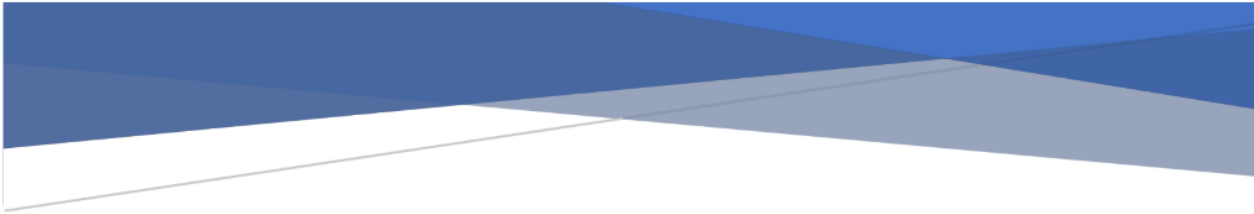


1701153Report

by Dylan Evans

FILE	914_DYLAN_EVANS_1701153REPORT_227510_1824126708.DOCX (249.82K)		
TIME SUBMITTED	16-JAN-2020 06:14PM (UTC+0000)	WORD COUNT	2715
SUBMISSION ID	118512037	CHARACTER COUNT	14362



NETWORK THEORY

BSc Software Engineering

Dylan Evans (1701153)
1701153@student.uwtsd.ac.uk

Table of Contents

Table of Figures	2
Table of Tables	2
1.0 Introduction	3
2.0 How does the Transmission Control Protocol (TCP) operate and endeavour to guarantee delivery of data?	3
2.1 What is TCP?	3
2.2 Two general's problem and the three-way handshake	4
3.0 What is the impact of the Maximum Transmission Unit (MTU) size on the transfer of data?.....	5
3.1 What is Maximum Transmission Unit (MTU)?	5
3.2 Size of protocol data unit (PDU)	5
3.3 OSI model (relative positions)	6
3.4 IP fragmentation	6
4.0 Synchronicity and asynchronicity in network protocols	7
4.1 What is synchronicity and asynchronicity?	7
4.2 I/O dispatch models	8
5.0 Interoperability for networked applications.....	9
5.1 What is Interoperability for networked applications?.....	9
5.2 How is it best achieved?	9
6.0 References	10

Table of Figures

Figure 1: TCP/IP Model	3
Figure 2: Three-Way Handshake.....	4
Figure 3: OSI Model.....	6
Figure 4: I/O Dispatch Models	8

Table of Tables

Table 1: Ethernet Standards and MTUs	5
--	---

1.0 Introduction

This report will cover the author's research on a variety of topics within the area of networks. The topics will include TCP, MTU, Synchronicity and asynchronicity, and interoperability. The relevance of design decisions within programming will also be covered within these topics as part of the author's research.

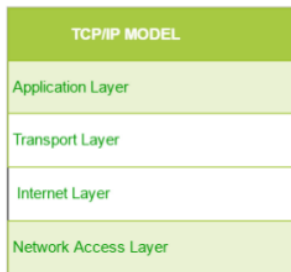
2.0 How does the Transmission Control Protocol (TCP) operate and endeavour to guarantee delivery of data?

2.1 What is TCP?

TCP (Transmission Control Protocol) is a standard used to establish connections with the Internet Protocol (IP). TCP/IP are the fundamental rules which define the Internet [1]. The idea of TCP was first brought up by Vinton G. Cerf and Robert E. Kahn in 1974 in the article entitled A Protocol for Packet Network Intercommunication [2]. The main reason that Vinton and Robert came up with the idea of TCP was to "Develop such networks to facilitate the sharing of computer resources" [2]. TCP solved the problems of communication between networks as there was no common protocol available for computers on separate networks to communicate.

TCP allows two hosts to establish a connection and allow the transfer of data. The TCP protocol guarantees delivery of data [3]. TCP does its best to make sure the data is delivered correctly and in the correct order. It does this by using sequence numbers to make sure the data is delivered with no missing data. The receiver then sends acknowledgments to the sender when the data is received. If

the sender does not receive any acknowledgments, then the server will resend the data until either the receiver sends the acknowledgments or if a timeout occurs due to something like a network outage.



The figure to the left shows the TCP/IP layer model. Each section has a task to perform for the TCP/IP model to work:

The application layer is where the programs on your computer directly interact with for example your internet browser or email client the application layer uses protocols like HTTP and SMTP.

Figure 1: TCP/IP Model [4]

The transport layer is where TCP lives. The application layer passes the information to the transport layer using port numbers, for example, HTTP uses port 80 or 8080 HTTPS uses port 443. Once this data is received by the transport layer this data is then put into packets, these packets then take the quickest possible route to the recipient. TCP supplies a header with every set of packets for the receiver to put the packets together again. This layer also has error checking to make sure that there is no missing data or that the data hasn't been packaged together incorrectly.

The internet layer is where the origin and destination IPs are stored.

The final layer is called the networking layer which defines how the data is sent over the network E.G. ethernet.

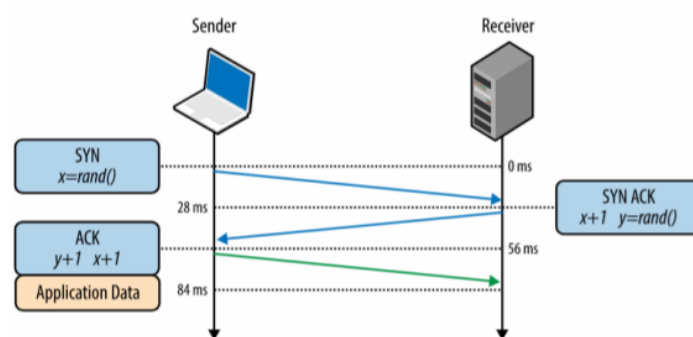
Error checking takes place within TCP in order to make sure that data has been delivered in order and intact. This is done via things like checksums to make sure that no segment of the delivery is corrupt. Acknowledgments are also used on the receiver side to let the sender know that the data is being received. If an error occurs in corruption or nothing was received, then retransmission will occur this is fast as duplicate segments that have not been received/corrupted will be resent instead of the whole transmission starting over.

2.2 Two general's problem and the three-way handshake

Given the problem where two armies had to attack a castle at the same time in order to defeat them, but these armies can only communicate via messengers how would both armies know that the message has been received on both ends. Army one could send a message to army two to attack at 8 pm but that message might not be received by army two then army one would attack and lose. If this was a computing problem how would it be solved? Army one could send the message to attack at 8 pm and then army two could send an acknowledgement that they have received the message. But what if army one didn't receive the acknowledgement then army two would attack and lose. There would be a never-ending stream of messages going back and forth from army one and army two.

In September 2018 there was a problem with Deliveroo orders [5]. The problem was very similar to the two generals problem, a customer sent an order through, payment was sent to the server and the server received the payment, but the server did not acknowledge the payment so customers sometimes received multiple orders as the app assumed the order failed. Some of the customers were even charged multiple times. How could this issue be prevented? The easiest way is to attach an idempotency token to each order. So if another request with the same token was received by the server then the order would be rejected and a copy of the acknowledgment would be resent.

The three-way handshake is used within TCP to establish a connection. It first establishes a connection with the server by sending a SYN (Synchronize Sequence Number) to ask the server if it's open for connections. If the server is open for connections it will then respond to the client with a SYN and an acknowledgement (ACK). After these two steps, the client will respond with an ACK and then the data transfer will begin.



the diagram to the left shows the three-way handshake.

Figure 2: Three-Way Handshake[6]

3.0 What is the impact of the Maximum Transmission Unit (MTU) size on the transfer of data?

3.1 What is Maximum Transmission Unit (MTU)?

Maximum Transmission Unit (MTU) is the size of the largest PDU(Protocol Data Unit) which is a unit of information passed along a network [7].

The MTU size for ethernet standard has been the same for some years sitting at 1500 bytes. While reducing processing overhead and improving performance can be achieved by increasing the MTU, using larger MTUs can cause delay and an increased probability of error [8].

Technology	Rate	Year	MTU	Serialization Delay
Ethernet	10	1982	1500	1200 μ s
Fast Ethernet	100	1995	1500	120 μ s
Gig Ethernet	1000	1998	1500	12 μ s
10-Gig Ethernet	10,000	2002	1500	1.2 μ s
100-Gig Ethernet	100,000	2010	1500	0.12 μ s

Table 1:Ethernet Standards and MTUs [9]

The above table shows the serialization delay for all the ethernet standards over the years. As seen the current delay is 0.12 microseconds for 100gig ethernet. Over time as ethernet technology gets better the serialization delay decreases. The serialization delay is the amount of time it takes for packets to be made ready for transmitted [9].

3.2 Size of protocol data unit (PDU)

A Protocol Data Unit is information that is sent as a unit to networks. This information contains things such as addresses control information or data. PDU is used within the OSI model especially in the first four layers the OSI model will be explained in the next segment of the report.

The terms used for the PDU within OSI model change within the first four layers:

Layer 1 the PDU is known as a bit (1s and 0s)

Layer 2 the PDU is known as a frame.

Layer 3 is known as a packet.

Layer 4 the PDU is known as a segment.

Layer 5 and above the PDU is known as data.

3.3 OSI model (relative positions)

OSI (Open Systems Interconnection) is a model used as a standard for applications to be able to communicate. As shown by the diagram below the OSI model is split into seven layers

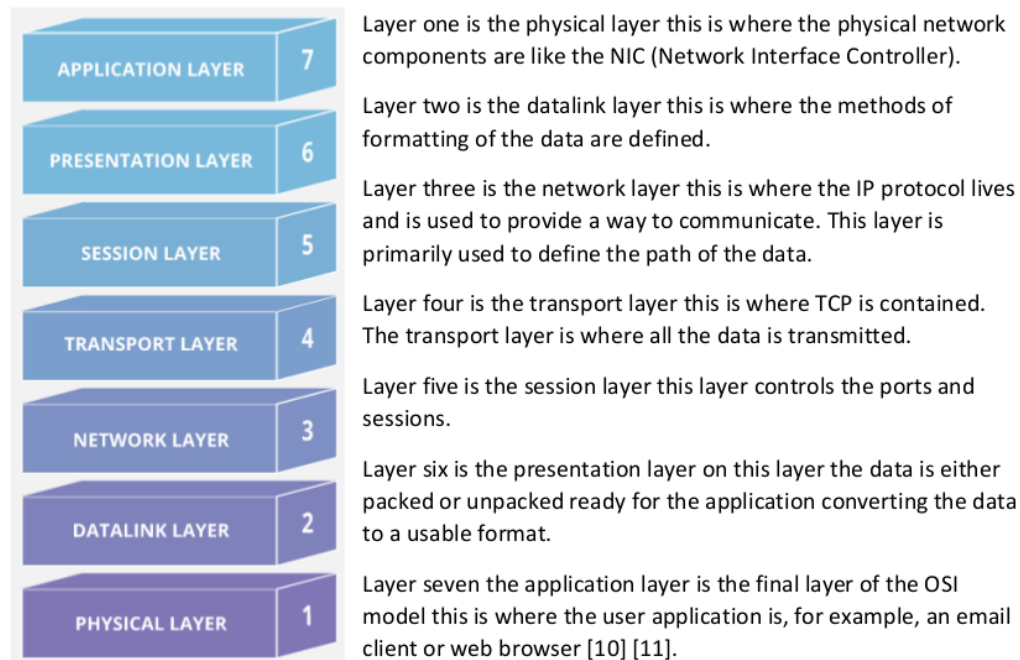


Figure 3: OSI Model[11]

3.4 IP fragmentation

Fragmentation is when an IP Datagram travels across a network in which the MTU is smaller than the size of the datagram. If the IP Datagram was larger than the MTU it would require fragmentation. If a larger Datagram isn't fragmented, then it would be discarded somewhere along the path. Once a datagram has been fragmented it will then be reassembled by the receiver.

Ip fragmentation can also be used to attack a network. The IP fragmentation attack can be used as a denial of service. The attacker uses the datagram fragmentation mechanisms to their advantage. There are two methods of attack

UDP and ICMP fragmentation attack is one type of attack this attack uses fake packets larger than the networks MTU which are unable to be reassembled this uses the resources of the server making it unavailable.

The second type of attack is called TCP Fragmentation attacks also known as teardrop attacks. This attack targets the reassembly mechanisms of TCP/IP preventing the receiver from reassembling fragmented packets. This type of attack will overwhelm the user's machine making it unavailable. This type of vulnerability was in early operating systems like windows 3.1 and 95 but did unfortunately reappear in windows 7 [12].

4.0 Synchronicity and asynchronicity in network protocols

4.1 What is synchronicity and asynchronicity?

Synchronicity is when the program waits for a task to complete before moving onto the next task. Asynchronicity allows the program to move onto the next task before the task completes. In terms of networks synchronous is when tasks happen in real-time. An example of where synchronous networks would be used is a live chat on things like Facebook messenger and snapchat. Asynchronous networks are used in cases where time isn't a supporting factor, time material can still take place within asynchronous networks. A good example of where asynchronous networks are used is places like forums, email, and blogs.

A good example of a synchronous application is uploading a video to the web to a page such as YouTube. This is synchronous because when you click upload you cannot do anything else within that application you have to wait for that code to finish executing.

Using the same example of an application that uploads a video to the web can be changed in order to make it asynchronous for example the programmer can add in the ability to change the title and description as the video is uploading.

In terms of networks, a synchronous network or full-duplex transmission is when data is sent and received at the same time. In order to achieve the bidirectional transmission, a signal clock is used. This clock needs to be synchronised on both the sender and receiver side for the transmissions to succeed this can be done via things like time servers.

Asynchronous is also known as half-duplex within networks. Asynchronous networks are used for intermittent transmissions for example walkie talkies. When the user wants to receive they need to put the device into receive mode the same the goes for sending a button needs to be pressed to put it into send mode [13].

4.2 I/O dispatch models

There are two types of dispatch models within input/output(I/O) these are synchronous and asynchronous. Within synchronous I/O the I/O begins on one thread and enters a wait state until the I/O request has completed. While asynchronous the I/O begins on one thread and the other threads can continue processing other jobs while the I/O completes.

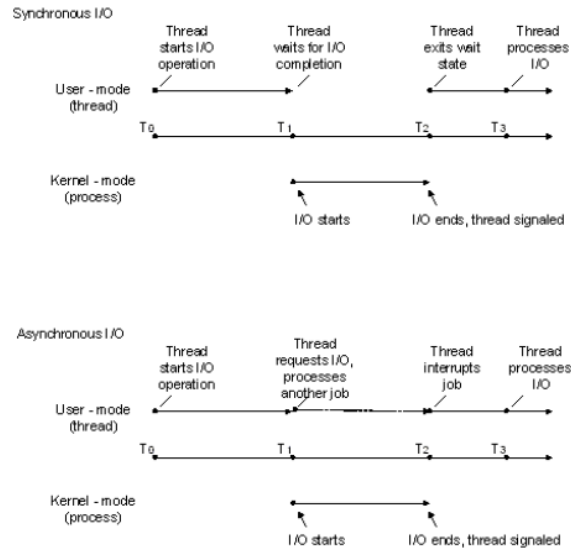


Figure 4: I/O Dispatch Models

a good diagram of Synchronous and Asynchronous is shown to the right. This diagram shows the differences between synchronous and asynchronous as explained above.

It is recommended when performing large I/O requests to perform these in asynchronous this makes them more efficient. On the other hand, if the I/O requests are small then making these asynchronous wouldn't be beneficial therefore making them synchronous would be the best option [14].

[14]

5.0 Interoperability for networked applications

5.1 What is Interoperability for networked applications?

Interoperability for networked applications requires a constant ability to receive and send data network to network. The networked applications need to be of top quality for the end-user and there should be no issues with the sending and receiving part of the application.

Network interoperability is important within things like telecommunications companies in order to achieve an end to end connectivity. A big advantage for the user within network interoperability is that they can communicate to anyone they want all from a single computer.

Interoperability requires that the base of every networked application needs to be able to communicate regardless of standards and protocols this can be achieved by either converting information from protocol to protocol or only using one protocol for all of the applications used which might not always be possible [15].

A great example of interoperability is telecoms no matter what telecoms provider a user chooses sky, BT or virgin media they can still all communicate with each other.

5.2 How is it best achieved?

Network interoperability is best achieved by first setting a common protocol to be used between the networks this way there will be no issues with translations between different protocols. It is best to make interoperability applications scalable so in the future, if more needs are required the application can be easily expanded upon [16].

6.0 References

- [1] "What is TCP (Transmission Control Protocol)?" [Online]. Available: <https://searchnetworking.techtarget.com/definition/TCP>. [Accessed: 13-Jan-2020].
- [2] V. G. Cerf and R. E. Kahn, "A Protocol for Packet Network Intercommunication," 1974.
- [3] "What is Transmission Control Protocol (TCP)? Webopedia Definition." [Online]. Available: <https://www.webopedia.com/TERM/T/TCP.html>. [Accessed: 13-Jan-2020].
- [4] "TCP/IP Model - GeeksforGeeks." [Online]. Available: <https://www.geeksforgeeks.org/tcp-ip-model/>. [Accessed: 13-Jan-2020].
- [5] "Deliveroo app crash leaves customers furious as some are charged SEVEN times." [Online]. Available: <https://www.thesun.co.uk/money/7284287/deliveroo-app-crash-customers-furious-charged-seven-times/>. [Accessed: 16-Jan-2020].
- [6] "Networking 101: Building Blocks of TCP - High Performance Browser Networking (O'Reilly)." [Online]. Available: <https://hpbn.co/building-blocks-of-tcp/>. [Accessed: 16-Jan-2020].
- [7] J. Postel, "Internet Protocol," p. 25, 1981.
- [8] P. Reviriego, A. Sánchez-Macián, J. A. Maestro, and C. J. Bleakley, "Increasing the MTU size for energy efficiency in ethernet," in *IET Conference Publications*, 2010, vol. 2010, no. 566 CP, pp. 124–128.
- [9] D. Murray, T. Koziniec, K. Lee, and M. Dixon, "Large MTUs and internet performance," in *2012 IEEE 13th International Conference on High Performance Switching and Routing, HPSR 2012*, 2012, pp. 82–87.
- [10] Neil Briscoe, "Understanding The OSI 7-Layer Model," 2000.
- [11] "What Is The OSI Model? | Cloudflare." [Online]. Available: <https://www.cloudflare.com/learning/ddos/glossary/open-systems-interconnection-model-osi/>. [Accessed: 15-Jan-2020].
- [12] "What is an IP Fragmentation Attack (Teardrop ICMP/UDP) | Imperva." [Online]. Available: <https://www.imperva.com/learn/application-security/ip-fragmentation-attack-teardrop/>. [Accessed: 16-Jan-2020].
- [13] "Synchronous & Asynchronous Networks in WAN | Study.com." [Online]. Available: <https://study.com/academy/lesson/synchronous-asynchronous-networks-in-wan.html>. [Accessed: 16-Jan-2020].
- [14] "Synchronous and Asynchronous I/O - Win32 apps | Microsoft Docs." [Online]. Available: <https://docs.microsoft.com/en-us/windows/win32/fileio/synchronous-and-asynchronous-i-o>. [Accessed: 16-Jan-2020].
- [15] "Know About The Network Interoperability in Telecom." [Online]. Available: <https://blog.comviva.com/all-you-need-to-know-about-network-interoperability/>. [Accessed: 16-Jan-2020].
- [16] K. Viswanath and K. Obraczka, "Interoperability of multicast routing protocols in wirelessad hoc networks," *Wirel. Commun. Mob. Comput.*, vol. 6, no. 2, pp. 225–234, Mar. 2006.

FINAL GRADE

55/100

GENERAL COMMENTS

Instructor

A mixed selection of source quality used. You really need to try to avoid weak web-based sources in future work - though they generally improve somewhat as you progress through the report.

The report overall is reasonable, though a little lacking in detail. You're a modest way shy of the word count - which isn't necessarily a problem, but has resulted in a proportional lack of depth here.

A reasonable attempt, but there are a few things here for you to work on during semester 2. See me if unclear.

Engagement with Literature Skills (20): 11

Knowledge and Understanding Skills (30): 14

Cognitive and Intellectual Skills (30): 15

Transferable Skills for Life and Professional Practice (20): 14

PAGE 1

PAGE 2

PAGE 3

PAGE 4

PAGE 5

PAGE 6

PAGE 7

PAGE 8

PAGE 9

PAGE 10

PAGE 11
