

SANDVINE TECHNOLOGIES

PREPARED BY

Shreya Singh (2018BTECHCSE119)

FACULTY GUIDE

Dr. S. Taruna



**Department of Computer Science & Engineering
Institute of Engineering and Technology (IET)
JK Lakshmipat University, Jaipur**

May-2022

SANDVINE TECHNOLOGIES

Practice School-2

Submitted in partial fulfilment of the requirements

for the degree of

Bachelor of Technology in Computer Science and Engineering

By:

Shreya Singh (2018BTECHCSE119)

Faculty Guide:

Dr. S. Taruna

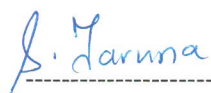


**Department of Computer Science & Engineering
Institute of Engineering and Technology (IET)
JK Lakshmipat University, Jaipur**

May-2022

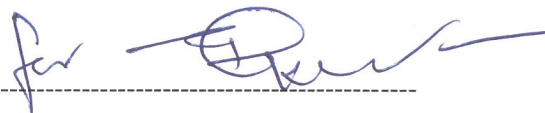
CERTIFICATE

This is to certify that the Practice School-2 project work entitled **Sandvine Technologies Pvt Ltd., Bangalore** submitted by **Shreya Singh (2018BTechCSE119)**, towards the partial fulfillment of the requirements for the degree of **Bachelor of Technology in Computer Science and Engineering** of JK Lakshmipat University Jaipur is the record of work carried out by them under my supervision and guidance. In my opinion, the submitted work has reached a level required for being accepted for Practice School-II examination.



Dr. S. Taruna

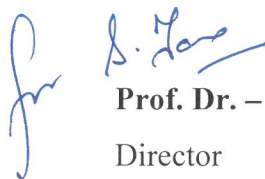
Associate Professor,
Department of Computer Science
Engineering
Institute of Engineering & Technology
(IET)
JK Lakshmipat University, Jaipur



Dr. Sonal Jain

Co-ordinator and Professor,
Department of Computer Science
Engineering
Institute of Engineering & Technology
(IET)
JK Lakshmipat University, Jaipur

Date of Submission: 13-05-2022



Prof. Dr. – Sanjay Goel

Director
Institute of Engineering & Technology
JK Lakshmipat University Jaipur

ACKNOWLEDGEMENTS

The internship opportunity I had with **Sandvine Technologies, Bangalore** was a great chance for learning and professional development. Therefore, I consider myself as a very lucky individual as I was provided with an opportunity to be a part of Sandvine. I am also grateful for having a chance to meet so many wonderful people and professionals who led me through this internship period.

I perceive as this opportunity as a big milestone in my career development. I have taken efforts in completing this internship. However, it would not have been possible without the kind support and help of many individuals and organizations. I would like to express heartfelt gratitude to **Prof Dr.Sanjay Goel**, Director, Institute of Engineering & Technology for providing this opportunity to me. Also, I would like to thank my faculty guide **Dr. S. Taruna** for her constant support and guidance.

I am highly indebted to **Sandvine Technologies** for their guidance and constant supervision as well as for providing necessary information regarding the project & also for their support in completing the project.

I would like to express my gratitude towards the members of Sandvine Technologies for their kind co-operation and encouragement which helped me in the completion of this project. I would like to express my gratitude and thanks to industry mentors **Mr. Manoj Mallappa, Mr. Ankit Chhibber, Mr. Nilanjan Sarkar and Mr. Neelesh Dwevedi** for giving me such attention and time.



Sincerely yours,

Shreya Singh
(2018BTechCSE119)

ABSTRACT

The transmission control protocol (TCP) is a widely used Internet transport layer protocol. It provides reliability, flow control, and congestion control services. TCP has a number of control methods that decide when a packet is considered lost, when it should be retransmitted, and when it should be slowed down. Many components of the TCP protocol have been fine-tuned over the years to improve overall TCP performance incrementally. But there are some issues with this protocol which are addressed by TCP Accelerator. TCP Accelerator solves the most frequent TCP problems, such as bandwidth availability and TCP speeds being incompatible. It also overcomes inefficiencies caused by several concurrent TCP connections by transparently bridging the access network with the transit network; it can manage all of the network's TCP connections as a group.

LIST OF FIGURES

Figure	Page No.
Figure 2.1.1: Congestion Control Algorithmn in TCP	3
Figure 2.1.2: Layer diagram for TCP describing network operators control over it	4
Figure 2.2.1: Basic flow of TCPA	5
Figure 2.2.2: Basic flow of activelogic	6
Figure 2.4.4: Graph showing Goodput Improvements due to TCPA	9
Figure 2.5.1: Graph showing efficient data transfers due to TCPA	10
Figure 2.5.2: Graph showing faster and consistent RTTs due to TCPA	11
Figure 2.5.1: Graph showing decrease in retransmission rates due to TCPA	11
Figure 2.5.2: Graph showing increase in network efficiency due to TCPA	12

LIST OF ACRONYMS

- **TCP:** Transmission control protocol
- **TCPA:** Transmission control protocol accelerator
- **QoE:** Quality of experience
- **MOCN:** Multi-Operator Core Network
- **UT:** Unit testing
- **CDMA :** Code-division multiple access

TABLE OF CONTENT

CERTIFICATE	i
ACKNOWLEDGEMENTS	ii
ABSTRACT	iii
LIST OF FIGURES	iv
LIST OF ACRONYMS	v
1. Introduction	1-2
1.1 About the organization	1-2
2. About the feature	3-12
2.1 Introduction to TCP protocol	3-4
2.2 Need of TCP Accelerator	4-6
2.3 Key features of TCP Accelerator	6-9
2.3.1 Transparency	6
2.3.2 Powerful TCP Acceleration Techniques	6-7
2.3.3 TCP Buffer Management	7
2.3.4 QUIC Traffic Control	8
2.3.5 Egress Burst Control	8
2.3.6 Universal Access Support	8
2.3.7 Audit Records and Historic Reporting	8
2.4 Benefits of TCP Accelerator	8-10
2.4.1 Taking control of layer-4	8
2.4.2 Accelerate TCP Slow Start to reach maximum speed faster	9
2.4.3 Managing Packetloss in the Access network	9
2.4.4 TCP Goodput Improvements	9
2.4.5 Managing Congestion in Access network	10
2.5 Performance Comparison of TCPA and TCP Protocol	10-12
2.6 Summary	12

3. Trainings and Feature Work	13-20
3.1 Training Workflow	13-15
3.1.2 General Trainings	13-14
3.1.3 Product Trainings	14-15
3.2 Tools and Technologies	15-17
3.3 Assigned Tasks	17-19
3.4 Gantt chart	20
4. Conclusion	21
References	22

CHAPTER 1: INTRODUCTION

1.1 About the organization



Sandvine Technologies serves as a leader in network and application intelligence.

With the help of contextual machine learning based insights and real time action, Sandvine's cloud-based Application and Network Intelligence helps its customer to analyze optimize and monetize application experiences. The market-leading classification of over 95% of traffic across mobile and fixed networks by user, application, device, and location significantly improves user-application interactions.

Their cloud-ready platform can be deployed on any hardware, and our data APIs can feed into other back-end decisioning systems that accelerate putting your network data to work for the business. This makes networks more efficient, enables service differentiation, rationalizes disparate data sources, and increases profit margins.

With 2.1 billion network users across more than 150 major operators and 500 enterprises benefiting from using Sandvine as their network and application intelligence innovator and partner in the next wave of network transformation.

Sandvine helps customer deliver exceptionally personalized services to improve the network performance by combining some real time contextual insights, prepackage use cases, customers quality of experience and network visibility in convenient. Active and passive intelligence allows for prompt action. Prioritizing and monitoring application and network traffic, maintaining quality of experience, controlling congestion, cyberthreats, and fraud, and adhering to regulations are just a few examples.

Sandvine's customers include communications service providers, governments, and businesses from all around the world. Their service provider solutions enable companies to provide outstanding application and network experiences by providing visibility and actionable information into network utilisation and application performance.

Their government solutions are designed to serve and protect citizens, secure infrastructure, and create a fair and competitive market environment in a contemporary, connected society. Their enterprise solutions aid in the management of rapidly growing network traffic, shifting traffic patterns, application usage and location patterns, and new cyber security risks.

Website	http://www.sandvine.com/
Industry	Networking Hardware and Software
Company size	501-1000 employees
Headquarters	Waterloo, ON, Canada
Type	Private
Founded	2001
Specialties	Network Analytics, Network Optimization, Network Revenue Generation, Network Security, Global Services, Internet Trends, Virtualization, NFV, Telecom Regulatory Compliance, Telecom Revenue Assurance, and Cloud Services

CHAPTER 2: ABOUT THE FEATURE

2.1 Introduction to TCP Protocol

TCP is a layer 4 connection-based protocol in which a client and server establish a connection before data is sent through an initial handshake. TCP uses a variety of strategies to prevent packet loss and ensure data delivery, including retransmission. TCP only cares about providing error-free data from start to finish.

TCP has four major approaches to assist it manage its network utilisation, avoid overloading them by demanding too much bandwidth, and respond to congestion, packet loss, and latency known as congestion control methods defined as:

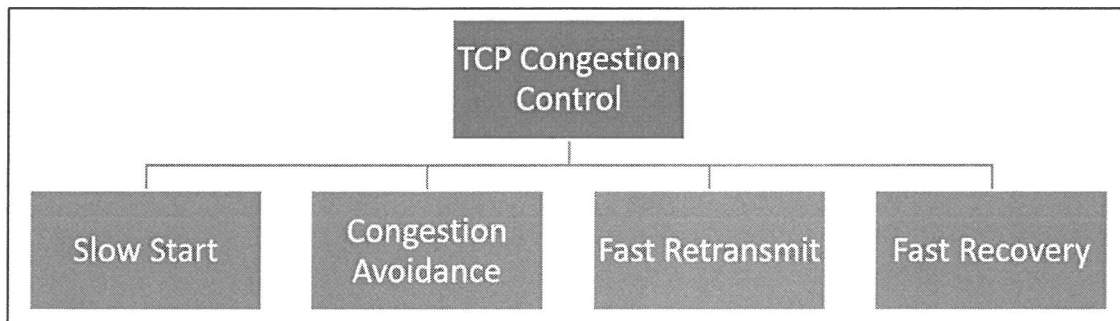


Figure 2.1.1: Congestion Control Algorithmn in TCP

1. Slow start is a technique for starting a connection slowly so that it does not consume too much bandwidth and generate congestion.
2. Congestion avoidance focuses on improving loss or bandwidth throughput and include a variety of subjects such as loss or delay.
3. Fast retransmit lowers the time spent waiting, when a retransmit is required.
4. Fast recovery is combined with fast retransmission to help ensure the connection stabilises quickly.

There are some key considerations about this TCP network protocol:

1. For the various algorithms to work, they should be supported by all vendors in the whole end to end connection.
2. Since client-server connections are used, each error and subsequent recovery affects a significant portion of the network and is not limited to a single issue region. For example, because a retransmit must originate from the server, last-mile packetloss affects the entire flow.

In the 7-layer OSI model, a network operator normally has control over their network from the actual components to the router. But controlling the layers above these few is generally very difficult. There is a situation where more than 80% of traffic is using a protocol they cannot easily control, let alone optimize.

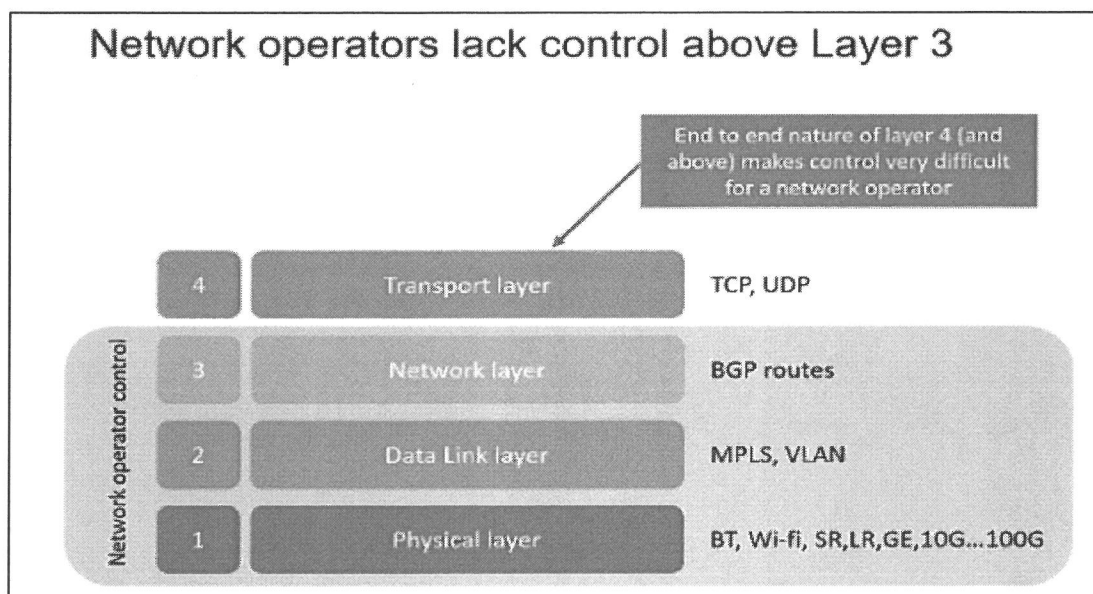


Figure 2.1.2: Layer diagram for TCP describing network operators control over it

The nature of how TCP runs causes three major problems:

- TCP may not be fast enough when more bandwidth is available.
- TCP may be excessively fast when there is insufficient bandwidth.
- The collection as a whole behaves inefficiently when there are several concurrent TCP connections.

For operators, these problems create significant consequences:

- Network efficiency is harmed, causing performance and return on investment to fall below expectations.
- Application QoE is impacted, decreasing customer satisfaction and potentially increasing churn.

Sandvine's TCP Accelerator allows network operators to dramatically improve network efficiency and quality of experience (QoE), without making changes to the network infrastructure.

2.2 Need of TCP Accelerator

TCP Accelerator addresses the most common TCP difficulties, such as bandwidth availability and TCP speeds being incompatible. It also overcomes inefficiencies caused by several

concurrent TCP connections by transparently bridging the access network with the transit network; it can manage all of the network's TCP connections as a group.

By adding Sandvine's TCP Accelerator to the network, operators can reap the following benefits:

- **Increased Network Performance:** Higher goodput-to-throughput ratios, improved resource usage, fewer retransmissions, and longer lifetimes for legacy infrastructure.
- **Improved QoE:** Faster data transmissions and increased application performance lead to measured and consistent QoE, which tends to increase customer satisfaction.
- **Rapid Return on Investment:** More efficient resource utilization defers capital and lets access network resources run 'hotter' without compromising QoE.

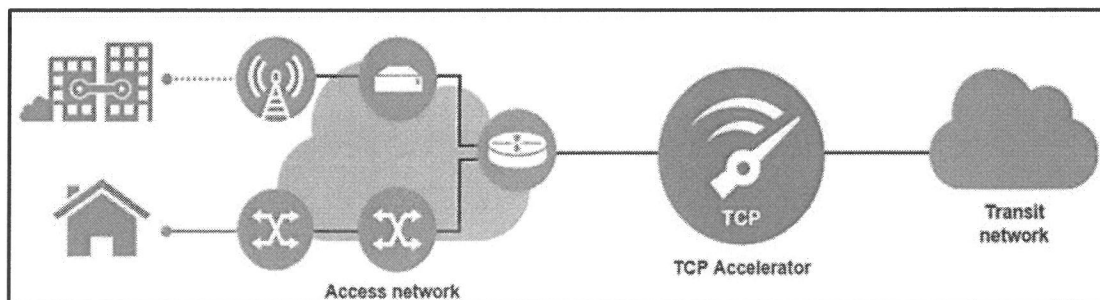


Figure 2.2.1: Basic flow of TCPA

The primary objective of TCPA is to boost TCP packet transmission to increase throughput and efficiency by reducing the total RTT for a packet. Presently TCPA exists as a standalone product. In order to reduce points of failure in the network and have lesser entities to maintain and configure for the customer, it is aimed to integrate this functionality with ActiveLogic.

TCP Accelerator is engineered to make TCP run better:

- TCP Accelerator ensures that TCP does not create a bottleneck when more bandwidth is available by accelerating the slow-start phase and ensuring that there is always enough data ready to be supplied.
- TCP Accelerator minimises effective latency by preventing buffer-bloat in access network resources when there is insufficient bandwidth (e.g., due to congestion, patchy mobile coverage, or a low-speed service plan).
- TCP Accelerator is able to manage all of the network's TCP connections as a whole, ensuring optimal efficiency, by transparently bridging the access network with the transit network.

Active Logic Operates between the internet and subscribers. All the traffic passes through active logic, and it parses and inspects the packet while bridging them. It can be configured to take various actions based on rules on specific traffic.

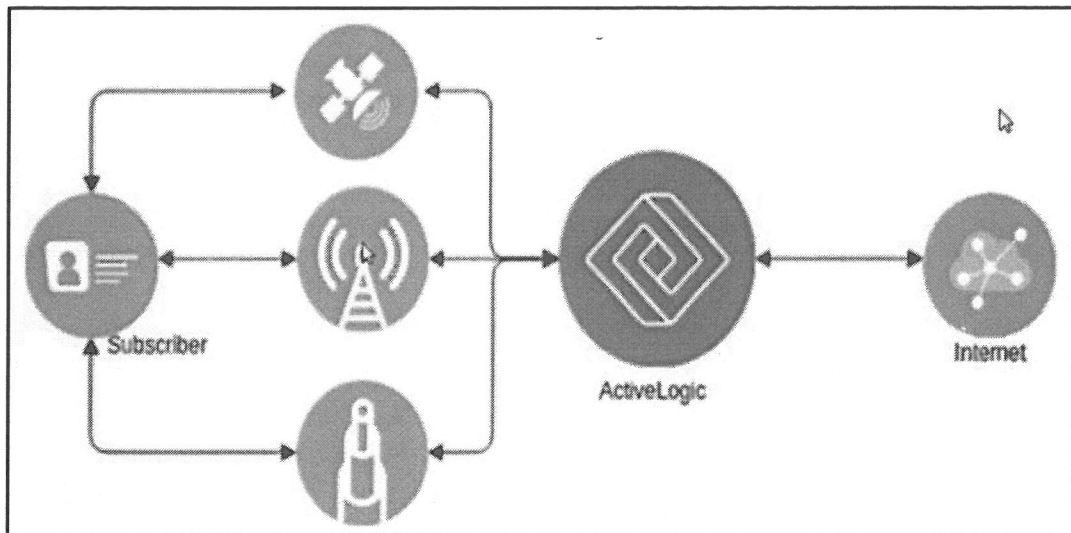


Figure 2.2.2: Basic flow of activelogic

2.3 Key Features of TCP Accelerator

2.3.1 Transparency

TCP Accelerator acts as a bridge rather than terminating the TCP connection, the acceleration is fully transparent to the endpoints, resulting in a variety of advantages:

- Connection migration:** From the perspective of either endpoint, connections can be transferred from optimised to non-optimized without impacting the connection.
- Connection resumption:** Long-lived but inactive connections time-out after a lengthy period of inactivity in order to recycle state memory; due to transparency, if a connection times out of TCP Accelerator but then resumes, the connection can be effortlessly resumed.
- Sequence number preservation:** Sequence numbers are preserved, ensuring that the client and server TCP connection states are identical after the TCP handshake.
- Step-out for quick upgrades:** TCP Accelerator can be turned off in minutes after the stores and forward buffers have been cleaned, without affecting traffic.
- TCP option transparency:** Endpoints can still negotiate the use of new TCP options or features if they are introduced (unlike terminating proxies, which would prevent its usage entirely).

2.3.2 Powerful TCP Acceleration Techniques

TCP accounts for 85 %to 90 % of fixed-access internet traffic and up to 96 % of mobile traffic, and TCP Accelerator accelerates virtually all of it, including:

- a. **Uploads and downloads:** Upload and download traffic are both accelerated.
- b. **Any application:** The default setup speeds up traffic regardless of the application, protocol, or service.
- c. **Encrypted traffic:** With more than 70% of traffic now encrypted, solutions must support encrypted traffic, which TCP Accelerator does.
- d. **HTTP2:** It is completely supported.

The only TCP that isn't accelerated is the traffic that shouldn't be accelerated. TCP acceleration is not implemented, for example, if the communication originates from a specialised device (e.g., a sensor or probe) that should not be accelerated.

In addition to the buffer management features described below, TCP is accelerated with a combination of techniques, including:

- **Reduced packet loss effect during slow-start:** Tunable system for reducing the impact of TCP packet loss early in the connection by ignoring such packet loss when modifying the congestion window.
- **Two-sided acceleration:** Most configuration variables can be set separately for the access side and the internet side.
- **Congestion control:** Distinguishes between true congestion events and radio faults (in mobile networks) and responds appropriately.
- **Fast retransmit:** Supports TCP Fast Retransmit, with configurable behavior.
- **Improved retransmission handling:** For the access side of the connection, it completely substitutes the normal timeout-based TCP retransmit, which has a huge impact on mobile networks (where hand-overs and jitter can cause bogus retransmissions).

Plus, TCP Accelerator is in a position to manage all the network's TCP connections as a collective whole, to ensure maximum efficiency.

2.3.3 TCP Buffer Management

When high-bandwidth downloads fill the buffers (queues) in the access network, other traffic gets stuck behind and the user experience suffers.

To prevent this scenario—called bufferbloat—in the access network, round-trip times need to be carefully managed; at the same time, a minimum amount of data has to be sent so as not to starve the radio network.

TCP Accelerator manages buffer queues by adjusting the sending rate to correspond to the level of buffered data in the access network. When TCP Accelerator determines that too much data is being queued for a particular user, it sends less data for the user in order to give time for the queue to shrink. TCP Accelerator learns the base latency for each user, which accounts for the varying radio access technologies and roaming users.

Importantly, TCP Accelerator prevents latency-insensitive applications from being favored over latency-sensitive ones. This approach treats all traffic fairly and has an enormous positive impact on QoE because it ensures sensitive applications aren't starved.

2.3.4 QUIC Traffic Control

QUIC is a UDP-based, Google provided protocol that has been designed to deliver data at lower latency rates than those of TCP. The protocol has gained some market traction with YouTube and other popular applications using it for transport.

QUIC does not work in a coordinated fashion with TCP, so it can have a serious and negative impact on TCP traffic performance. Left unchecked, QUIC has a tendency to overwhelm the same network buffers used by TCP, degrading the performance of TCP-based services that are on their way to users.

Before choosing a TCP optimization solution, make sure that the solution addresses both TCP and QUIC traffic management to ensure that applications running over both protocols can continue to perform as intended.

2.3.5 Egress Burst Control

When a large amount of data is released at once from a server close to the core network, this traffic can overwhelm any switch where there is a speed differential (e.g., going from 10G to 1G); the buffers in the switch may be insufficient to accommodate micro-bursts resulting from the speed differential, and as a result they might be forced to simply drop packets. This behavior has a negative impact on the experience of all users on the impacted switch.

To avoid this issue, Sandvine's TCP Accelerator limits the amount of traffic provided to a user in a configurable fraction of a second, preventing buffer overflow (typically a millisecond). This limitation allows the switch to manage the burst without running out of buffers, lowering retransmission rates significantly.

This functionality can relieve a network operator from replacing a huge number of switches.

2.3.6 Universal Access Support

TCP Accelerator works in any access network and with any combination of access technologies (e.g., Cable, DSL, 3G, 4G, WiFi, Satellite, WiMAX, etc.). TCP Accelerator facilitates implementation in shared RAN environments with a Multi-Operator Core Network(MOCN) in mobile networks .

2.3.7 Audit Records and Historic Reporting

TCP performance measurements and statistics are logged and can be used for audit purposes or examined for business and operational intelligence.

2.4 Benefits of TCP Accelerator

There are several key benefits to network operators

2.4.1 Taking Control of Layer 4

It allows a network operator to isolate and optimise their own portion of a TCP connection for their specific network conditions. This indicates that network operators, at least for their own network, can assume control over layer 4.

2.4.2 Accelerate TCP Slow Start to reach maximum speed faster

TCP Slow Start almost ensures that a Radio Access Network is underutilised at first. The end-to-end performance is increased by optimising each network separately, including access to the internet. The Access network's maximum speed is attained faster and maintained for a longer period of time.

2.4.3 Managing Packetloss in the Access network

Packetloss equals congestion in TCP Congestion Control, although this isn't necessarily the case in the access network. Packet loss can be caused by a variety of factors, including:

- Genuine coverage / interference problems / wi-fi outages
- Buffer bloat caused by microbursts
- Events involving cell handover

These aren't TCP congestion events, hence TCP isn't need to slow down. The TCP flow can be maintained by buffering these events in the TCP Accelerator.

2.4.4 TCP Goodput Improvements

The top-line bandwidth speed that can be achieved cannot be increased by using a TCP Accelerator. Rather, it will increase efficiency by better utilising network bandwidth to transfer more data. TCP Goodput - the amount of "excellent throughput" that may be obtained – is a term used to describe this gain in bandwidth efficiency. Goodput improvements for lossy networks can be extremely substantial.

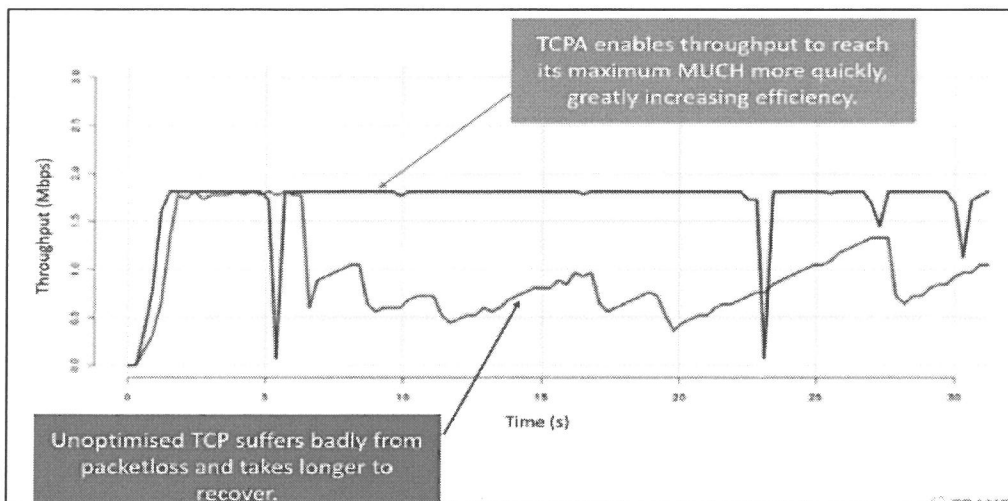


Figure 2.4.4: Graph showing Goodput Improvements due to TPCA

2.4.5 Managing Congestion in the Access Network

Congestion can sometimes occur in an access network. This usually indicates that access equipment buffers are full and packets are queued. Packet loss occurs when the buffers are unable to keep enough packets and must drop them.

An increase in Round Trip Time is one of the outcomes of the above. This is a common early sign of traffic congestion. Latency can be managed, packet loss can be prevented, and a more consistent user experience can be provided by installing a TCP Accelerator to prevent buffers from filling up.

2.5 Performance comparison of TCPA and TCP protocol

- **More Efficient Data Transfers**

The accelerated traffic (blue) gets up to speed faster, is faster overall, is more consistent, and recovers faster after a genuine packet loss, as shown in this graph of a single TCP flow.

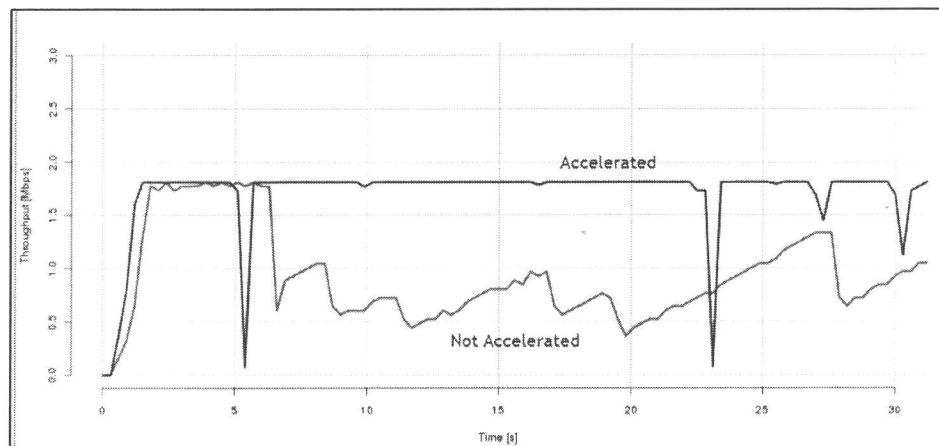


Figure 2.5.1: Graph showing efficient data transfers due to TCPA

- **Faster and More Consistent Round-Trip Times**

The difference in flow round-trip timings when TCP Acceleration (including buffer management) is enabled and when it is disabled is depicted in this graph. The average round-trip time in this mobile network was decreased in half (164ms average versus 320ms average), and round-trip timings were significantly more consistent.

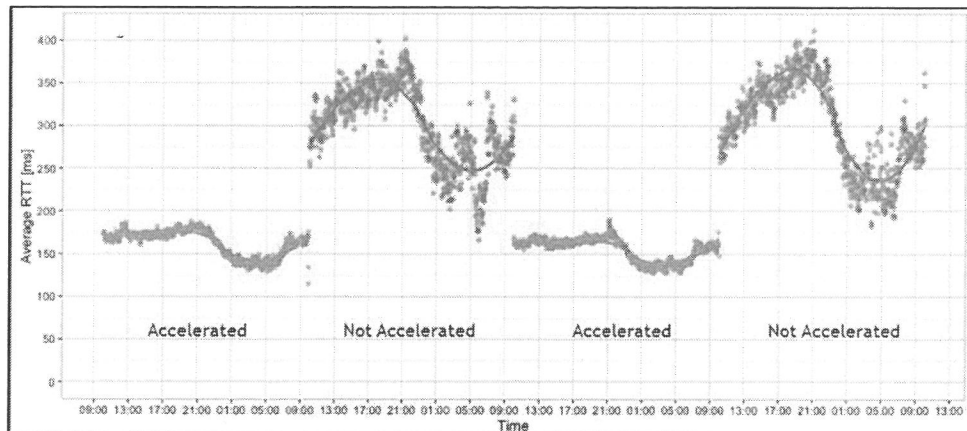


Figure 2.5.2: Graph showing faster and consistent RTTs due to TCPA

- **Fewer Retransmissions**

When TCP acceleration is enabled, the retransmission rate is substantially lower and much more constant, as shown in this graph from a CDMA network.

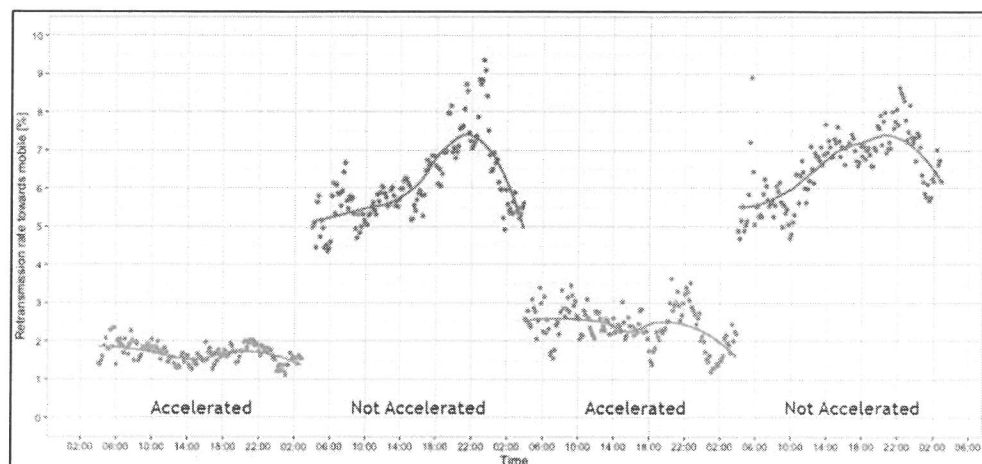


Figure 2.5.3: Graph showing decrease in retransmission rates due to TCPA

- **More Efficient Networks**

The graph, which is also from a CDMA network, indicates the overall positive influence of TCP acceleration on network efficiency. In this case, network efficiency (goodput as a percentage of total throughput) increased by 5% and consistency improved significantly.

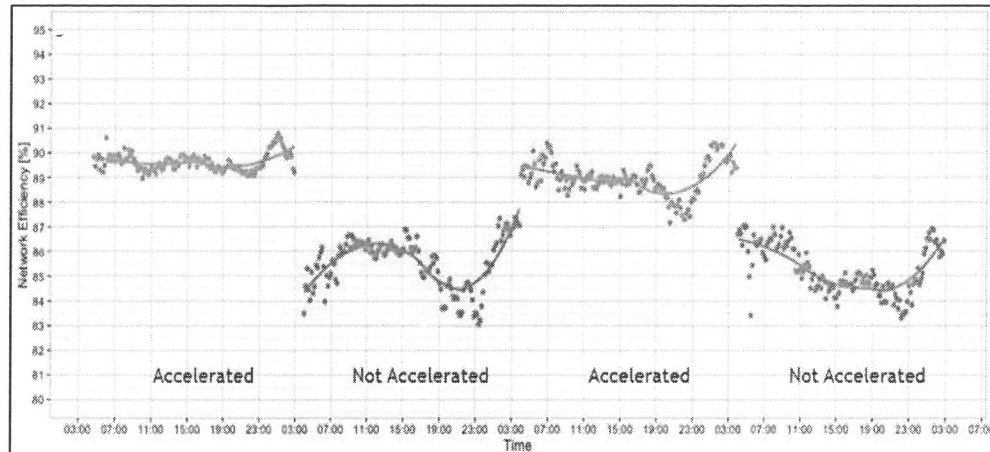


Figure 2.5.4: Graph showing increase in network efficiency due to TCPA

2.6 Summary

The impact of TCP Acceleration on a network depends mainly on the two network KPIs:

- Packetloss
- Latency

TCP Acceleration could have a huge influence on the end user experience and greatly increase overall network efficiency if the network is having packetloss or bad latency whether in the last mile or wireless access.

CHAPTER 3: TRAININGS AND FEATURE WORK

With the start of our Internship, we had a two months training and then we were assigned various tasks as a part of that feature.

3.1 Training Workflow

- **Phase-I** : General training related to C, C++, networking and operating systems. This will be followed by the product training related to TCP accelerator.
- **Phase- II**: we will work on their product- TCP accelerator. Sandvine's TCP Accelerator is a broadband optimization network function designed to resolve network congestion challenges.

It is placed at the edge of the access network where it runs TCP/IP algorithms that are carefully tuned to react in a manner

that is optimal for lossy high delay product networks, while still using simple algorithms towards the internet. My work will be mainly focused on the feature developments for TCP Accelerator solutions.

3.1.1 General Training

C++ Basics Course - C++ is an middle level programming language developed by Bjarne stroustrup starting in 1979 at bell labs. In this course, I have studied about the basic concepts of C++ language including the program structure, important terminologies of C++ , precedence and associativity, conditions in C++, strings, loops ,arrays, functions, structure in C++, data type conversion in C++, introduction to object oriented programming, constructor, operator overloading, inheritance, pointers, polymorphism, friend function and friend class, static member, variable and function, file handling, exception handling, standard template library, templates in C++, Lambda in C++ and multithreading in C++.

Operating Systems Course -In this course we had four specialization. The first one was virtualization where we studied about how virtual CPU works, how process in execution takes place, set mechanism, process API, direct execution of the process, CPU scheduling algorithm, multilevel feedback, lottery schedule, address spaces, memory API, and address translation base and bounds.

The next specialization was memory management where we studied about segmentation, free space management, paging, translation lookaside buffer, advanced page tables, swapping mechanism and swapping policy.

The third specialization was concurrency where we studied about concurrency and threads ,thread API, locks, locked data structure, condition variables, CPU scheduling, concurrency bugs and event based concurrency.

The last specialization was persistence where we studied about I/O devices ,hard disk drives ,redundant disk arrays, files and directories, file system implementation, FSCK and Journaling, log structured file system and flash based SSDs.

C advanced Course -In this course we have studied about storage classes, advanced data types, type qualifiers, bit manipulation, advanced control flow, input and output functions, advanced function concepts, unions, preprocessor, macros, advanced debugging analysis, static libraries and shared objects, useful C libraries, interprocess communication and signals, threads and networking (sockets).

Networking Course -In this course we studied about computer networks and their types, important terminologies, identifying and working of networking devices like hub, switches, types of data transmission, router, modem, network interface card, topologies, bridges, repeater, OSI model, TCP/IP model, transmission types, types of Internet connections, types of protocols and wireless technologies.

3.1.2 Product Trainings

PRE Overview: In this training, we were given the overview of the PRE product and its purpose. Basically, it was a session on analysis of that product- what actions/decisions will be taken while processing the network traffic.

Build and Unit Testing: In this session, we were told about how to make the build of the code changes you have done to do testing of your code. They gave a demo to set up docker environment on our laptop and how to perform required actions related to the given tasks using it.

Load-balancer: In this training session, we were told about that architecture of the PRE product and how load is balanced between the input traffic and the output traffic using that product.

Profile Push, testing with PRE, PLClient: In this training session, we were taught about how to do the profile push of the changes you have made in the code or the implementation part and then how to test it on the reserved PRE. They've also given the demo about how to use the PL client for the testing purpose.

Git: In this training session, we were taught about how to different useful git commands to perform the given tasks.

Flow-sync: In this training session, we were taught about how the traffic flows between the user side to the server side and how it is handled by the products produced by sandvine. And its various use cases over that network traffic.

Session Context: In this training session, we were taught about how various components of the ALclient communicate with the MPE (which is a control plane product) in order to do various tasks.

Shaping : In this training session, we were taught about how shaping is done and why it is required, they discussed about the components of shaping function

Layer 3 Feature: In this training session, we were taught about architecture and functionalities of Layer-3.

Coding Guidelines and Code review: In this training session, we were taught about various rules one should keep in mind while writing a piece of code or making any changes in the existing code.

ConfD-yang: In this training session, we were taught about ConfD and Yang models implementation on Dist.

VPL: In this training session, we had given a brief introduction about TehVPL and its infrastructure and how one can write VPL test cases.

Ruleset and PLRCD: In this training session, we were taught about the architecture of PLRCD which is responsible for compiling the ruleset before sending the same across to Engine. Ruleset is what makes sure that the correct policies are applied to traffic.

PLD and PLSD: In this training session, we were taught about the basic flow of PLD and PLSD and how they work.

3.2 Tools and Technologies

- **Docker:** Docker is a build and deployment tool that enables developers to create, compile and deploy their functions. It's a set of tools that allow them to automate the process of building and deploying containers. These containers are basically the package of your code, dependencies, config, processes and network protocol information which can be deployed in any machine.
One might look at docker as an advanced chroot, or more like a hybrid between chroot and a virtual machine. It does not run as an emulator, and it does not work as lightweight as

chroot. It's able to provide it's own network interfaces, and it act as it's own machine, it contains:

1. Runtime system
2. System libraries
3. System tools

It does more or less encapsulate a whole Operating System, except the kernel. Instead we divide the kernel into different spaces, encapsulated from each other.

Pros

1. enhanced chroot
2. internal network between services
3. security increased due to service encapsulation
4. Makes it easier to mock individual services

Cons

1. Security problems, services running as root user in docker has the same privilege as root on host.

Certain services from one machine to another. Then you just want to be able to setup the container and apply your tool and configuration on another machine. It's also easy to apply only difference between docker setups, since docker make use of intermediate container commits, so if you want to make run mysql and postgres services on ubuntu, then they share common disk space for common data to reduce disk usage.

Instead changes are written as delta in a overlay system. This makes it quick and rapid to start new system and with low footprints. Every image are based on a "commit" of an intermediate container. So, an easy way to create an image is to run.

- **VSCode:** Visual Studio Code is a type of integrated development environment (IDE) that allows developers to write and test code simultaneously. It's also compatible with other platforms such as console and docker. With support for over 200 languages, it's easy to get started with code creation. With the SSH extension, we can easily open a remote folder on any server or virtual machine. It lets us interact with files and folders located on the remote filesystem of our company. We can easily access code present at our lab build and test servers.
- **Mobaxtream:** It's the ultimate remote computing toolkit. It combines all of the essential remote network utilities, such as SSH, into a single Windows application that can run several SSH servers at the same time. You will be able to run the following Unix commands from this terminal window: ls, cd, grep, awk, tail, cut, sed, wget, rsync, telnet, ssh, rlogin, rsh. All of the necessary Unix tools are available. You can download free plugins if you need an application that isn't included in the core MobaXterm package.

- **Git:** It is a DevOps tool used for source code management. It is a free and open-source version control system used to handle small to very large projects efficiently. Git is used to tracking changes in the source code, enabling multiple developers to work together on non-linear development. We can use git commands to access others code changes, commit our changes and put it remotely so that they can be accessed by other employees as well.
- **C language:** The C programming language is a procedural programming language. ActiveLogic product code is mainly written in C language. Low-level memory access, a small collection of keywords, and a clean style are all qualities that make C language excellent for system programming.

3.3 Assigned tasks

- 1) **Timestamp validity check function:** This function will extract the timestamp and will check whether the timestamp is valid or not.

Methodology adopted for this task

- Read all the documents available on TCP Accelerator like SDD, SRS and the architecture page.
 - Attended all TCPA trainings.
 - Watched the available videos on the Code walkthrough of the TCPA code and skimmed various important code files of TCPA.
 - Understood the functionality of the existing function and compared it with the activelogic code to get an idea how to make that function in activelogic.
 - Had a look at the similar existing function present on activelogic to understand the working how packets are parsed in activelogic.
 - Made the function to extract the timestamp and then validate that timestamp and compiled it using Docker.
 - This task is merged to main feature branch.
- 2) **TCPA Zone creation task:** For every feature, a zone is created where all the counters are stored.

Methodology adopted for this task

- Understood the need of zone in our feature.
- Gone through the existing code that how they are created and registered.
- Tried to create a dummy counter to use it somewhere in code to test whether the zone is visible on ALClient software or not.
- This task is merged to main feature branch.

- 3) **Unit Test(UT) for Timestamp validity check function:** A UT tests various conditions present in the function to check whether it's working properly or not.

Methodology adopted for this task

- First of all, I have gone through the training videos on how to make a UT.
- Then we looked for similar UTs to get an idea how to make one related to our function.
- This task is merged to main feature branch.

- 4) **Window Validity Task and it's UT:** This function will check whether the window size is within the defined range or not. It will validate the window.

Methodology adopted for this task

- First of all, I have gone through the conditions required for a valid window in TCP.
- Then had a code walkthrough of the existing code of window validation in the TCP Accelerator and understood the requirements of that function.
- After getting the requirements, I cross checked it with the activelogic code and made those functions which were not present there.
- Then implemented the window validation function.
- Wrote down it's Unit test(UT) for each condition possible and validate the results with the required output.
- This task is merged to main feature branch.

- 5) **Impossible ACK Task and it's UT:** This function will check whether the ACK is valid or not as per the defined conditions.

Methodology adopted for this task

- First of all, I have gone through the conditions required for a valid ACK in TCP.
- Then had a code walkthrough of the existing code of Impossible ACK function the TCP Accelerator and understood the requirements of that function.
- After getting the requirements, I cross checked it with the activelogic code and made those functions which were not present there.
- Then implemented the ACK validation function.
- Wrote down it's Unit test(UT) for each condition possible and validate the results with the required output.
- This task is merged to main feature branch.

- 6) **Free segment Task:** This function will free the memory assigned to the segment.

Methodology adopted for this task

- Have gone through the function making the segment.
- Then had gone through the existing functions that frees memory and how they can be used in my function.
- Then implemented the free segment function.
- Wrote down it's Unit test(UT) for each condition possible and validate the results with the required output.
- This task is merged to main feature branch.

- 7) **Impossible ACK and window validity function integration with tcpa handle function Task and it's UT:** This task was the use case of the impossible ack and window validity function. Tcpa_handle() function manages various of the functionalities and one among those is checking validity of ack and window.

Methodology adopted for this task

- First of all, I have gone through the functionalities of integrating a function with tcpa_handle().
- Then had a code walkthrough of the existing code of that in the TCP Accelerator.
- After getting the requirements, understood what tcpa_handle() is doing till now and when should we check the validity of the window and ACK.
- Then implemented this part in tcpa_handle().
- Wrote down it's Unit test(UT) for each condition possible and validate the results with the required output.
- This task is merged to main feature branch.

- 8) **Buffer Segment and it's UT:** This function will insert segment into BTree.

Methodology adopted for this task

- First of all, had a code walkthrough of the existing code of that in the TCP Accelerator.
- After getting the requirements, worked on who to implement that on activelogic.
- Wrote down it's Unit test(UT) for each condition possible and validate the results with the required output.
- This task is merged to main feature branch.

- 9) **Fault in TCP Accelerator Code:** In this task, we need to fix the fault in the code.

Methodology adopted for this task

- First of all, I understood the logic behind the existing code and the root cause of the fault.
- After getting the cause, I restored that functionality without any fault in the code.
- This task is merged to main feature branch.

3.4 Gantt chart



CHAPTER 4: CONCLUSION

At Sandvine Technologies Pvt. Ltd., the Practice School- II opportunity provided a lot of knowledge in networking technologies. I can affirm that the professional experience and knowledge I acquired here would really assist my overall career. I have accomplished my internship learning lots of new tools and technologies.

Working in a professional setting, such as SandvineTechnologies Pvt. Ltd., has also taught me valuable skills such as teamwork and basic professional work ethics. The mentors here are extremely helpful and supportive.

Moving on to the project work. My contributions to the project work at the company were highly appreciated by my mentors and my seniors. Working on a real-life networking project had improved my knowledge and skillsets. During this internship period, I have worked on the first phase of the feature named “TCP Accelerator on ActiveLogic”, which was aimed at Accelerating TCP flows. As part of the project, I have worked on the scrum activities mainly focusing on updating the timestamp of the TCP flow on the Rx side and handling buffer segments and writing Unit tests.

Lastly, I am very excited to receive full time employment offer from Sandvine Technologies for the role of **Software Engineer**.

REFERENCES

1. **C++ BASICS COURSE** :<https://www.udemy.com/course/master-cpp-byteboard/>
2. **OS COURSE**:<https://www.coursera.org/specializations/codio-introduction-operating-systems>
3. **NETWORKING COURSE**: <https://www.youtube.com/watch?v=QKfk7YFILws>
4. **C ADVANCED**:<https://www.udemy.com/course/advanced-c-programming-course/>

Date: 06 May 2022

Sandvine Technologies (I) Pvt. Ltd.
RMZ Ecoworld SEZ,
Building 1A, Ground Floor,
Devarabeesanahalli, ORR
Bengaluru 560103
India
t: +91 (80) 6774 3333
f: +91 (80) 6774 3232
e: info@sandvine.com
www.sandvine.com

INTERNSHIP CERTIFICATE

TO WHOMSOEVER IT MAY CONCERN

This is to certify that Shreya Singh has done her internship as Trainee at Sandvine Technologies (I) Pvt Ltd. , Bangalore, from 3rd Jan, 2022 to 15th May, 2022 for the duration of 4.5 months .

She has worked on a project titled TCP-A on Active logic. This project was aimed at Accelerating TCP flows. As part of the project, she has worked on the scrum activities mainly focusing on updating the timestamp of the TCP flow on the Rx side and handling buffer segments and writing Unit tests.

During her internship she has demonstrated her skills with self-motivation to learn new skills. Her performance exceeded our expectations and she was able to complete the project on time.

We wish her all the best for her upcoming career.



 *Manoj Kavithala Mallappa*
Manoj Kavithala Mallappa
Manager, R&D
Sandvine Technologies (I) Pvt Ltd