In OPNET, the FTP application models the process of transferring files at a time. By default, FTP is running on top of the TCP. When using the default transport protocol, it opens a separate TCP connection for each file. Unlike in real networks, during the procedure of

one file's transferring operations, the FTP model uses the same TCP connection to send control messages and data modeling [7]. OPNET FTP is described in three key properties: Command Mix (Get/Total) specifies the ratio between the get operands and the total number of FTP operations[8]; Inter-Request Time (seconds) to specify the amount of time between consecutive FTP operation; File Size (bytes) specifies that you want to transfer the file size in bytes.

### III. SIMULATION EXPERIMENTS DESIGN

This experiment is based on simulation software OPNET 17.5 version to compare and analyze performance of HTTP protocol and FTP protocol [9]. The simulation experiment includes two parts: the first part of simulation content is to test the performance of HTTP protocol in different network bandwidth and different number of concurrent access client; the second part of simulation content is to compare the transmission performance of HTTP and FTP in same network environment [10].

#### A. HTTP Protocol Performance Analysis

This section contains three groups, and each group contains three scenarios. The number of client nodes and the network bandwidth is different in each scene. We set the packet loss rate of the Internet cloud in each scenario be 1%. Also all through the traffic on the introduction of an additional delay is set to be 200ms. Set the web browser of every scene running on top of HTTP1.1, and configure each client node requests a new page every 10s. Set the size of each web page to 100 bytes, which is consist of a single text object and 7 large images. The Web application configuration is shown as TABLE I. as follows, and the network topology details table of these nine scenes is shown as TABLE II. as follows:

TABLE I. WEB CONFIGURATION TABLE

HTTP	Property	
	HTTP specification	Page property
Parameter	HTTP 1.1	100B single text objects and 7 large embedded images

TABLE II. HTTP CONFIGURATION TABLE

Scene	Network Topology	
	Client Number	Client<->The Internet Server<->The Internet
1	10	Ppp_28k
2	20	Ppp_28k
3	30	Ppp_28k
4	10	Ppp_33k
5	20	Ppp_33k
6	30	Ppp_33k
7	10	Ppp_56k
8	20	Ppp_56k

Scene	Network Topology	
	Client Number	Client<->The Internet Server<->The Internet
9	30	Ppp_56k

The network topology of scenario 2 is shown as Fig.1. as follows:

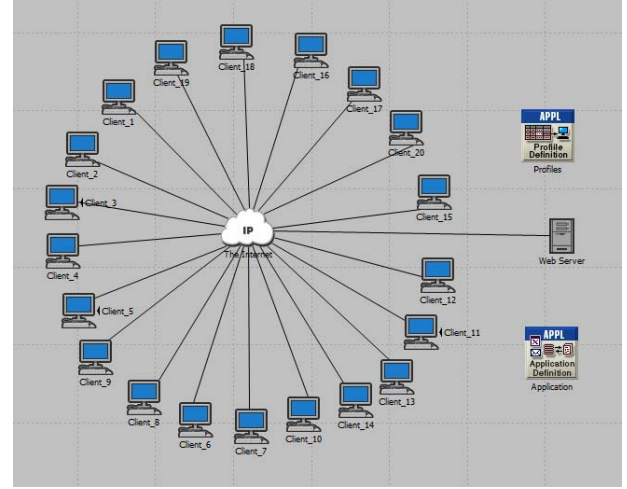


Fig.1. Network Topology of Sence 2

#### B. FTP Protocol Performance Analysis

The simulation model contains two scenes in this group, in the first scene the Web application is the HTTP protocol, the second one is FTP protocol, the network topology of the two scenes are observed as Fig.2. as follows:

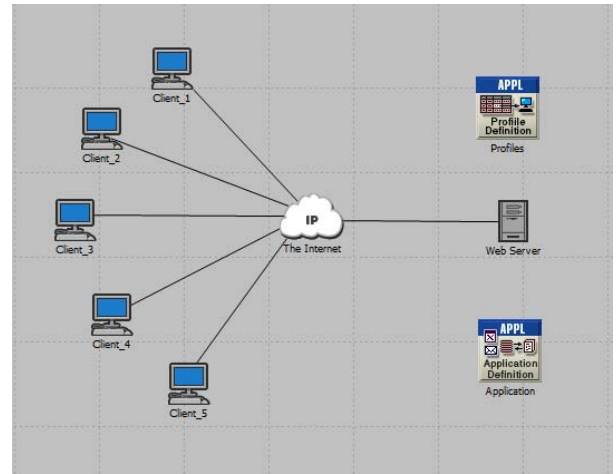


Fig.2. Network Topology

These two scenes of network topology group scene are much the same with group\_1, the only difference is that the link object is set to ppp\_DS1. The Web application configuration of HTTP scene is shown as TABLE III. and the Web application configuration of FTP scene is shown as TABLE IV. as follows:

TABLE III. HTTP CONFIGURATION TABLE

HTTP	Property	
	<i>HTTP specification</i>	<i>Page property</i>
Parameter	HTTP 1.1	100B single text objects and 7 large embedded images

TABLE IV. FTP CONFIGURATION TABLE

FTP	Property		
	<i>Command Mix</i>	<i>File Size</i>	<i>Symbolic Server Name</i>
Parameter	50%	100 bytes	FTP Server

#### IV. EXPERIMENT RESULTS ANALYSIS

##### A. HTTP Protocol Performance Analysis

OPNET provides various kinds of simulation and statistics, as well as the ability to configure the collection mode, but a simulation in OPNET does not automatically select which statistics to collect. In order to validate different types of network bandwidth and request the client node number effects on the performance of HTTP protocol, we need to manually configure the simulation in each scene to collect the following statistic: all Client HTTP node statistics, all point-to-point link statistics. And then repeatedly perform the simulation in the results browser view and collect the results. We collect the queuing delay in the link from client\_1 to the Internet in every scene, and then export the result to an excel file. From these useful data, we calculate the average Queuing delay in every scene. Comparison of the simulation results are shown as Fig.3. as follows:

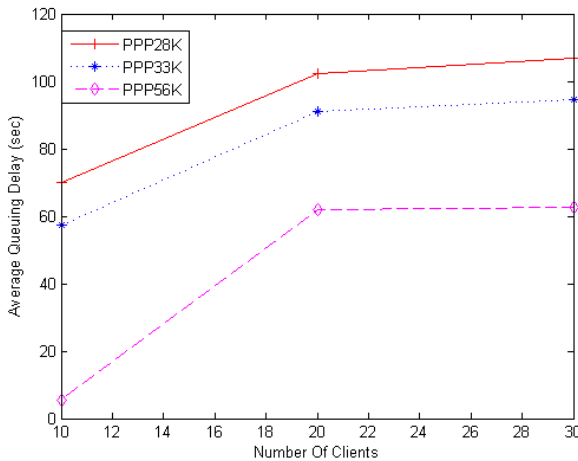


Fig.3. HTTP Performance Comparison Chart

From Fig. 3, we can observe that with the increase in the number of client nodes, the end-to-end queuing delay of the

client node to the server increases. With the increasing of link bandwidth, the end-to-end queuing delay of the client node to the server decreases. When the number of the clients is increased from ten to 20, the end-to-end queuing delay increased by a big margin, as the number of the clients is increased from 20 to 30, the end-to-end queuing delay increased relatively less. Thus we get the following results: client node number of HTTP requests made more big effect on end-to-end queuing delay in the link, HTTP protocol properties relatively made smaller effect. The larger the link bandwidth are, the smaller the end-to-end queuing delay will be, and the performance of HTTP protocol will be better accordingly.

##### B. Performance Results analysis of HTTP and FTP

In order to verify the performance of HTTP and FTP protocol in the same network environment, we configure the set of simulation to collect statistics for two scenarios: all nodes from the class node of the statistics for TCP. After you run the simulation, collecting and comparing results. The results of this group are shown as Fig.4. as follows:

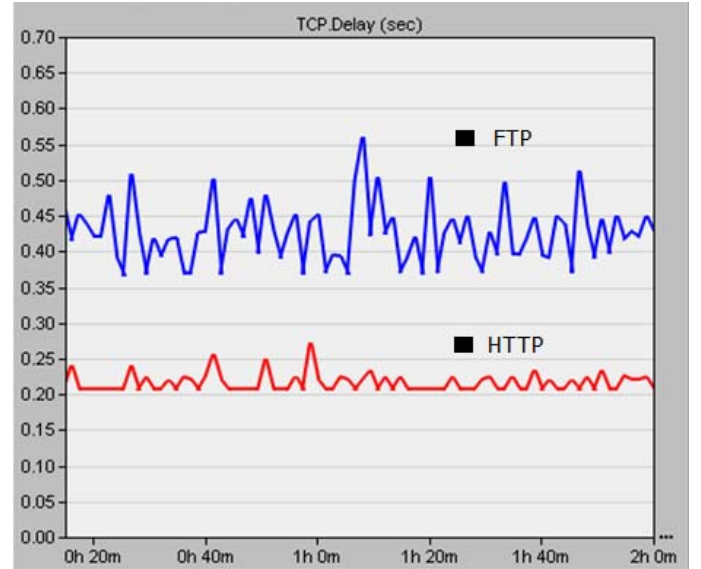


Fig.4. TCP Delay of HTTP and FTP

We collect the delay in each scene, and export the result to an excel file. Then we calculate the average delay in each scene..

Case 1: In the case of client number=5, link =ppp\_DS1, Web application = HTTP 1.1, the average delay is 0.21s.

Case 2: In the case of client number=5, link = ppp\_DS1, Web application = FTP, the average delay is almost 0.43s.

From Fig. 4, we can observe that: In the same network conditions, it can be seen that the average delay of HTTP is 200% of FTP. In a word, under the same network environment, the performance of the HTTP protocol performance is better than the FTP protocol.

## V. CONCLUSION

This paper is based on OPNET, and completes a simulation of HTTP protocol and FTP protocol. By comparing the queuing delay in different scenarios based on HTTP protocol, we find what do the link bandwidth and client node number effect on the performance of HTTP; through comparing the TCP Delay in HTTP protocol and FTP protocol scene, we learned the performance of FTP protocol and HTTP protocol on Web application under the same network environment. All the analysis are based on simulation. On the basis of this study, performances of related network system with multiple traffics can be conducted.

## ACKNOWLEDGMENT

This paper is partly supported by “the Excellent Young Teachers Training Project (the second level, Project number: YXJS201508)”, “Key Cultivation Engineering Project of Communication University of China (Project number: 3132016XNG1606 and 3132016XNG1608)”, “Cultural technological innovation project of Ministry of Culture of P.R.China (Project number: 2014-12)”, and partly supported by “The comprehensive reform project of computer science and technology, department of science and Engineering”. The research work was also supported by “Chaoyang District Science and Technology Project (CYXC1504)”.

## REFERENCES

- [1] J. Evangelista. Netto, E. Hernane. Paulicena, R. Amorim. Silva, A. Anzaloni. “Analysis of Energy Consumption Using HTTP and FTP Protocols Over IEEE 802.11,” IEEE Journals & Magazines, 2014, 12(4): 668-674
- [2] L.D.Circiumarescu,G.Predusca, N.Angelescu, D.Puchianu. “Comparative Analysis of Protocol RIP, OSPF, RIGRP and IGRP for Service Video Conferencing, E-mail, FTP, HTTP,” IEEE Conference Publications, 2015:584-589.
- [3] Andrew.S.Tanenbaum, David.J.Wetherall. “COMPUTER NETWORKS,” Tsing University Press, 2013.
- [4] Adarshpal.S.Sethi, Vasil.Y.Hnatyshin. “The Practical OPNET User Guide for Computer Network Simulation,” CHINA MACHINE PRESS, 2014, 1.
- [5] R. Fielding, UC Irvine. “RFC 2616: Hypertext Transfer Protocol -- HTTP/1.1,” IETF, June 1999.
- [6] J. Postel, J. Reynolds. “RFC 959: FILE TRANSFER PROTOCOL (FTP), ISI, October 1985.
- [7] M. Allman, V. Paxson. “RFC 5681: TCP Congestion Control,” ICSI, September 2009.
- [8] Manju Sharma, Manoj Kumar, Ajay.K. Sharma. “Http And Ftp Statistics For Wireless And Wire-Line Network With And Without Load Balance Based On Opnet,” Institute for Scientific Computing and Information, international journal of information and systems sciences Volume 5, Number 1, Pages 112-125©2009
- [9] Manju Sharma, Manoj Kumar. “Comparative investigation on Media Access Delay and Client Object response Time for a Switched and Routed Wireless LAN based on OPNET,” (2008)
- [10] IT Guru Academic Edition OPNET Technologies, [http://www.opnet.com/University\\_program/](http://www.opnet.com/University_program/) it guru academic edition,(2007)