University of Texas Rio Grande Valley

ScholarWorks @ UTRGV

Theses and Dissertations

12-2020

Microsoft’s Surface Pro 4 Performance under Denial of Service Attack

Adrian Guerrero

The University of Texas Rio Grande Valley

Follow this and additional works at: https://scholarworks.utrgv.edu/etd

Part of the Electrical and Computer Engineering Commons

Recommended Citation

Guerrero, Adrian, "Microsoft’s Surface Pro 4 Performance under Denial of Service Attack" (2020). *Theses and Dissertations*. 456.

https://scholarworks.utrgv.edu/etd/456

This Thesis is brought to you for free and open access by ScholarWorks @ UTRGV. It has been accepted for inclusion in Theses and Dissertations by an authorized administrator of ScholarWorks @ UTRGV. For more information, please contact justin.white@utrgv.edu, william.flores01@utrgv.edu.

MICROSOFT’S SURFACE PRO 4 PERFORMANCE UNDER DENIAL OF SERVICE ATTACK

A Thesis

by

ADRIAN GUERRERO

Submitted to the Graduate College of
The University of Texas Rio Grande Valley
In partial fulfillment of the requirements for the degree of

Master of Science in Engineering

December 2020

Major Subject:Electrical Engineering

MICROSOFT'S SURFACE PRO 4 PERFORMANCE UNDER

DENIAL OF SERVICE ATTACK

A Thesis
by
ADRIAN GUERRERO

COMMITTEE MEMBERS

Dr. Sanjeeve Kumar
Chair of Committee

Dr. Kuang
Committee Member

Dr. Dong
Committee Member

December 2020

Copyright 2020 Adrian Guerrero
All rights reserved

ABSTRACT

Guerrero, Adrian, Microsoft's Surface Pro 4 Performance Under Denial of Service Attack, Master of Science in Engineering, December 2020, 105 pp., 3 tables, 75 figures, 3 schematics, references, 47 titles.

Microsoft Surface Pro 4 (SP4) is a new device in the line of hybrid computers today. It aims to work as both as table and laptop computer. There are multiple versions of SP4 that include devices equipped with i3, i5 and i7 core processors.

This makes SP4 a great addition for hospital, clinics or schools for its portability and features. As these locations can be prone to cyber-attacks, we plan to test this computer's reliability and performance under cybersecurity attacks. The device tested is built with Intel Core i7-6650U 2.21 GHz processor with automatic overclocking capabilities up to 3.4 GHz, 16 GB Random Access Memory (RAM), 256GB hard drive and Windows 10 Professional operating system 64-bit version, plus one USB 3.0. The USB 3.0 port was used as Ethernet port with Gigabit adaptor from Freegene.

In this paper, we measure and present various computing parameters to understand the effects of cybersecurity attacks on a Microsoft Surface Pro 4 computer.

DEDICATION

Este logro no hubiera sido posible sin la paciencia y el apoyo de mi familia. Mi madre Maria Escobedo de Guerrero, mi padre Marcos Guerrero, Hermanos y hermanas, sobrinos y sobrinas todos ellos y ellas tuvieron que lidiar con un montón de preocupación e incertidumbre por mí. Tomó todo su amor y comprensión para mantenerme motivado y ver este logro concretarse.

This achievement would not have been possible without the patience and support of my family. My mother Maria Escobedo de Guerrero, my father Marcos Guerrero, Brothers and sisters, nephews and nieces all of them had to deal with a lot of worry and uncertainty for me. It took all their love and understanding to keep me motivated and see this achievement materialize.

ACKNOWLEDGMENTS

I am thankful for the support that my alma matter has provided all these years. The work study, Direct Wage and Student Organizations I was part of have contributed to my growth. Most of all the guidance from my professors has helped me build that confidence in my abilities to learn and apply my new skills in this discipline.

Especial thanks to Dr. Kumar for his tireless belief in my abilities to contribute to this conversation. Thanks also to my thesis committee for their input and feedback.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF SCHEMATICS

Page

CHAPTER I

INTRODUCTION

Microsoft has been a leader in computer devices since late 1970s and late 1980's. It started out with Bill Gates' interest in computers and his vision to improve the early personal computers of the time, such as MITS Altair, using his skills of computer programming. After helping improve Altair with its programming, Gates and his partners created the foundation of what is now Microsoft, one of the most well recognize brand in computer devices throughout the world [8][9]. Computers have evolved from their origins and consumers now demand faster, more reliable, and portable devices for their modern productivity needs. To meet that demand came the need to improve the components of the computer. Sometimes the question of whether smaller devices will take away speed from the processing power comes up, but computers, as we have seen, have followed a trend of becoming more efficient as their size reduces and in turn, become more portable. This trend has shown that when computer devices reduce in size their processing speeds improve as well. This is due to multiple factors, including distance reduction. We have seen it in our schools when we were using those bulky Macintosh computers in the mid-90s that used a heavy tower to house the mother board and the hard disk. On top of that we also needed a screen that filled almost the whole desk. Now, with modern technology and computers that have become part of our every-day life, it is necessary to understand how modern computer devices can hurt this new way of life. Computer devices, whether small or big, they all have something in common. They will need to connect to a network at one point or another.

1.1 Motivation

Nowadays, the Internet is considered part of our lives and we hardly notice that it is there. Most of the time when we go to work or at school, the first thing we do when we turn on a computer device is search under the Network & Internet Settings of our computer to check for available Wi-Fi networks to connect us to the Internet before we begin working on our assignments for the day. Many electronics manufactures have emerged these days to tap into the portable devices market, including Microsoft. Microsoft has been keeping a leadership tradition in innovation around computer platforms. It has dominated the computer world since the 90s. To keep up with that tradition, Microsoft has developed the Surface computers, which are classified as both laptop and tablet due to its portability features and processing capabilities, a series of portable devices that started with Surface Pro in late 2015 and continues today with their most recent release of Surface Pro 7 and Surface Pro X. Although, these devices are small enough to be portable, they are considered in the category of computers that can carry your productivity software like powerpoint, excel, word and even some heavy graphics, and are able to process instructions as fast as any laptop that would be consider much faster due to its size.

1.2 Statement of the Problem

Following in the line of previous research done at UTRGV on computer performance on platforms using Windows operating system as their base by former students [9][10] [11], we tested a personal computer, Surface Pro 4, to find out more in-depth details about its performance. Since in previous works there were observations of memory being affected by cyberattacks, but the information was not very conclusive, reading more about computer memory we figured memory is more complex and to dismiss the possibility of it not affecting the computer performance would be a mistake. There are different memory units in a computer, Disk, Main memory, and cache memory. This memory units all work at different speeds at different levels, the fastest being cache, which sits next to the computer processing unit (CPU). We wanted to know how well Surface Pro 4 performed when popular Distributed Denial of Service attacks was executed. As we ran tests targeting memory, we decided to include other components to get a wider picture of all the components that are actually affected by these attacks. This is explained in more detail in chapter two section one.

1.3 Thesis Outline and Concluding Remarks

This work is structured in the following format. Chapter 1 gives a brief introduction to all the relevant components to this thesis, processor, cyberattacks, communication protocols, and hybrid computers. In Chapter 2 an explanation of the rationale behind our project is offered. We present the background of four popular cyberattacks that have kept network security personnel busy in the past few decades. In Chapter 3 and Chapter 4 we explain our focus for this paper. We present tests done under a controlled environment simulating some of the most damaging and common types of attacks. We dig in deep into the processor and find out what areas of the processor are affected during one of these attacks. In Chapter 5 we offer our concluding remarks using a comparison of the two most damaging attacks when these layers 3.5 and layer 4 attacks are involved.

CHAPTER II

MICROSOFT SURFACE PRO 4 PERFORMANCE

2.1 Hybrid Computers

Hybrid computers are a new generation of computer devices that aim to perform as good as a laptop, possessing the portability features of a tablet. Here, when we say hybrid computer, we refer to definition used in www.computerhope.com [1]. These devices have made an entrance into our society in the last decade and everything indicates they are here to stay. Microsoft has released a series of devices called Surface Pro, and Apple has done its part releasing its iPad Pro into the market [2]. With this knowledge in mind, we ventured into finding out the performance of a well-known hybrid-computer made by Microsoft, called Surface Pro 4 (SP4), released in 2015. This device is equipped with intel core i7 processor, the latest in processor technology for portability and performance. More precisely, we want to know how the computer processor in this device performs when well-known cyberattacks make their way through the network. The type of cyberattacks we observed are in a category called Denial of Service (DoS) or Distributed Denial of Service (DDoS) attacks. We pick this category of cyberattack because these are the ones that aim to drain the resources of a computer device, including the processing resources [3]. Computer processing resources, as defined by Britannica, include computer processor unit (CPU), memory, input and output devices and network [7]. Each of these resources is composed of different components that affects the performance of the whole computer in different ways and degrees.

2.2 Components of Interest

One of the main components of a computer is its processors. Within the processor there are millions of transistors that make up different components of a processor. A component that has become essential part of the CPU is the cache. Cache, over the last few decades, has been placed inside the CPU to help its processing activity serving as a temporary storage of the most used data by the processor, which allows the processor to save time in the trip to look for data in storage components like RAM and Disk. This design method helps the processor conserve resources and speed up processing activity because data that is often used in processing is kept near the processing unit. This allows the processor to shorten the time it would take to process its instructions. Memory, Logical Disk, Physical Disk, Process, Processor and System are the computer components we opted to analyze in search of a more in-depth understanding of the impact of cyber-attacks that commonly affect networks around the world.

The computer processor has come a long way since Intel’s early days in early 70’s with its first microprocessor, the intel 4004, which was made up of only twenty-three hundred transistors in ten micrometer architecture [4][6]. It was used in Busicom calculators to do arithmetic manipulation and was discontinued after ten years in 1981. Now, the intel core i7-6500U is made up of 1.7 billion transistors in the fourteen-nanometer architecture. The Intel i7 is used in the Microsoft Surface Pro 4, much more powerful and much lighter than the computers back then. A processor, as a base, follows five basic steps: Fetch, Decode, Execute, Read, and Write.

2.3 Cyber Security

Security of computer networks around the world has taken special attention since the first cyber-attack happened in early 1980s. Keeping a computer device from getting tampered with nowadays has become essential. It can almost be considered an art. Computers, and our electronic devices are constantly connected to a network that gives us access to the information that we need from the Internet. If we are at the store, nowadays we can get connected to their Wi-Fi, which in turn gives us access to the Internet. If we are on our school campus, whether it is High School or University, our devices are, most of the time connected to The Internet. This puts us and our networks in a vulnerable situation. To get access to information we need, be it for entertainment or to help us in our daily tasks, we enter this exchange of information with the Internet. And Whether we know it or not other people have access to our devices information, which is all they need to send our device or the network we are connected to a cyber-attack. Cybersecurity is the discipline that investigates and performs security measures related to the Internet. Cybersecurity is concern with three components of computer communication principles known as the CIA Triad: Confidentiality, Integrity and Availability [9]. Confidentiality, Integrity, Availability are the guiding principles that network communication follows to make sure information resources are flowing in the direction and in the timely matter that they are needed.

2.4 Concluding Remarks

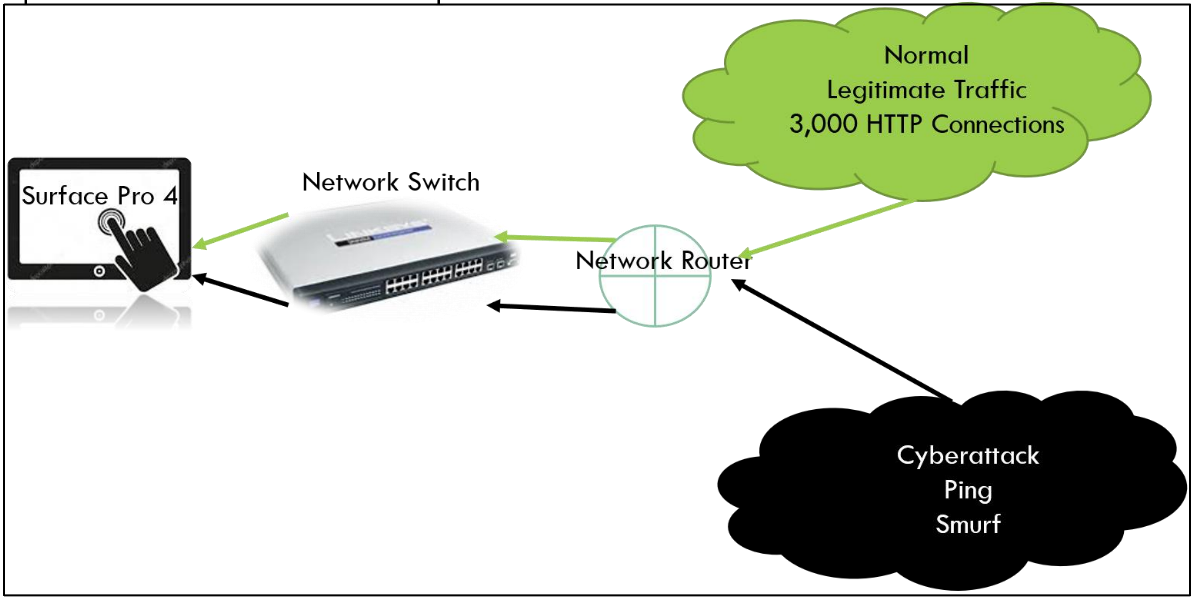
In this chapter we explained what a hybrid computer is and talked about what they look like. An example of such device being the Surface Pro (sp4), which is manufactured by Microsoft. We also mention some of the computer components that are affected by common cyberattacks. Then we give a brief explanation of what is cybersecurity, which is concerned with the safe communication of devices on the Internet.

CHAPTER III

SMURF VERSUS PING ATTACK RESULTS

3.1 Experimental Setup

In this chapter we cover the elements we used to perform our Layer 3.5 Cyberattack tests Ping and Smurf attacks. In the first section we include the hardware and software that was necessary to run our tests. Then, we explain our results in section two. Section two is divided into definition of the object, definition of the counter in quotation marks as given by Microsoft, and the graph representing the results obtained for that part of our test. The graph is followed by our description of the results. In section three, we conclude with a synthesis of the experiment we describe in this chapter.



Schematic 3 - 1 : Ping Attack and Smurf Attack Experimental Setup

In Schematic 3-1 we can see the setup of the experiment we ran to analyze the performance of SP4. Going from left to right, the two clouds represent the Internet. The green cloud represents legitimate user connections. These connections represent a simulation of 3,000 HTTP connections. The black cloud represents simulated cyberattack. This attack traffic is added to the legitimate traffic. The traffic from these two cloud networks comes into the network through the router, which in this case a simulated. The Network Switch just connects multiple devices in our network. The device on the far left represents the SP4 device we tested. This is how we test how well a computer or server performs under cyberattacks. Below is a list of the actual devices we used and some of their features.

3.2 Hardware and Software

The tools we used in this project start with Microsoft Surface Pro 4 (SP4) computer. SP4 is built with an Intel Core i7 processor, which runs at 2.2 GHz. It has a 256 GB Harddrive, and a micro-USB port. We used Ethernet-to-USB 3.0 adaptor from Freegene and a Linksys Business Series SRW2024 version 1.2 network switch to transfer traffic generated for our test onto SP4. Data was collected using Performance Monitor, a feature embedded in Windows operating systems, to analyze the performance of many computer parameters, including Cache, Logic Disk, Memory, Physical Disk, Process, Processor and System. The data comes in a comma separated value (csv) file, so we used Microsoft Office Excel to analyze our data to then convert it to graphs that can be easier to read and analyzed. We learned that CSV files are not very reliable when you need to make changes to it. Many times, the data was lost when we tried graphing the data directly from a CSV file. To not go through the same issue any more we converted the extracted CSV file to excel file. This worked out better because we were able to create our graphs and we preserved the integrity of the original data. We worked from the excel file and kept the CSV file untouched for the remaining of the project.

3.3 Performance Objects of Interest

The computer parameters we use in our evaluation are presented in this chapter. We used ten different traffic intensities from 10% to 100% of the 1 Giga bits per second bandwidth available in our test for comparisons. At 10% to 100% we introduced simulated Internet traffic to compare to traffic coming from a cyberattack. We grouped these intensities, comparing the base traffic with the cyberattacks to see the distinction and effects the intensities of the attacks have on the computer. Base 10% to 100% represented in gray bars, we let the computer run with simulated legitimate traffic to capture a representation of Normal traffic, to then compare it to that of cyberattacks. Ping attack traffic is represented by orange bars and Smurf attack traffic is represented by blue bars. Ping and Smurf attacks are considered part of Layer 3.5. Starting at 10%, the cyberattack traffic is added. Each percentage of the attack intensity is kept for five minutes.

3.3.1 Cache

“The Cache performance object consists of counters that monitor the file system cache, an area of physical memory that stores recently used data as long as possible to permit access to the data without having to read from the disk. Because applications typically use the cache, the cache is monitored as an indicator of application I/O operations. When memory is plentiful, the cache can grow, but when memory is scarce, the cache can become too small to be effective.” For cache we looked at three elements: Dirty Pages, Dirty page threshold and Pin Read Hits %.

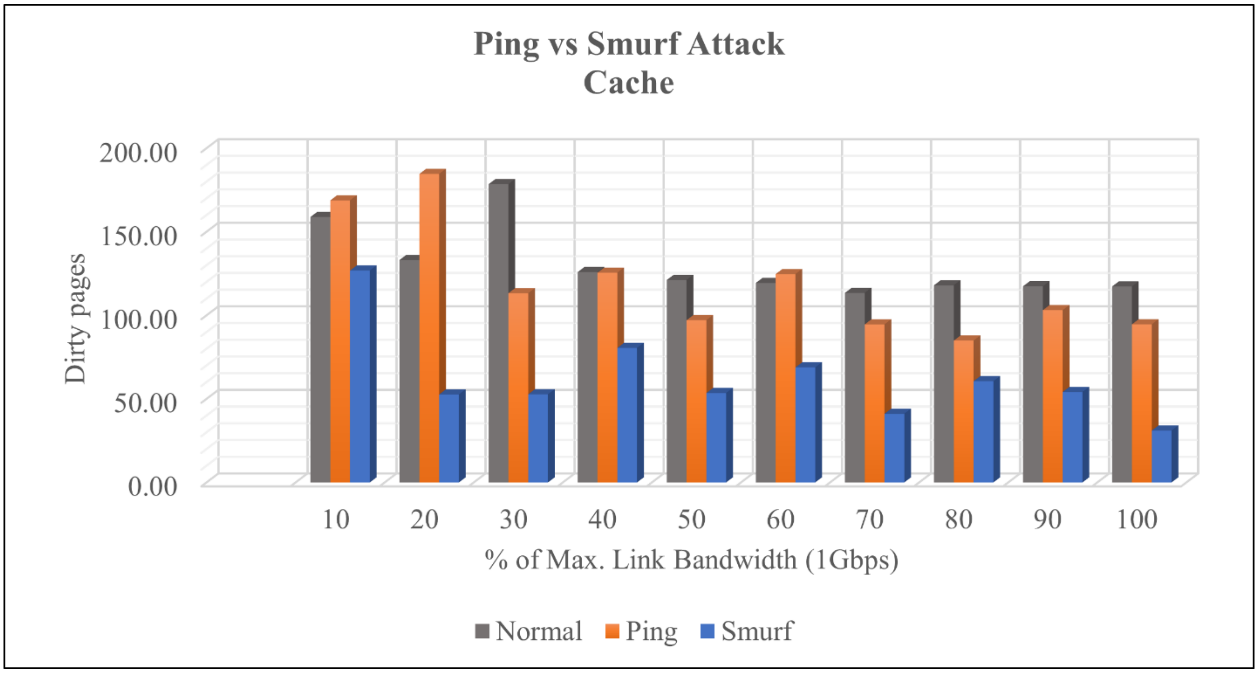


Figure 3 - 1 Cache: Dirty Pages

"Total number of dirty pages on the system cache"

A page is a block of memory in virtual memory. A dirty page is a block of data or memory that has been changed or modified while residing in cache. In Figure 3-1, we can see how memory is affected, indicated by the number of Dirty Pages that are generated. The shaded bars represent normal traffic, orange bar represents Ping attack, and the blue bar is Smurf attack. This graph indicates how during a Smurf attack there are less dirty pages generate than during a Ping attack. For both attacks we can observe, the greatest number of dirty pages generated happens at the beginning of the attacks. This makes sense, since we know that the cache if filled with data reused over and over in memory closed to the processor to prevent more delays. At 0% of Max. Link Bandwidth Normal traffic is equal to idle computer. This graph clearly shows that the greatest number of dirty pages is generated at the beginning, generating the data needed for process. Once the data is in cached it is reused over and over, which is why the remaining percentages do not show as high bars as the initial ones.

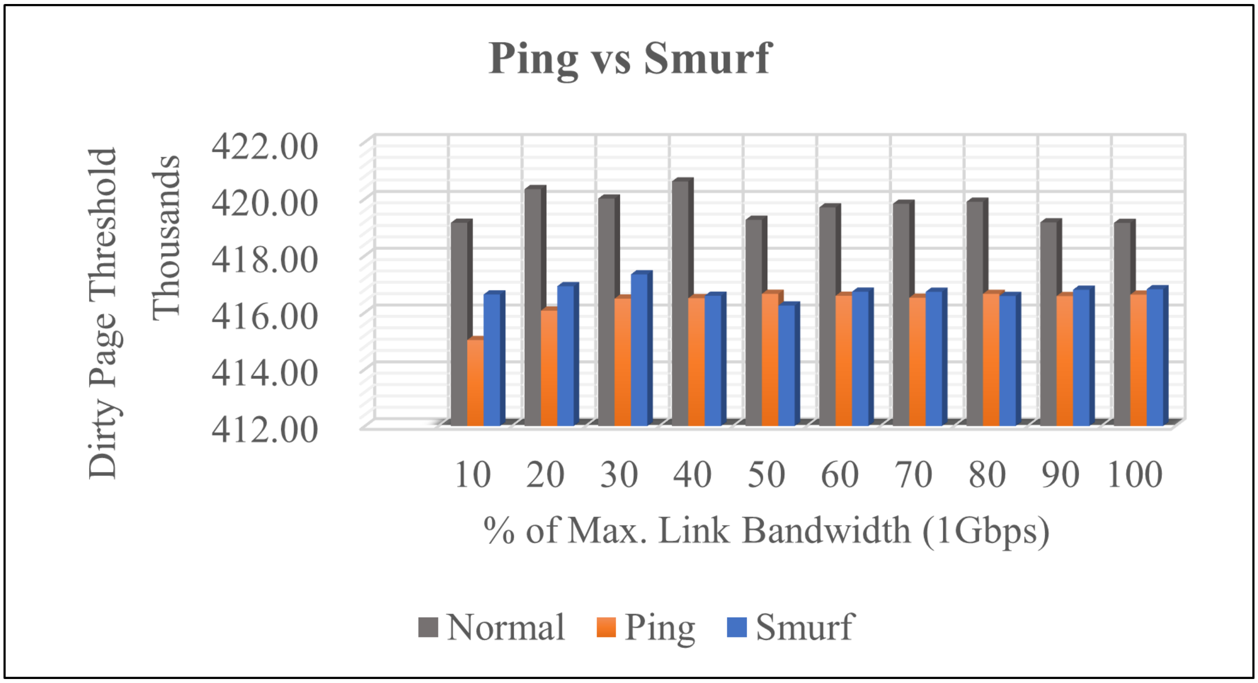


Figure 3 - 2 Cache: Dirty Page Threshold

“Threshold for number of dirty pages on system cache”

In Figure 3-2, we can see neither attack reached the threshold limit set for Dirty Pages. In this graph we can see not much difference in the fluctuations of the threshold as the intensity of the attacks is increased. Both attacks show a slight increase at the beginning of the attack, but both reached a peak at 30% of the attack intensity. Smurf attack, based on the observation of the graph, generated the greatest threshold increase. According to [10] and [11] this is due to the operating system reducing the threshold to get rid of the dirty pages in the cache quicker to keep the system from failing, which would affect the performance of the computer and integrity and availability of data or any output needed.

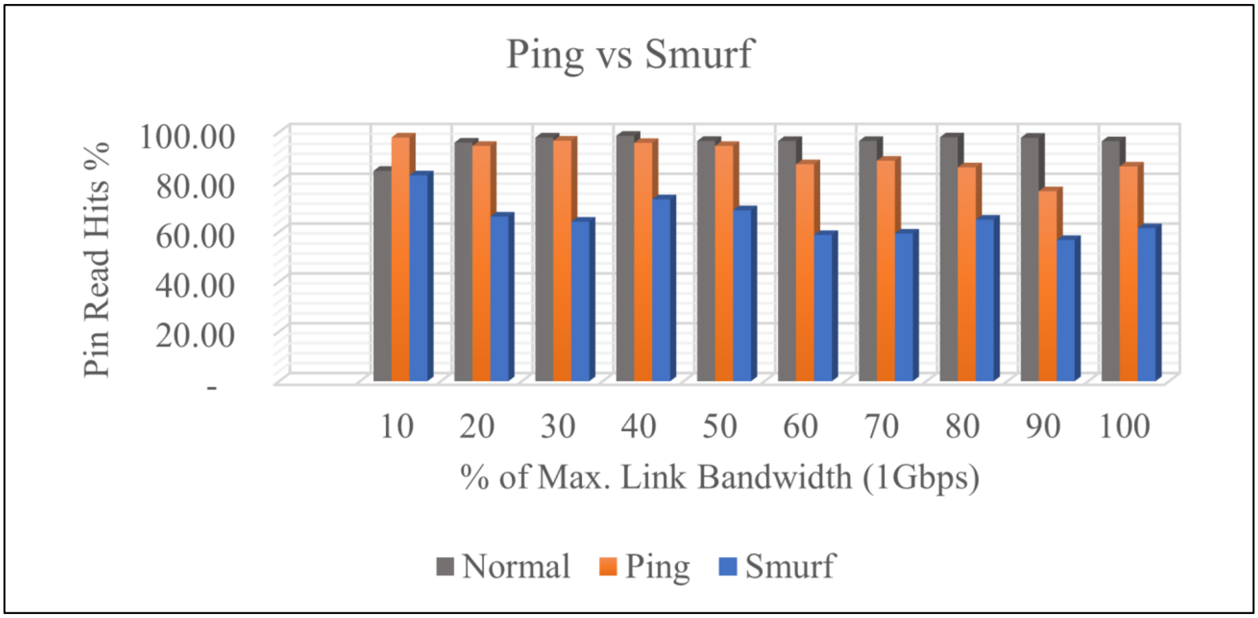


Figure 3 - 3 Cache: Pin Read Hits Percent

"Pin Read Hits is the percentage of pin read requests that hit the file system cache, i.e., did not require a disk read in order to provide access to the page in the file system cache. While pinned, a page's physical address in the file system cache will not be altered. The LAN Redirector uses this method for retrieving data from the cache, as does the LAN Server for small transfers. This is usually the method used by the disk file systems as well."

In figure 3-3, we can see that in a normally functioning computer, a computer without any traffic, the cache is almost immediately entrusted with holding the necessary data for the computer to run. If we focus on the gray bars, starting with zero bar, which is when the computer was running by itself, we can see the computer communicating with disk 45% of the time. Then, when traffic was introduced at 10% of the 1 Gbps, the cache started getting busy. According to the definition of Pin Read Hits Percent, during Smurf attack there are more reads required from disk than during a Ping attack. This means that the pin read hits are focused on the file system cache, where processor can access data quickly because it's already stored near it.

3.3.2 Logical Disk

At first, it was unclear how much disk time DDOS attacks were taking -- we thought it would be minimum, however, when we decided to isolate some counters related to disk we found some interesting results.

“The Logical Disk performance object consists of counters that monitor logical partitions of a hard or fixed disk drives. Performance Monitor identifies logical disks by their a drive letter, such as C.”

In Logical Disk performance object, we focused on % Idle Time, Disk Bytes/sec, Disk Write Bytes/sec and Disk Writes/sec. Each counter has its definition from Microsoft in quotation marks and our explanation of the results below its graph. Here we focus on % Idle Time, Disk Bytes/sec, Disk Write Bytes/sec and Disk Writes/sec counters to get a better understanding of how the Logical Disk performance is affected by Ping and Smurf attacks.

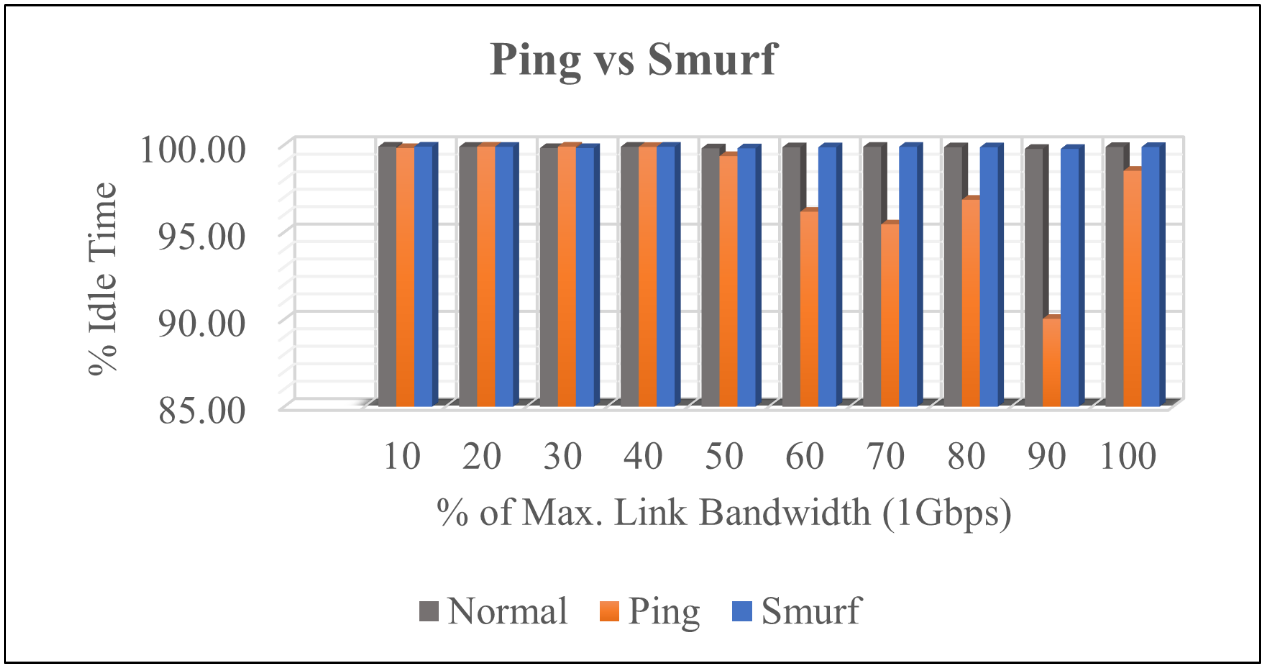


Figure 3 - 4 Logical Disk: Percent Idle Time

“% Idle Time reports the percentage of time during the sample interval that the disk was idle.”

Figure 3 – 4 shows hardly any activity for Logical Disk without cyberattacks, indicated by the gray-shaded bars, marked Normal, which represents the computer activity caused by legitimate traffic alone. During Ping attack, the Logical Disk stayed idle until the 50% attack intensity was reached, at which point the % Idle Time started decreasing, which means that the activity of the Logical Disk increased during Ping attack. During Smurf attack, no logical disk activity was noticed. This graph reveals that Ping attack does affect Logical Disk.

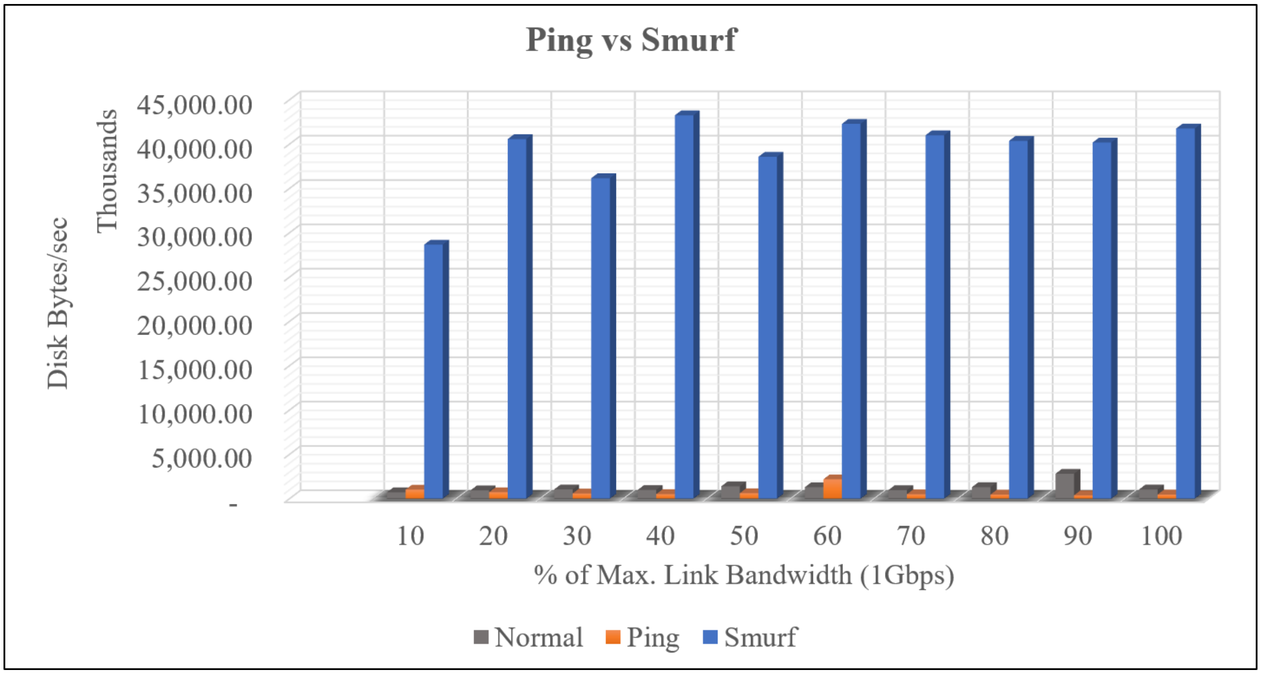


Figure 3 - 5 Logical Disk: Disk Bytes per second

“Disk Bytes/sec is the rate bytes are transferred to or from the disk during write or read operations.”

The graph, Disk Bytes per second in Figure 3 -5 show some revealing results, a very aggressive attack on the Logical Disk by Smurf attack. This is shown by the blue bars a pattern we can follow for the TCP attack. For UDP Attack we can see an immediate rise in the number of bytes being transferred back and forth, when the attack began at 10% of the bandwidth intensity, reaching a maximum close to one thousand bytes. Then, there is a steady decline in the migration of bytes, reaching a minimum at 90% of the bandwidth intensity with around 250 bytes per second. This counter could be studied farther. What happens if the bandwidth continues to increase in intensity to 200%? Would the bytes transferred per second increase, or would they stay below 500 disk bytes per second? On the other hand, we can see that for test when there was no traffic there was a tendency for bytes to increase. This may indicate that there is more communication with the disk when there is no attack than when there is one happening. This could be communication interference and could be due to saturation of processor resources.

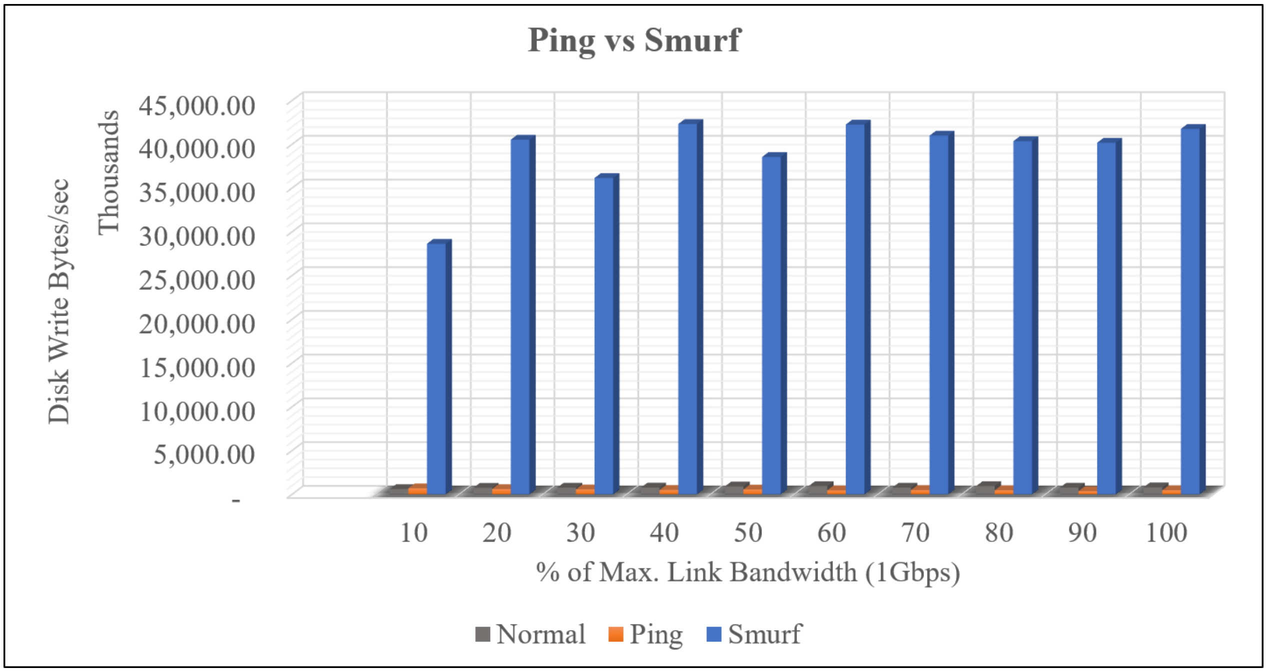
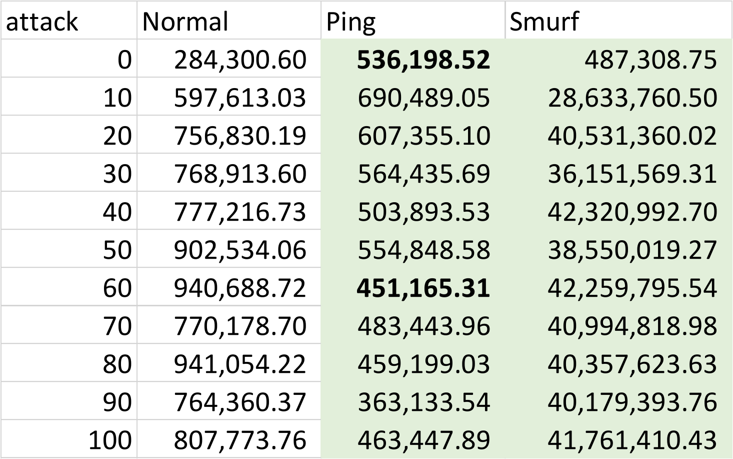


Figure 3 - 6 Logical Disk: Disk Write Bytes per second

“Disk Write Bytes/sec is rate at which bytes are transferred to the disk during write operations.”



attack	Normal	Ping	Smurf
0	284,300.60	**536,198.52**	487,308.75
10	597,613.03	690,489.05	28,633,760.50
20	756,830.19	607,355.10	40,531,360.02
30	768,913.60	564,435.69	36,151,569.31
40	777,216.73	503,893.53	42,320,992.70
50	902,534.06	554,848.58	38,550,019.27
60	940,688.72	**451,165.31**	42,259,795.54
70	770,178.70	483,443.96	40,994,818.98
80	941,054.22	459,199.03	40,357,623.63
90	764,360.37	363,133.54	40,179,393.76
100	807,773.76	463,447.89	41,761,410.43

Table 3 - 1 Logical Disk: Disk Write Bytes per second

Figure 3-6 and Table 3-1 relate to Disk Write Bytes per second. We include the table related to the graph to show the actual values for each attack since the values for Smurf attack cause Normal and Ping attack traffic to not be clearly displayed. With both, table and graph, we can clearly see how Smurf attack causes so many more write bytes to disk than Normal and Ping attack traffic.

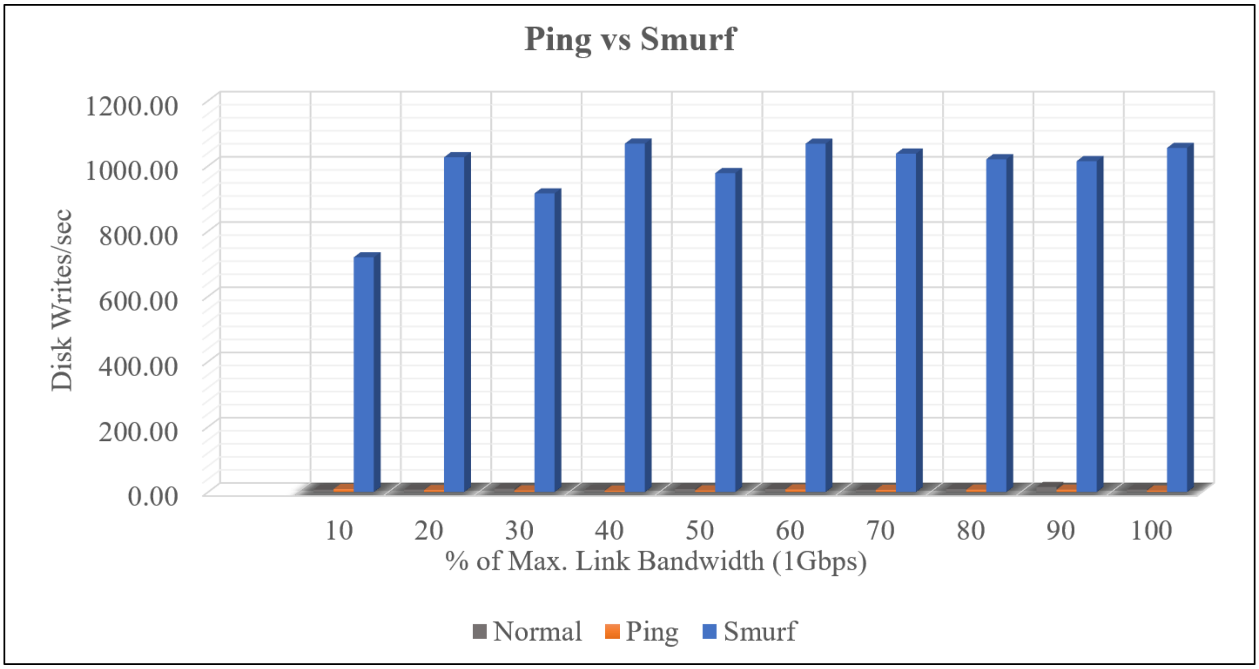
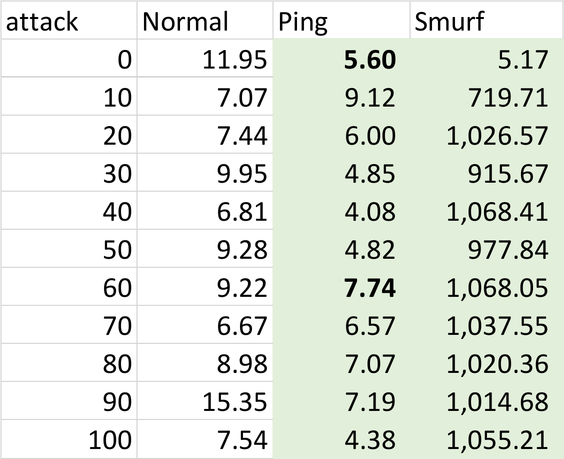


Figure 3 - 7 Logical Disk: Disk Writes per second

"Disk Writes/sec is the rate of write operations on the disk."



attack	Normal	Ping	Smurf
0	11.95	**5.60**	5.17
10	7.07	9.12	719.71
20	7.44	6.00	1,026.57
30	9.95	4.85	915.67
40	6.81	4.08	1,068.41
50	9.28	4.82	977.84
60	9.22	**7.74**	1,068.05
70	6.67	6.57	1,037.55
80	8.98	7.07	1,020.36
90	15.35	7.19	1,014.68
100	7.54	4.38	1,055.21

Table 3 - 2 Logical Disk: Writes per second

The rate of write operations to Logical Disk is shown in Figure 3-7 and Table 3-2. Here we can see how fast the write operations are being transferred to Logical Disk. Smurf attack started at 720 writes per second, on average, when the intensity of the attack was 10%. Then the writes climbed up to 1,026 writes per second. The highest average value recorded is 1,068.41 at 40% of the attack intensity.

3.3.3 Memory

Our Memory performance object contains results from % Committed Bytes In Use, Available Mbytes, and Page Faults/sec. Following is the definition of Memory performance object as given by Microsoft, followed by our test results and the definition for each memory counter we analyzed.

“The Memory performance object consists of counters that describe the behavior of physical and virtual memory on the computer. Physical memory is the amount of random access memory on the computer. Virtual memory consists of the space in physical memory and on disk. Many of the memory counters monitor paging, which is the movement of pages of code and data between disk and physical memory. Excessive paging, a symptom of a memory shortage, can cause delays which interfere with all system processes.”

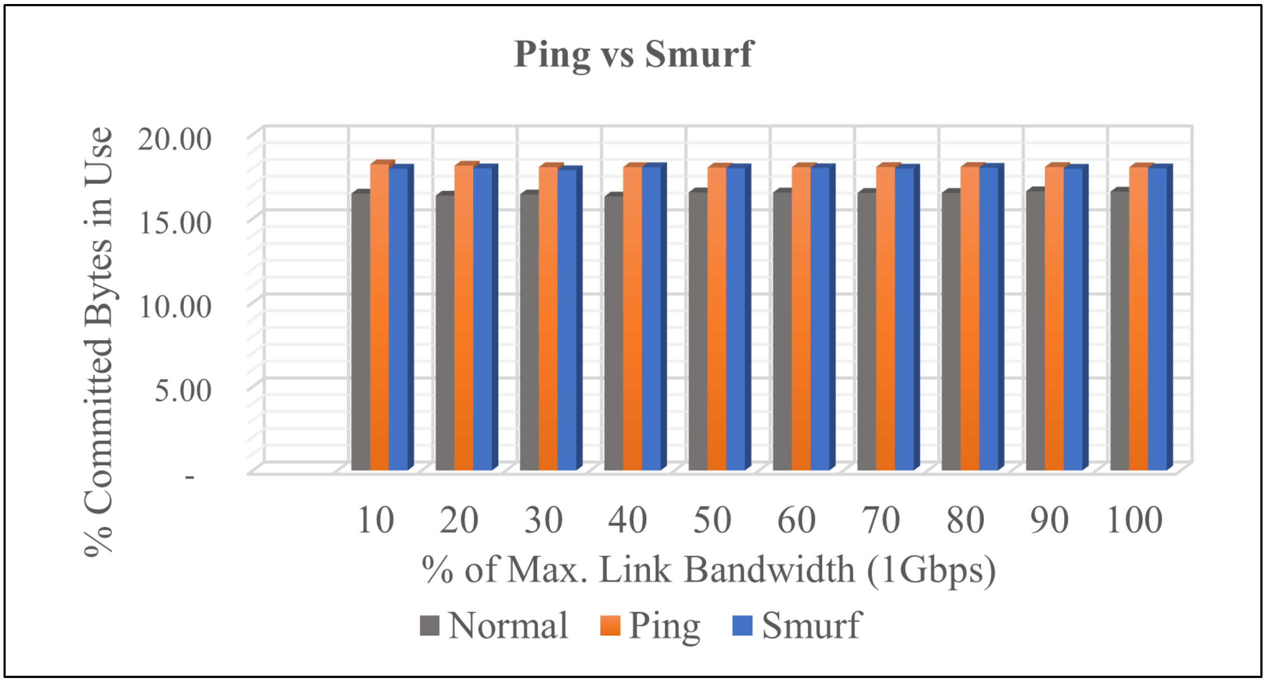


Figure 3 - 8 Memory: Percent Committed Bytes in Use

"% Committed Bytes In Use is the ratio of Memory\\Committed Bytes to the Memory\\Commit Limit. Committed memory is the physical memory in use for which space has been reserved in the paging file should it need to be written to disk. The commit limit is determined by the size of the paging file. If the paging file is enlarged, the commit limit increases, and the ratio is reduced). This counter displays the current percentage value only; it is not an average."

In Figure 3-8 the percentage of memory committed bytes is shown. The computer is equipped with 16 GB of RAM. This graph shows that only about 18% of the 16 GB is being used while either Ping and Smurf attack are happening. This is not much different than the percentage committed when Normal traffic and when the computer was idle.

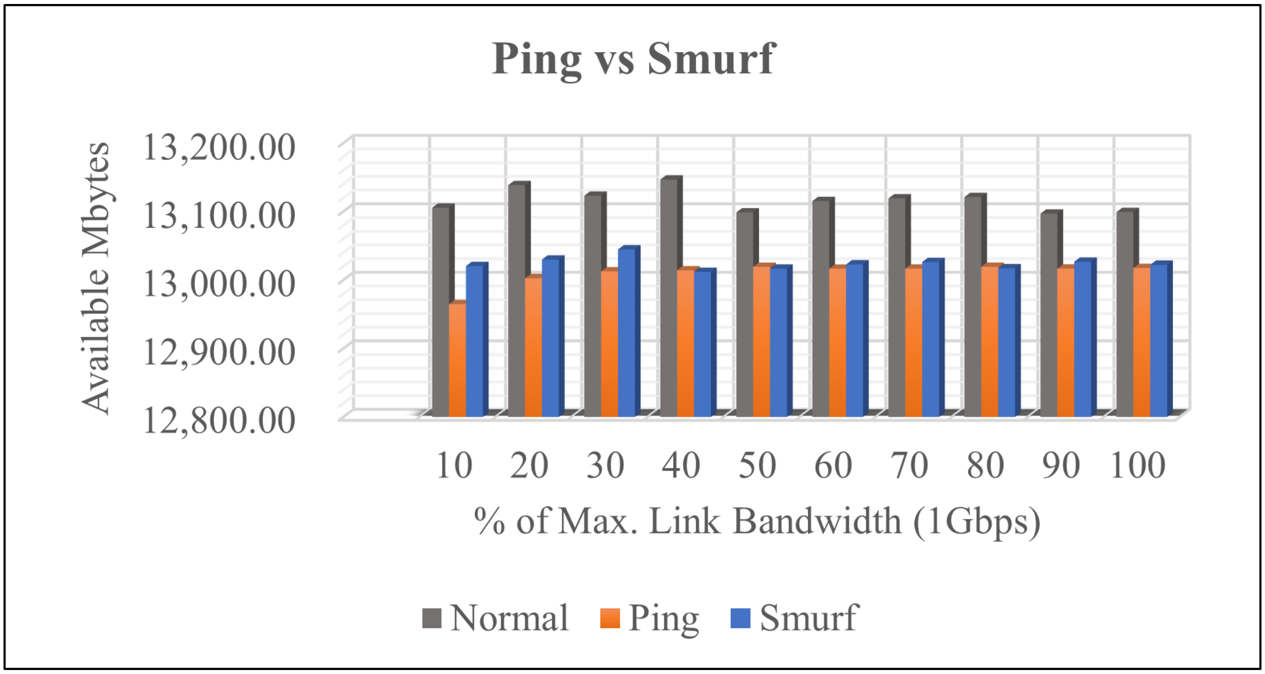
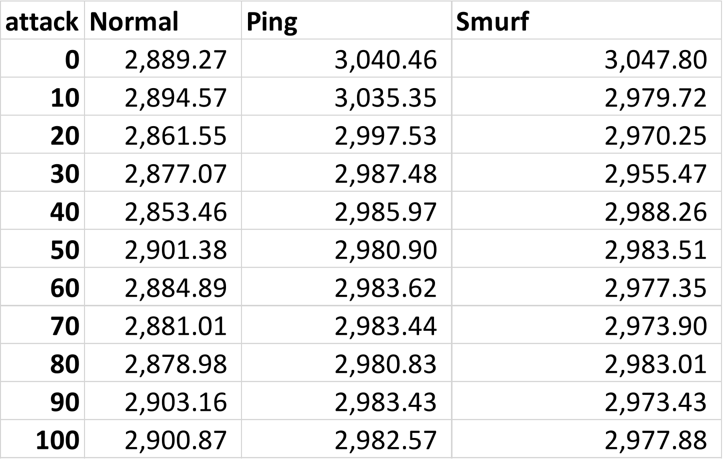


Figure 3 - 9 Memory: Available Megabytes

"Available MBytes is the amount of physical memory, in Megabytes, immediately available for allocation to a process or for system use. It is equal to the sum of memory assigned to the standby (cached), free and zero page lists."

To find out how much memory is being used we must subtract the Available memory from the quantity of main memory that is installed on the computer. In this case we use 16 GB minus the memory displayed on the chart. In Figure 3-9 we can see that the memory usage is being affected by Ping and Smurf attacks. Both Ping and Smurf use about 200 Megabytes more than what is used during Normal traffic.



attack	Normal	Ping	Smurf
0	2,889.27	3,040.46	3,047.80
10	2,894.57	3,035.35	2,979.72
20	2,861.55	2,997.53	2,970.25
30	2,877.07	2,987.48	2,955.47
40	2,853.46	2,985.97	2,988.26
50	2,901.38	2,980.90	2,983.51
60	2,884.89	2,983.62	2,977.35
70	2,881.01	2,983.44	2,973.90
80	2,878.98	2,980.83	2,983.01
90	2,903.16	2,983.43	2,973.43
100	2,900.87	2,982.57	2,977.88

Table 3 - 3 Megabytes Used

On Table 3-3 we can see the actual megabytes used during the three different activities.

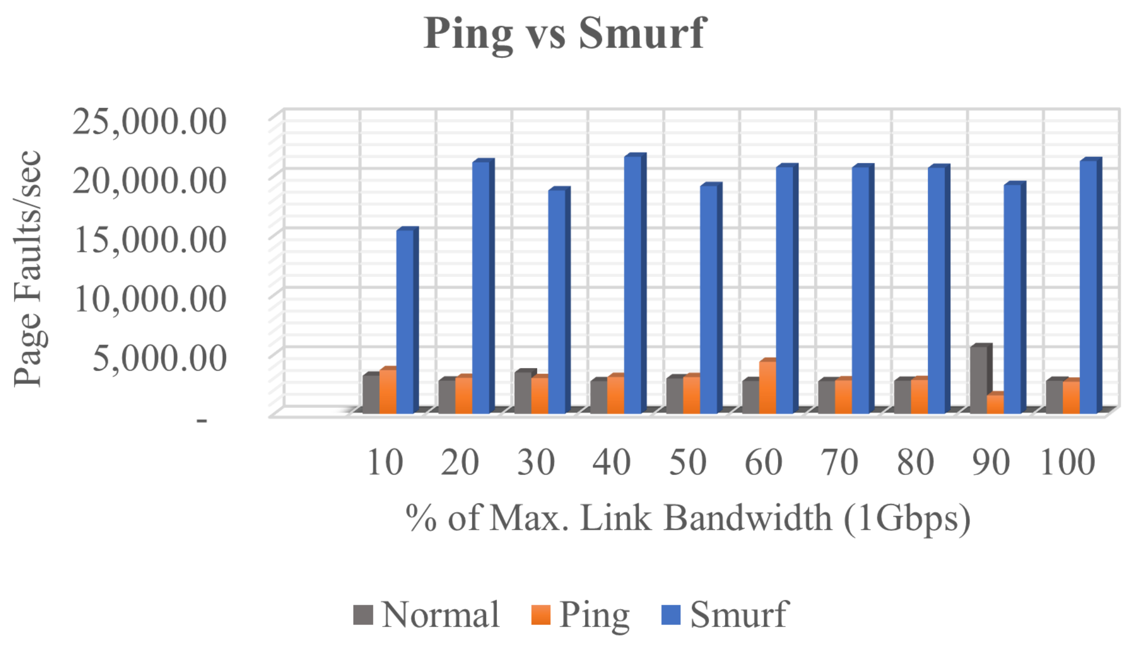


Figure 3 - 10 Memory: Page Faults per second

“Page Faults/sec is the average number of pages faulted per second. It is measured in number of pages faulted per second because only one page is faulted in each fault operation, hence this is also equal to the number of page fault operations. This counter includes both hard faults (those that require disk access) and soft faults (where the faulted page is found elsewhere in physical memory.) Most processors can handle large numbers of soft faults without significant consequence. However, hard faults, which require disk access, can cause significant delays."

Based on the definition of Page Faults/sec by Microsoft and the graph on Figure 3-10, we can determine that Smurf attack generates the most Page Faults per second. Ping attack generates as many Page Faults per second as Normal traffic.

3.3.4 Physical Disk

In Physical Disk performance object we looked into % Disk Time, Disk Write Time, % Idle Time object counters.

"The Physical Disk performance object consists of counters that monitor hard or fixed disk drive on a computer. Disks are used to store file, program, and paging data and are read to retrieve these items, and written to record changes to them. The values of physical disk counters are sums of the values of the logical disks (or partitions) into which they are divided."

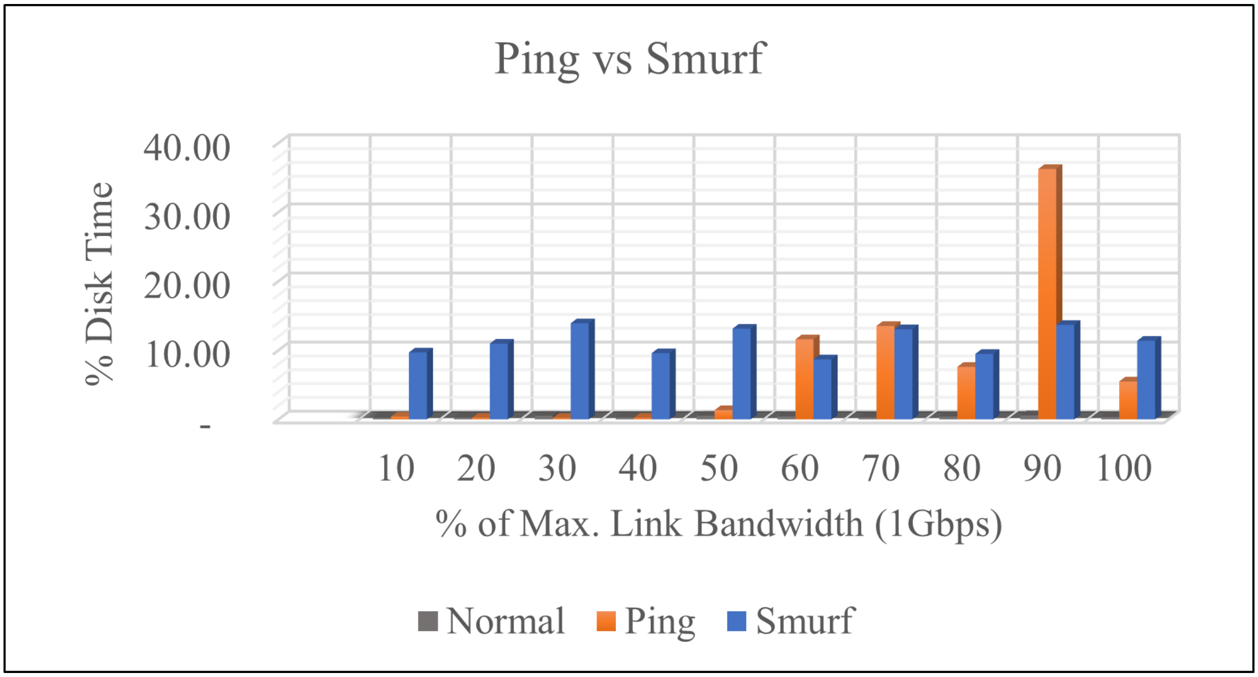


Figure 3 - 11 Physical Disk: Percent Disk Time

“% Disk Time is the percentage of elapsed time that the selected disk drive was busy servicing read or write requests.”

In the graph of Figure 3-11 we can see the percentage of disk time Ping and Smurf attack used. Smurf attack’s percentage use of disk time immediately rises to ten percent when the attack starts, and it stay around that mark until the end of the test for almost all intensities. Ping attack percent usage of disk time does not increase until the 50% attack intensity mark. Before that, Ping attack does not seem to affect disk time. From 60% to 100% attack intensity Ping attack usage of disk time stays above 5%. It’s peak is reached at 90% intensity when it rises to above 35% disk time.

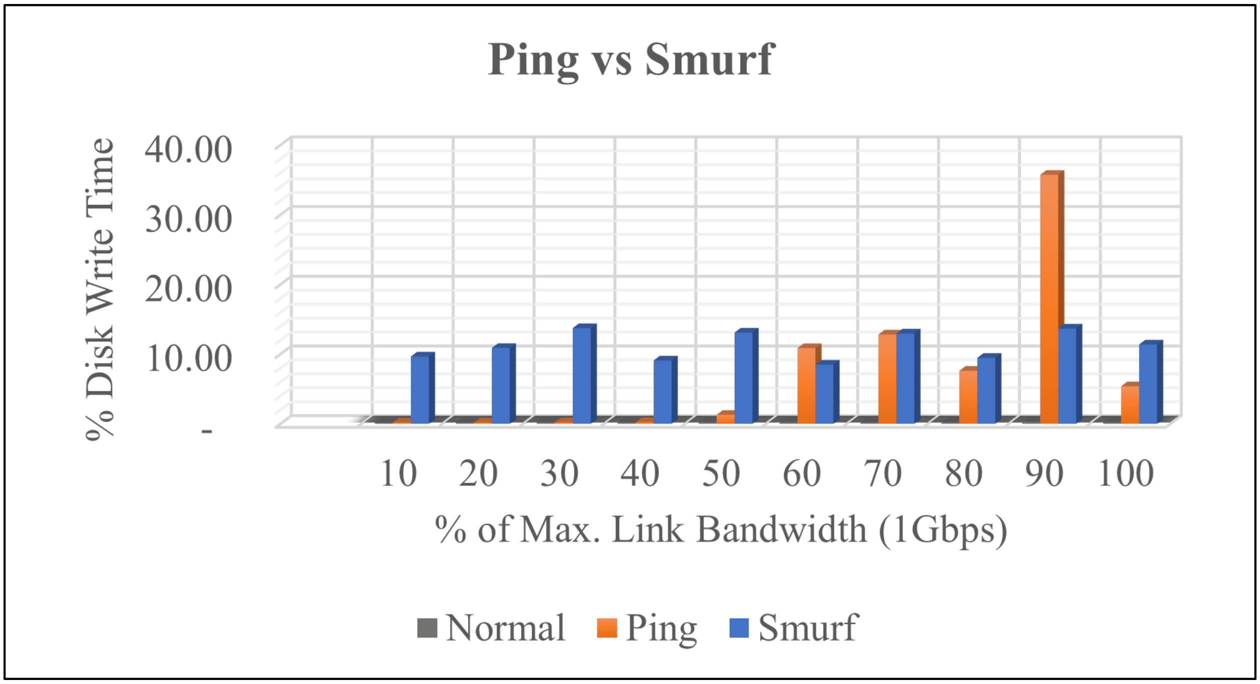


Figure 3 - 12 Physical Disk: Percent Disk Write Time

“% Disk Write Time is the percentage of elapsed time that the selected disk drive was busy servicing write requests.”

In Figure 3-12 we can see the percentage of Disk Write Time. Looking back at Figure 3-11 we can tell that % Disk Time was utilized servicing write request more than reads.

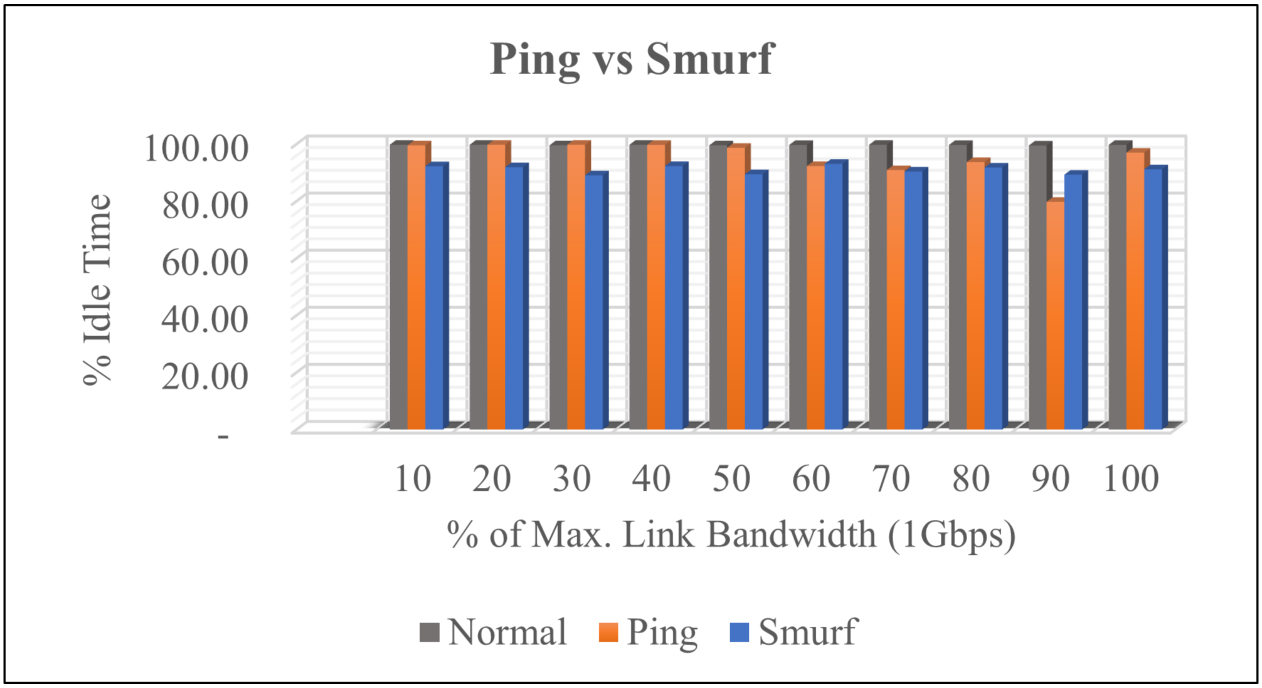


Figure 3 - 13 Physical Disk: Percent Idle Time

"% Idle Time reports the percentage of time during the sample interval that the disk was idle."

Figure 3-13 shows the percentage of time physical disk was not busy (idle). Here we see that during Smurf attack Physical Disk was idle about 90% of the time. During Ping attack Physical Disk was idle close to 100% of the time, except when the attack reached 60% of its intensity, at which point the idle time reduced to 90%. This information corresponds to the results gathered in Figure 3-12 and Figure 3-13.

3.3.5 Process

The Process performance object, in our test, is composed of the following performance counters: % Privileged Time, % Processor Time, % User Time, I/O Write Operations/sec, Page Faults/sec.

"The Process performance object consists of counters that monitor running application program and system processes. All the threads in a process share the same address space and have access to the same data."

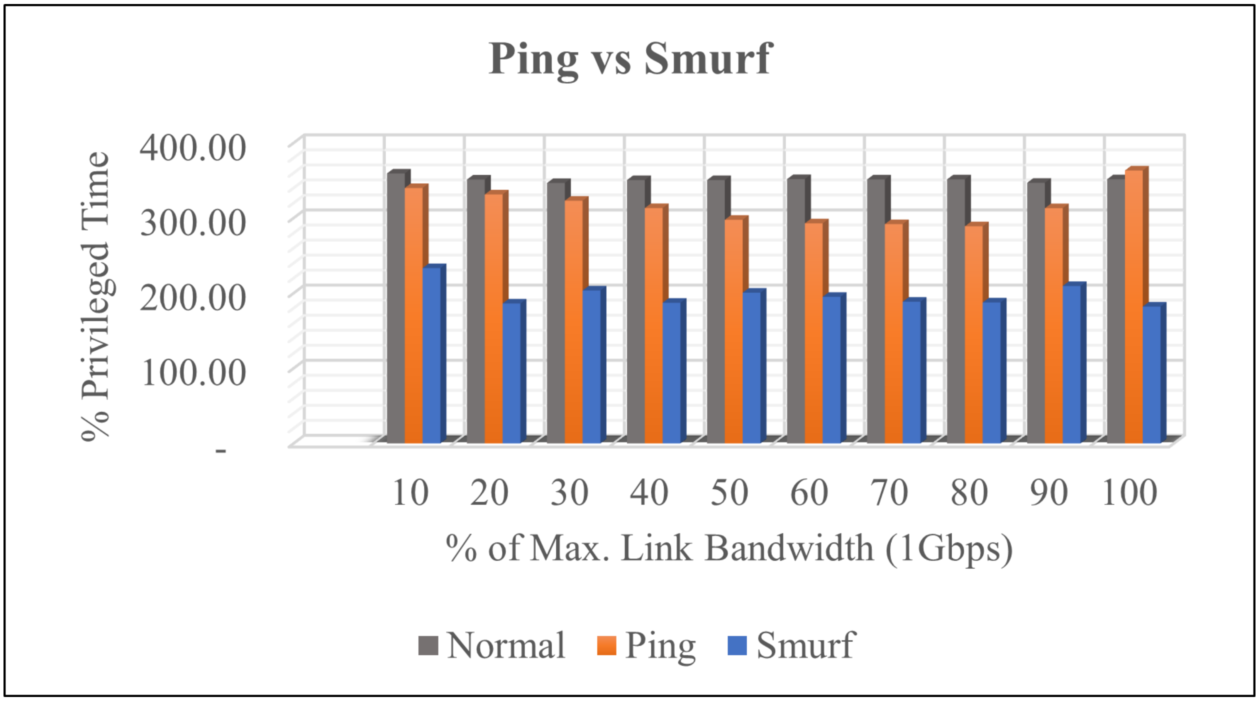


Figure 3 - 14 Process: Percent Privileged Time

“% Privileged Time is the percentage of elapsed time that the process threads spent executing code in privileged mode. When a Windows system service is called, the service will often run in privileged mode to gain access to system-private data. Such data is protected from access by threads executing in user mode. Calls to the system can be explicit or implicit, such as page faults or interrupts. Unlike some early operating systems, Windows uses process boundaries for subsystem protection in addition to the traditional protection of user and privileged modes. Some work done by Windows on behalf of the application might appear in other subsystem processes in addition to the privileged time in the process. “

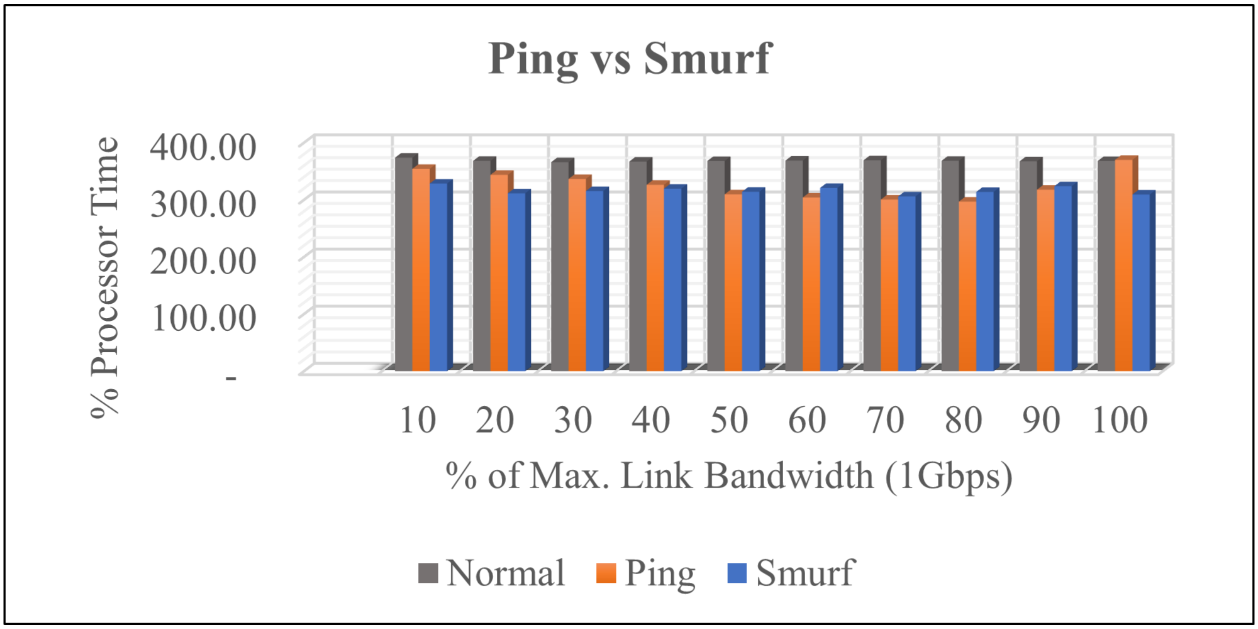


Figure 3 - 15 Process: Percent Processor Time

“% Processor Time is the percentage of elapsed time that all of process threads used the processor to execution instructions. An instruction is the basic unit of execution in a computer, a thread is the object that executes instructions, and a process is the object created when a program is run. Code executed to handle some hardware interrupts and trap conditions are included in this count.”

Percent Processor Time in Figure 3-15 and Percent Privileged Time in Figure 3-14 need a more in-depth analyzes as the percentages are out of any logic. It seems like the base line of this process is above 350% for Normal traffic reaching close to 400%.

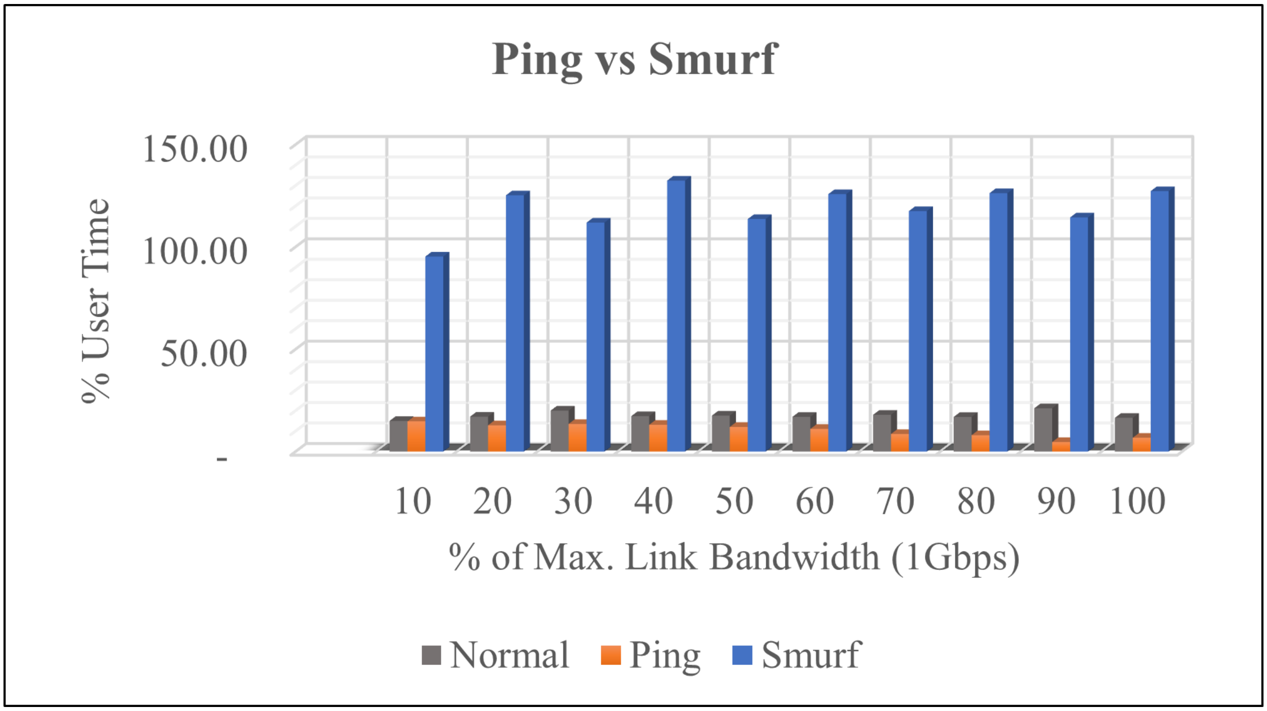


Figure 3 - 16 Process: Percent User Time

"% User Time is the percentage of elapsed time that the process threads spent executing code in user mode. Applications, environment subsystems, and integral subsystems execute in user mode. Code executing in user mode cannot damage the integrity of the Windows executive, kernel, and device drivers. Unlike some early operating systems, Windows uses process boundaries for subsystem protection in addition to the traditional protection of user and privileged modes. Some work done by Windows on behalf of the application might appear in other subsystem processes in addition to the privileged time in the process."

Figure 3-16 is the Percentage User Time that user mode was used to execute process threads. Here we see Smurf attack using 100% more the user mode than a Ping attack. A Ping attack seems to suppress the use of User Mode. We can see this when we compare the Ping attack traffic to the Normal traffic.

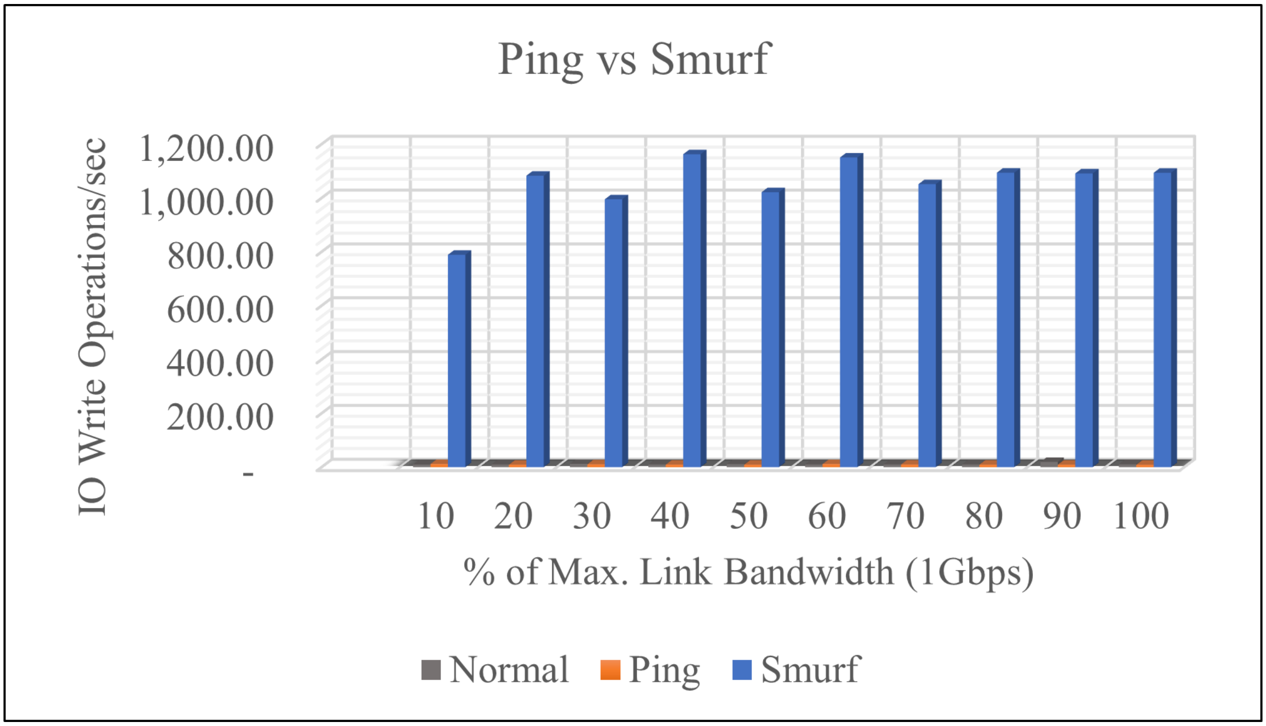


Figure 3 - 17 Process: Input/Output Write Operations per second

> “The rate at which the process is issuing write I/O operations. This counter counts all I/O activity generated by the process to include file, network and device I/Os.”

Figure 3-17 is another good graph of the activity of a Smurf attack compare to a Ping attack. Here we can see that the Input/Output Operations per second is very high during a Smurf attack. This indicates that a Smurf attack forces the computer to issue write operations that can slow it down.

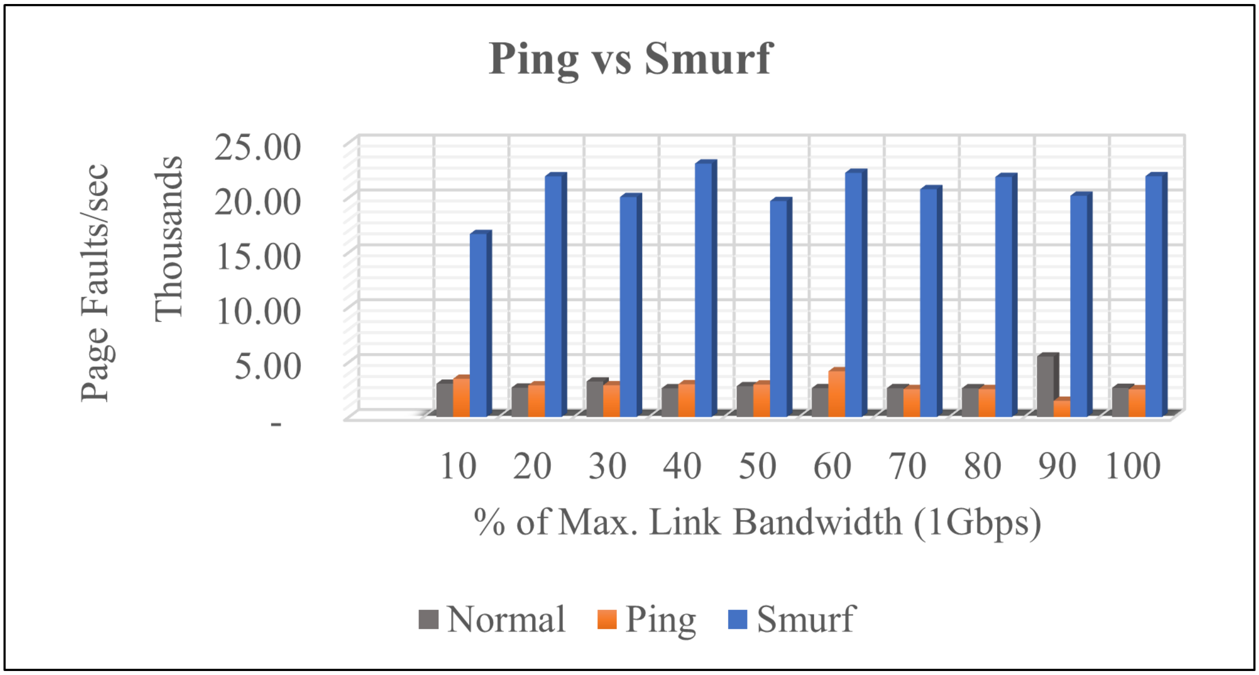


Figure 3 - 18 Process: Page Faults per second

"Page Faults/sec is the rate at which page faults by the threads executing in this process are occurring. A page fault occurs when a thread refers to a virtual memory page that is not in its working set in main memory. This may not cause the page to be fetched from disk if it is on the standby list and hence already in main memory, or if it is in use by another process with whom the page is shared."

In Figure 3-18 we can see the excess of page faults that happen during a Smurf attack compared to the page faults happening during Ping attack.

3.3.6 Processor

% Idle Time, % Processor Time, and Interrupts/sec are counters selected to represent the activity of the Processor performance object.

"The Processor performance object consists of counters that measure aspects of processor activity. The processor is the part of the computer that performs arithmetic and logical computations, initiates operations on peripherals, and runs the threads of processes. A computer can have multiple processors. The processor object represents each processor as an instance of the object."

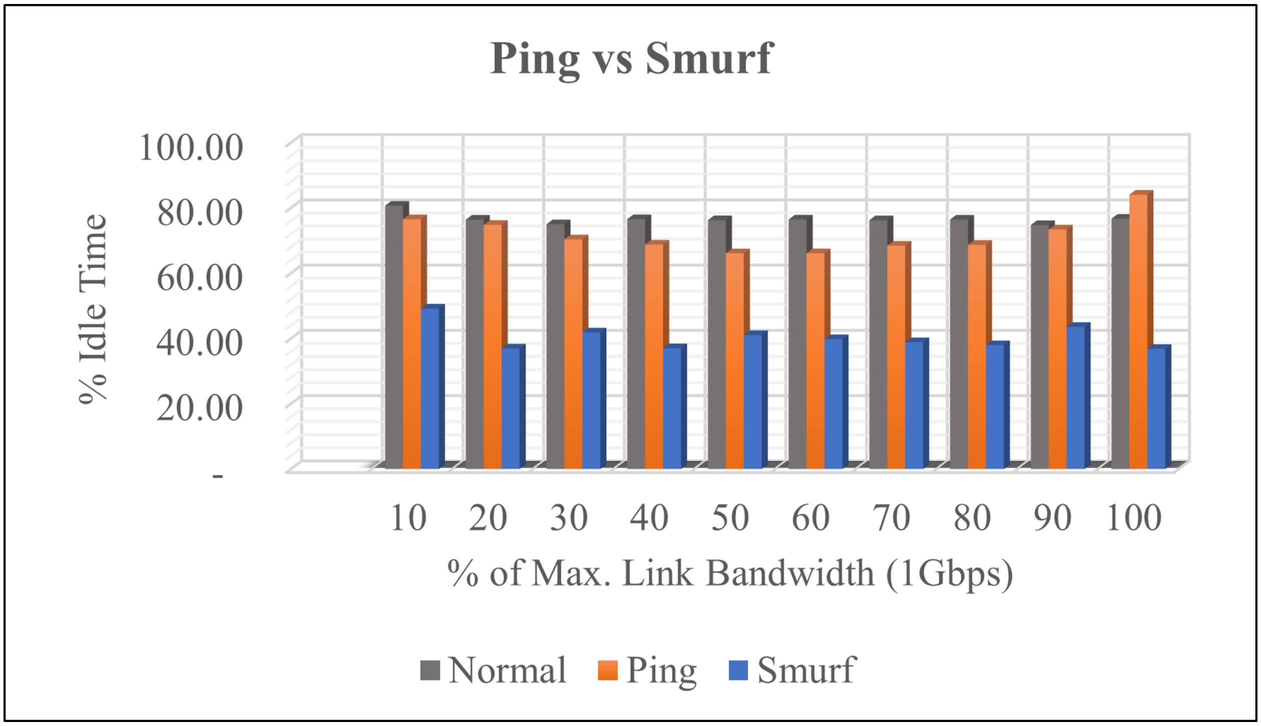


Figure 3 - 19 Processor: Percent Idle Time

"% Idle Time is the percentage of time the processor is idle during the sample interval"

The Percent Idle Time of Processor can be understood as the time the processor was active. If we look at the empty space of each graph as the time the processor was active we can better understand. For example, in Figure 3-19, looking at Smurf attack graph (the blue bars), we can tell the processor performance object was active around 40% of the time after the attack reached 10% its intensity. Ping, on the other hand, hardly differed its activity from a normally active computer.

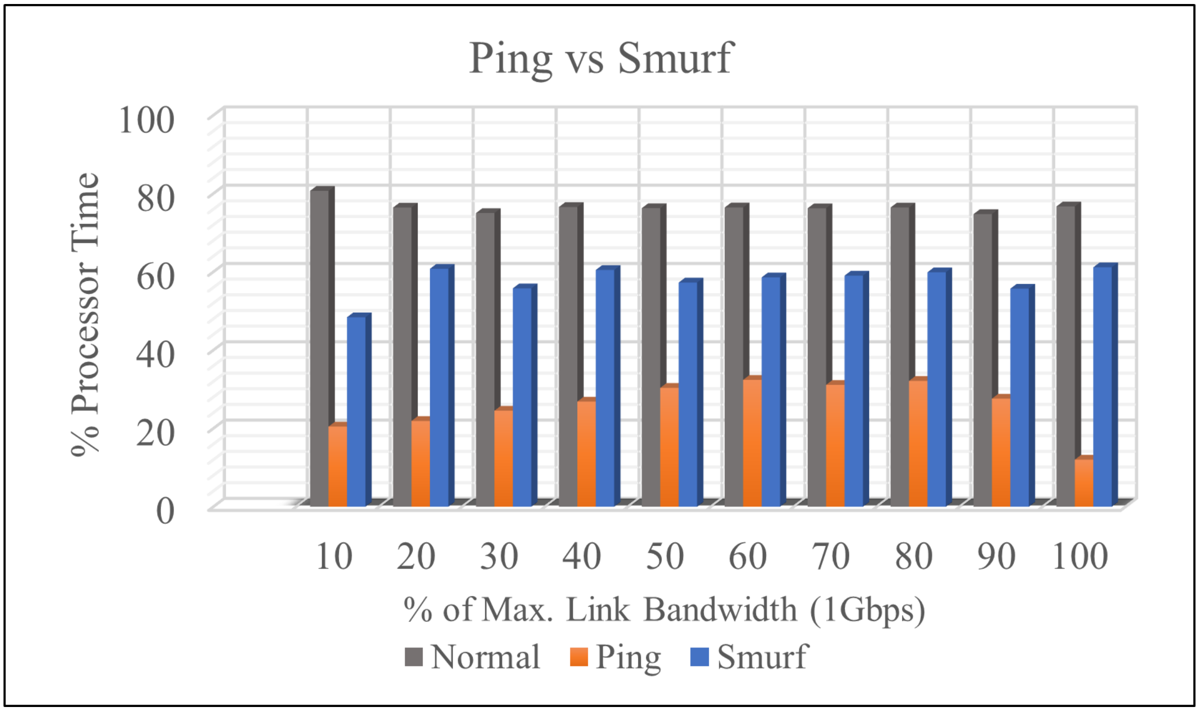


Figure 3 - 20 Processor: Percent Processor Time

“% Processor Time is the percentage of elapsed time that the processor spends to execute a non-Idle thread. It is calculated by measuring the percentage of time that the processor spends executing the idle thread and then subtracting that value from 100%. (Each processor has an idle thread that consumes cycles when no other threads are ready to run). This counter is the primary indicator of processor activity, and displays the average percentage of busy time observed during the sample interval. It should be noted that the accounting calculation of whether the processor is idle is performed at an internal sampling interval of the system clock (10ms). On todays fast processors, % Processor Time can therefore underestimate the processor utilization as the processor may be spending a lot of time servicing threads between the system clock sampling interval. Workload based timer applications are one example of applications which are more likely to be measured inaccurately as timers are signaled just after the sample is taken.”

Figure 3-19 and Figure 3-20 are related in the fact that they are opposites of each other. Figure 3-19 shows the processor’s idle time and Figure 3-20 is the processor time. There are some discrepancies, however. In Normal traffic, at the zero percent bar we can see both instances reach ninety percent. If these two counters are opposites of each other this should not happen. The same happens with the rest of the bars for Normal traffic, every bar reaches eighty percent or close to it. The bars that display the attack information both are consistent with the logic of these two counters being opposites but there seems to be a difference when analyzing the data closely.

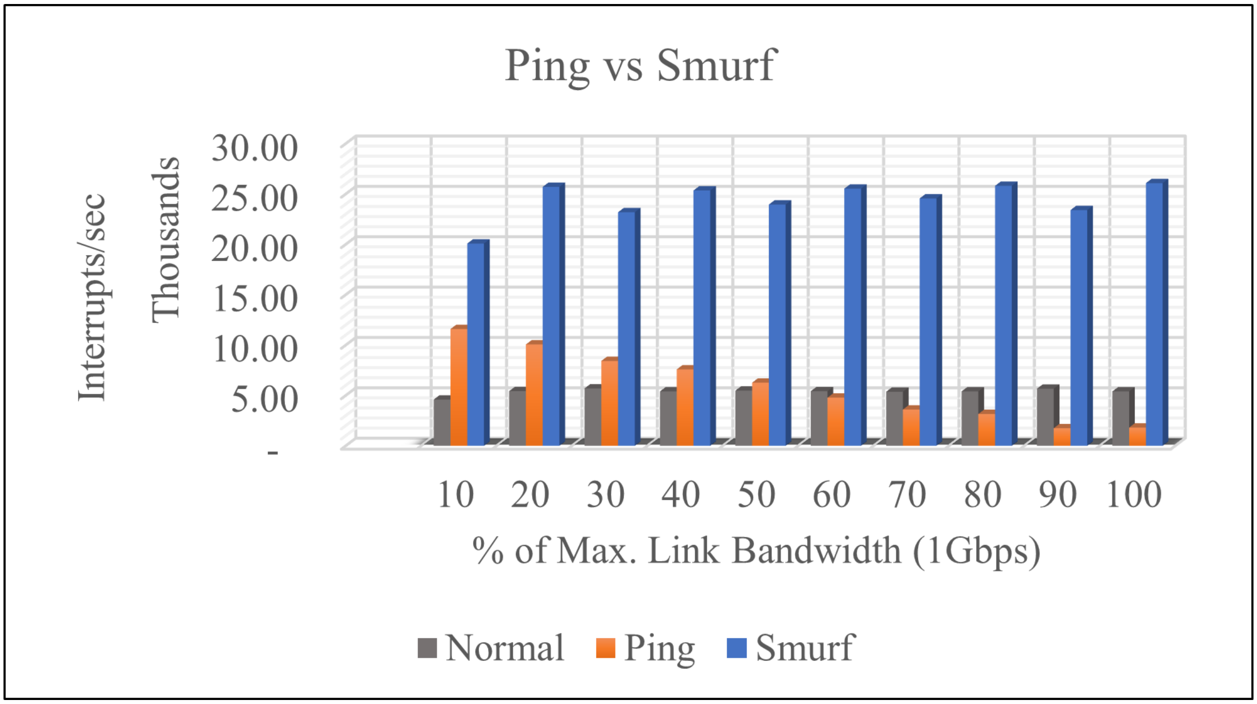


Figure 3 - 21 Processor: Interrupts per second

“Interrupts/sec is the average rate, in incidents per second, at which the processor received and serviced hardware interrupts. It does not include deferred procedure calls (DPCs), which are counted separately. This value is an indirect indicator of the activity of devices that generate interrupts, such as the system clock, the mouse, disk drivers, data communication lines, network interface cards, and other peripheral devices. These devices normally interrupt the processor when they have completed a task or require attention. Normal thread execution is suspended. The system clock typically interrupts the processor every 10 milliseconds, creating a background of

> interrupt activity. This counter displays the difference between the values observed in the last two samples, divided by the duration of the sample interval."

In Figure 3-21 we can see that the Interrupt activity for Smurf attack is great, exceeding that of Ping attack and Normal traffic. We can see that Ping attack Interrupts activity is greatest at the lower intensities. This could be due to the data transferring. Indicating that at the earlier lower intensities the processor cache is busy gathering new data from different levels of memory. Once the data is in cache the processor does not have to be so active fetching new data from other places than processor cache.

3.3.7 System

System performance object is analyzed in the paper using the following counters: % Registry Quota In Use, File Write Bytes/sec, and File Write Operations/sec and System Calls/sec.

> "The System performance object consists of counters that apply to more than one instance of a component processors on the computer."

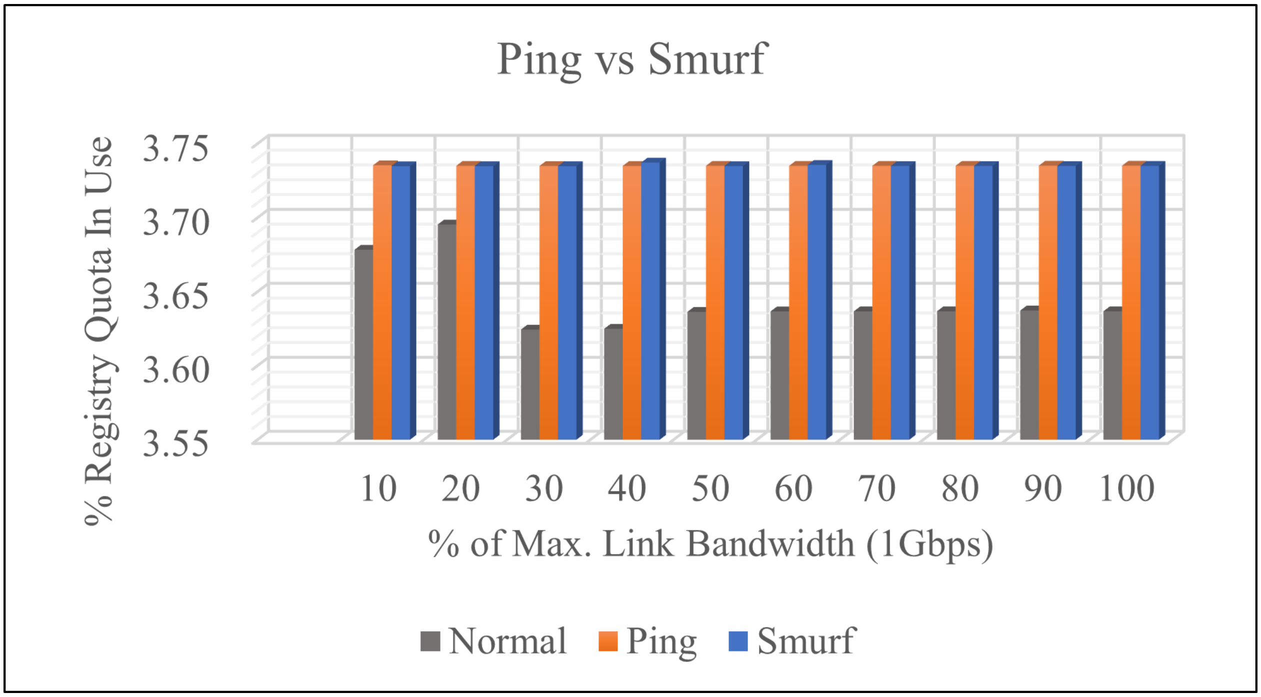


Figure 3 - 22 System: Percent Registry Quota In Use

"% Registry Quota In Use is the percentage of the Total Registry Quota Allowed that is currently being used by the system. This counter displays the current percentage value only; it is not an average."

The registry, as explained in the book Internet Information Services (IIS) 6.0 by Microsoft Press, holds all type of information related to the computer, from what type of fonts the system uses to decryption classes, and component information that make the computer. The description of this counter says it is not 'an average' but we took samples every two seconds. Then, we averaged those samples to get this graph averaging those samples.

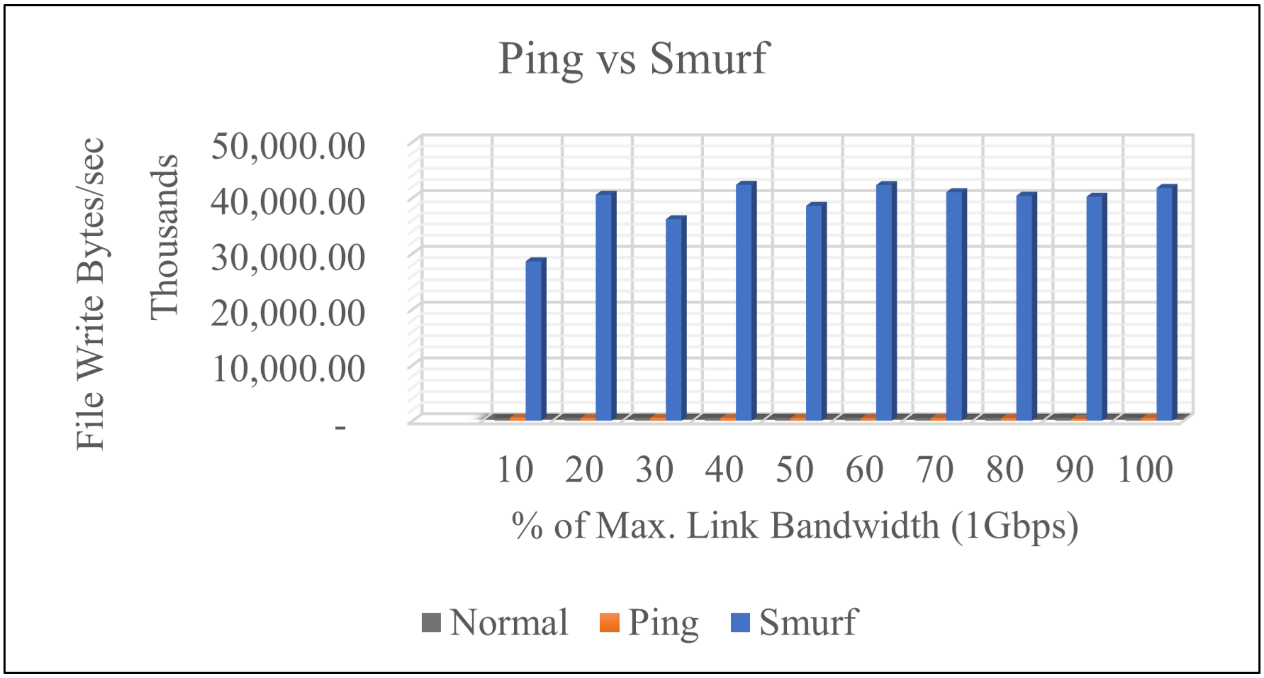


Figure 3 - 23 System: File Write Bytes per second

“File Write Bytes/sec is the overall rate at which bytes are written to satisfy file system write requests to all devices on the computer, including writes to the file system cache. It is measured in number of bytes per second. This counter displays the difference between the values observed in the last two samples, divided by the duration of the sample interval.”

File Write Bytes/sec counter in Figure 3-23 displays some interesting results. Here we can clearly see how a Smurf attack causes many more write requests to the files system compared to what a Ping attack and a Normal computer activity causes.

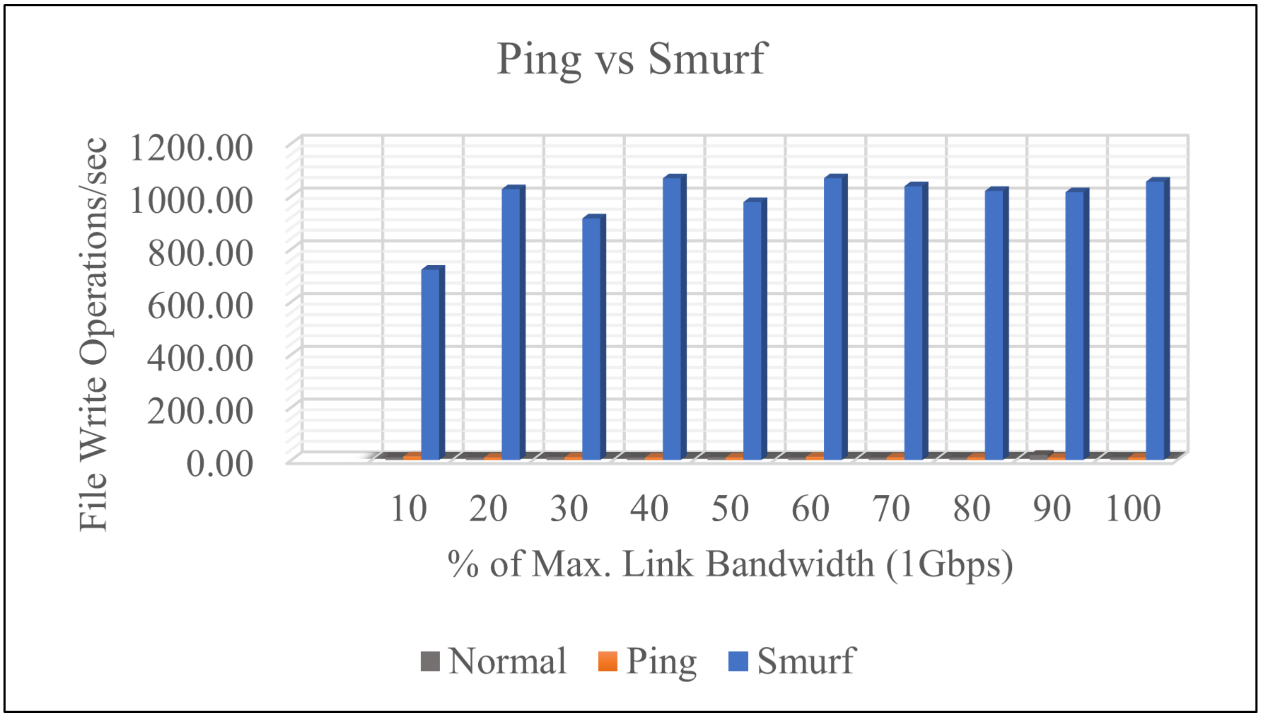


Figure 3 - 24 System: File Write Operations per second

System Calls/sec, System Calls/sec "Operations/sec is the combined rate of the file system write requests to all devices on the computer, including requests to write to data in the file system cache. It is measured in numbers of writes. This counter displays the difference between the values observed in the last two samples, divided by the duration of the sample interval."

It is interesting how File Write Operations increase so drastically for Smurf attack compared to Normal traffic and Ping attack. Both Normal traffic and Ping attack have File Write Operations per second under one hundred while Smurf attack chart displays from 700 to 1100 File Write Operations per second. Perhaps the way Smurf attack drains the resources of the computer is by increasing its File Write Operations.

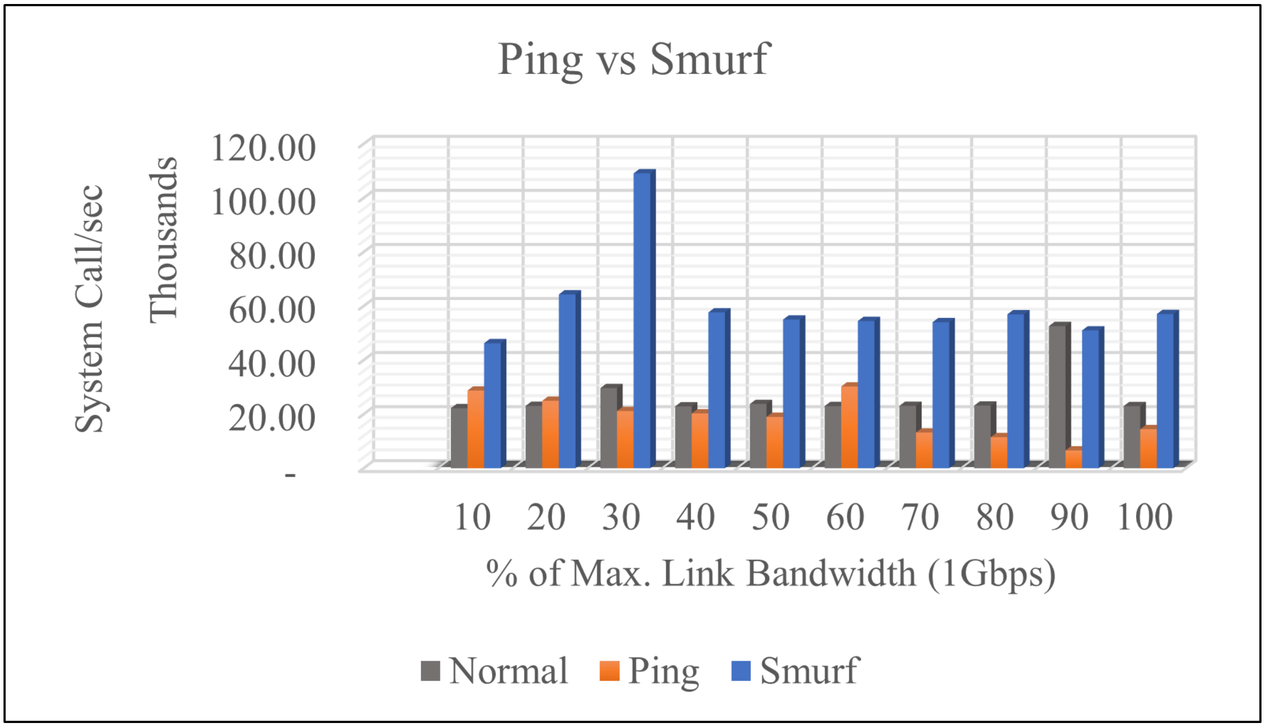


Figure 3 - 25 System: System Calls per second

"System Calls/sec is the combined rate of calls to operating system service routines by all processes running on the computer. These routines perform all of the basic scheduling and synchronization of activities on the computer, and provide access to non-graphic devices, memory management, and name space management. This counter displays the difference between the values observed in the last two samples, divided by the duration of the sample interval."

Figure 3-25 displays the calls to the operating system service routines made by computer processes. Here, again, we can see the dominance of a Smurf attack over the computer as compare with a Ping attack and the Normal computer activity when servicing legitimate traffic.

3.4 CONCLUDING REMARKS

In this chapter we presented an in-depth look at what happens inside a computer when a Layer 3.5 cyberattack is affecting it. We observed seven computer performance objects to get a better understanding of what happens to the computer. Here, we focus on the counters that show activity that could give us a better understanding to compare the results to a Layer 3.5 type of attack. The results for a Layer 4 type of attack are presented in chapter 4. For Cache, we looked at three elements that gave use the most activity: Dirty Pages, Dirty page threshold and Pin Read Hits %,. In Logic Disk we looked at % Idle Time, Disk Bytes/sec, Disk Write Bytes/sec and Disk Writes/sec. The Memory object is analyzed using % Committed Bytes In Use, Available Mbytes, and Page Faults/sec. In Physical Disk performance object we looked into % Disk Time, Disk Write Time, % Idle Time object counters. The Process performance object, in our test, is composed of the following performance counters: % Privileged Time, % Processor Time, % User Time, I/O Write Operations/sec, Page Faults/sec. % Idle Time, % Processor Time, and Interrupts/sec are counters selected to represent the activity of the Processor performance object. Lastly, System performance object is analyzed in the paper using the following counters: % Registry Quota In Use, File Write Bytes/sec, and File Write Operations/sec and System Calls/sec.

Our observation of the computer activity using the above-mentioned performance objects how a computer's activity is affected when it is attack by commonly known DDoS attacks. We conclude that a Smurf attack is more damaging to a Microsoft Surface Pro 4 computer than a Ping attack in the Layer 3.5. In this chapter we compare Ping attack with Smurf attack. Two commonly known cyberattacks that affect networks around the world. In Chapter 5 we will compare the most damaging of this, Layer 3.5, to the most damaging attack from Layer 4.

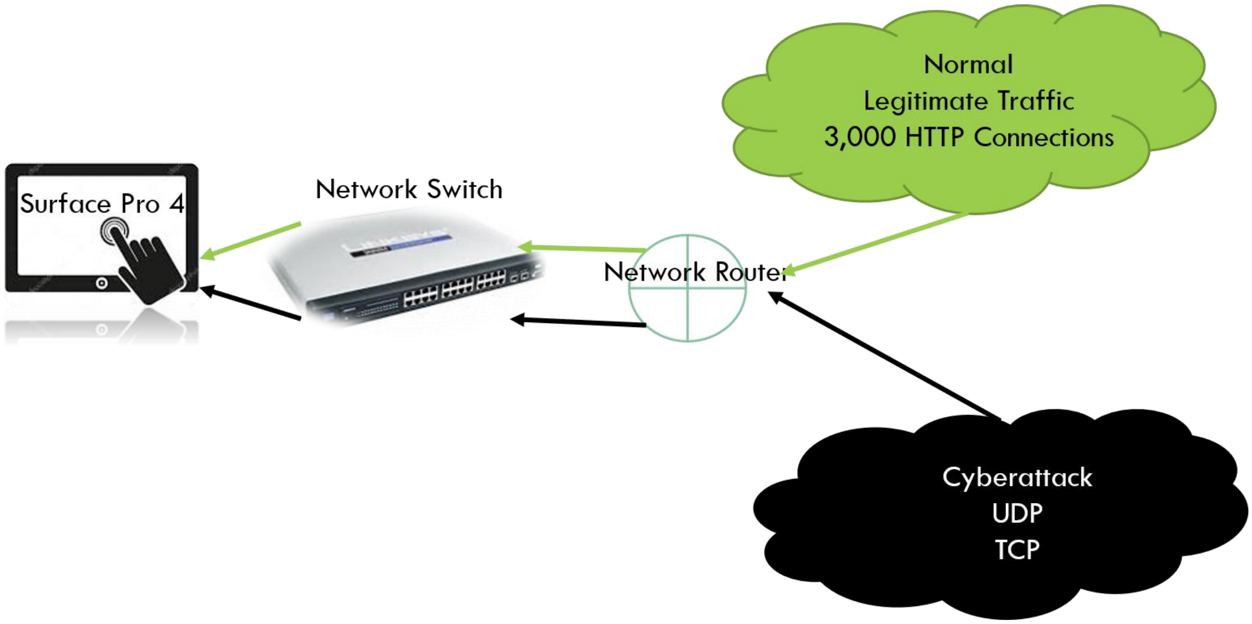
CHAPTER IV

LAYER 4 CYBERATTACK: TCP VERSUS UDP ATTACK RESULTS

4.1 Experimental Setup

In this chapter we cover the elements we used to perform our Layer 4 Cyberattack tests TCP and UDP. In the first section we include the hardware and software that was necessary to run our tests. Then, we explain our results in section two. Section two is divided into definition of the object, both definitions of the counters are in quotation marks as given by Microsoft, and the graph representing the results obtained for that part of our test. The graph is followed by our description of the results. In section three, we conclude with a synthesis of the experiment we talk about in this chapter.

In Schematic 4-1, we can see the setup of the experiment we ran to analyze the performance of SP4 under UDP attack and TCP attack. Going from left to right, the two clouds represent the Internet. The green cloud represents legitimate user connections. These connections represent a simulation of 3,000 HTTP connections. The black cloud represents simulated cyberattack. This attack traffic is added to the legitimate traffic caused by Normal traffic. The traffic from these two cloud networks comes into the network through the router, which in this case is simulated and represented by the round figure with two lines going through the center. The Network Switch simply connects multiple devices in our network. The device on the far left represents the SP4 device we tested. This is how we test how well a computer or server performs under cyberattacks. Below is a list of the actual devices we used and some of their features.



Schematic 4 - 1 : UDP Attack and TCP Attack Experimental Setup

4.2 Hardware and Software tools

The tools we used in this project are like the tools we used in chapter 3. We used Microsoft Surface Pro 4 as our victim device. It is built with an Intel Core i7 processor, which runs at 2.2 GHz [4]. We used Ethernet-to-USB 3.0 adaptor from Freegene to transfer the traffic generated for our test on the SP4. Data was collected using Performance Monitor, a feature embedded in Windows operating systems to analyze the performance of many computer parameters, including Cache, Logic Disk, Memory, Physical Disk, Process, Processor and System. The data comes in a comma separated value (CSV) file, so we used Microsoft Office Excel to analyze our data and convert it to graphs that can be easier to read. We learned that CSV files are not very reliable when you need to make changes to it. Many times, the data was lost when we tried graphing the data directly from a CSV file. To not go through the same issue any more we converted the extracted CSV file to excel file. This work out better because we were able to create our graphs and we preserved the integrity of the original data in CSV format. We worked from the excel file and kept the CSV file untouched for the remaining of the project.

4.3 Performance Objects of Interest

The computer parameters we use in our evaluation are presented in this chapter. We used ten different traffic intensities from 10% to 100% in our comparisons. We grouped these intensities, comparing the base traffic with the cyberattacks to see the distinction and effects the intensities of the attacks have on the computer. Base results are represented in 10% to 100% in gray bars. We let the computer run with simulated legitimate traffic combined to capture a representation of it, to then compare it to that of cyberattacks. TCP attack traffic is represented by red bars and UDP attack traffic is represented by yellow bars.

4.3.1 Cache

"The Cache performance object consists of counters that monitor the file system cache, an area of physical memory that stores recently used data as long as possible to permit access to the data without having to read from the disk. Because applications typically use the cache, the cache is monitored as an indicator of application I/O operations. When memory is plentiful, the cache can grow, but when memory is scarce, the cache can become too small to be effective."

In Cache performance object we focus on Dirty Page Threshold, Dirty Pages, and Pin Read Hits %. Each counter has its definition from Microsoft in quotation marks and our explanation of the results below its graph.

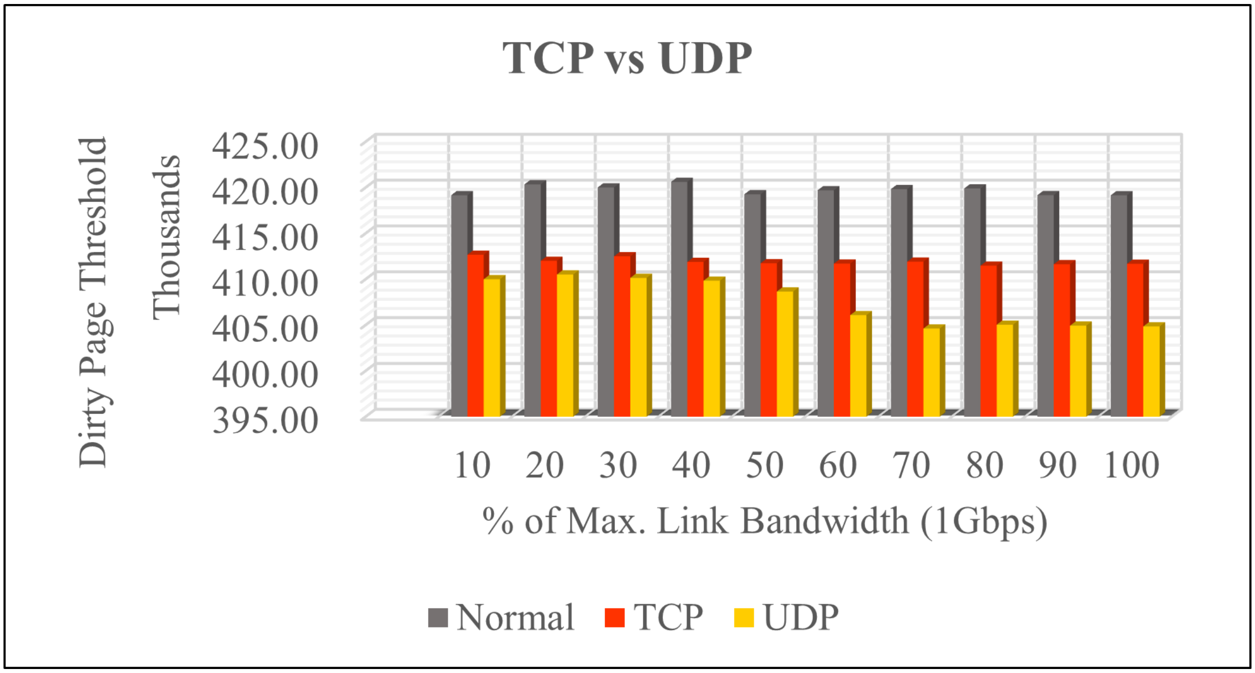


Figure 4 - 1 Cache: Dirty Page Threshold

“Threshold for number of dirty pages on system cache”

In Figure 4-1 the Dirty Page Threshold is affected the most during a UDP attack, it gets reduced the most. The threshold of dirty pages remains at the same limit throughout the eleven intervals for TCP attack, while UDP’s threshold is reduced as the attack intensity increases. According to [10] and [11] this is due to the operating system reducing the threshold to get rid of the dirty pages in the cache quicker to keep the system from failing, which would affect the performance of the computer and integrity and availability of data or any output needed.

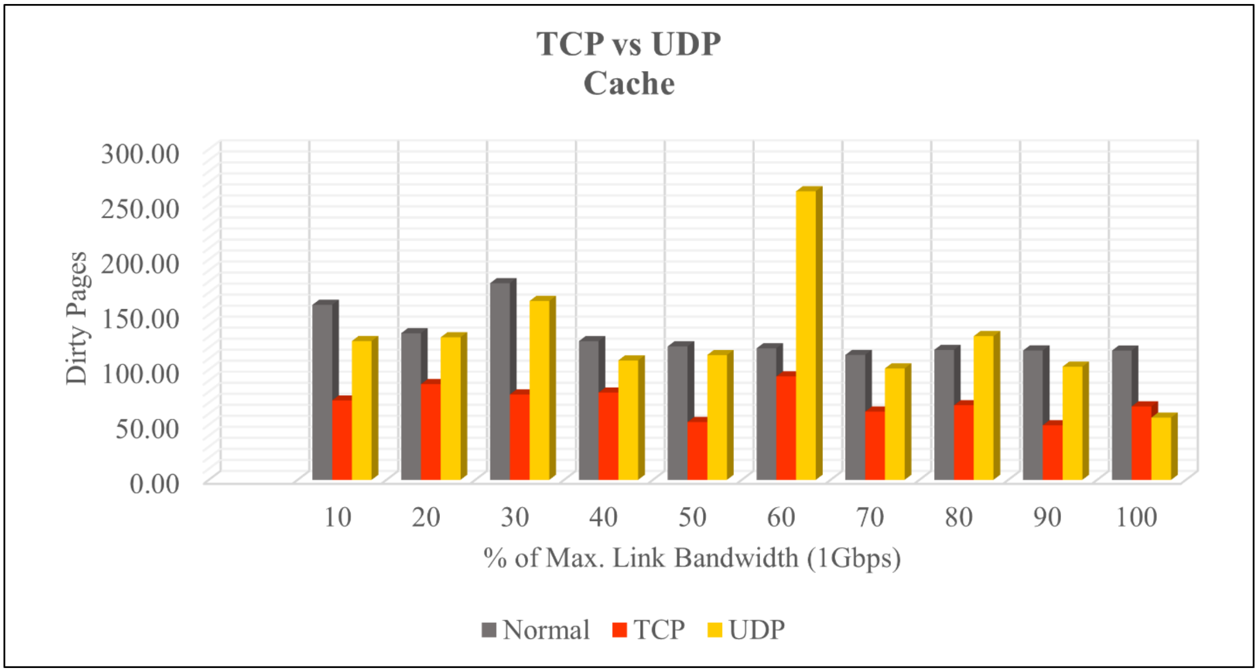


Figure 4 - 2 Cache: Dirty Pages

"Total number of dirty pages on the system cache"

Figure 4 – 2 shows the effect a TCP attack has on Dirty Pages compared to UDP. It is not very clear what is going on here. From the graph we can tell the number of dirty pages during a UDP attack are more than during the TCP attack. However, there are instances where a more in-depth analysis needs to be done. For example, at 60% of the attack intensity UDP suffers a spike increase in dirty pages, while during the TCP attack the number of dirty pages stay close to the trend.

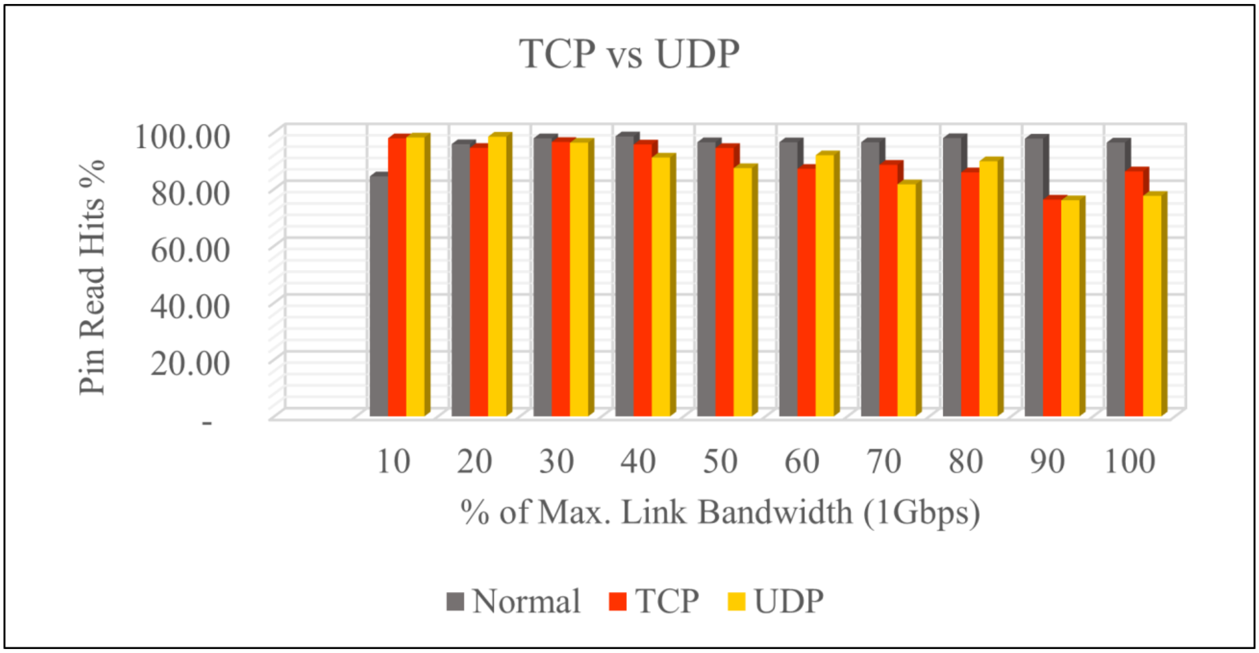


Figure 4 - 3 Cache: Pin Read Hits Percent

“Pin Read Hits is the percentage of pin read requests that hit the file system cache, i.e., did not require a disk read in order to provide access to the page in the file system cache. While pinned, a page's physical address in the file system cache will not be altered. The LAN Redirector uses this method for retrieving data from the cache, as does the LAN Server for small transfers. This is usually the method used by the disk file systems as well.”

The Pin Read Hits Percent graph in Figure 4 – 3 indicates that during both layer four attacks, TCP and UDP, the request for disk reads were about the same. The disk read request is increased during both TCP attack and UDP attack as the intensity of the attacks increments. The red bar is network simulated traffic of TCP attack and the yellow bar is simulated traffic of UDP attack. The simulated attack does not come into play until the next set of bars at 10% of the attack where we see the highest number of Pin Read Hits %. The Pin Read Hits % then slowly decreased until reaching 75%

4.3.2 Logical Disk

At first, it was unclear how much disk time DDOS attacks were taking -- we thought it would be minimum since in previous papers it was hardly mentioned. However, when we decided to isolate some counters related to disk, we found some interesting results.

> "The Logical Disk performance object consists of counters that monitor logical partitions of a hard or fixed disk drives. Performance Monitor identifies logical disks by their a drive letter, such as C."

In Logical Disk performance object, we focused on % Idle Time, Disk Bytes/sec, Disk Write Bytes/sec and Disk Writes/sec. Each counter has its definition from Microsoft in quotation marks and our explanation of the results below its graph. Here we focus on % Idle Time, Disk Bytes/sec, Disk Write Bytes/sec and Disk Writes/sec counters to get a better understanding of how the Logical Disk performance is affected by TCP and UDP attacks.

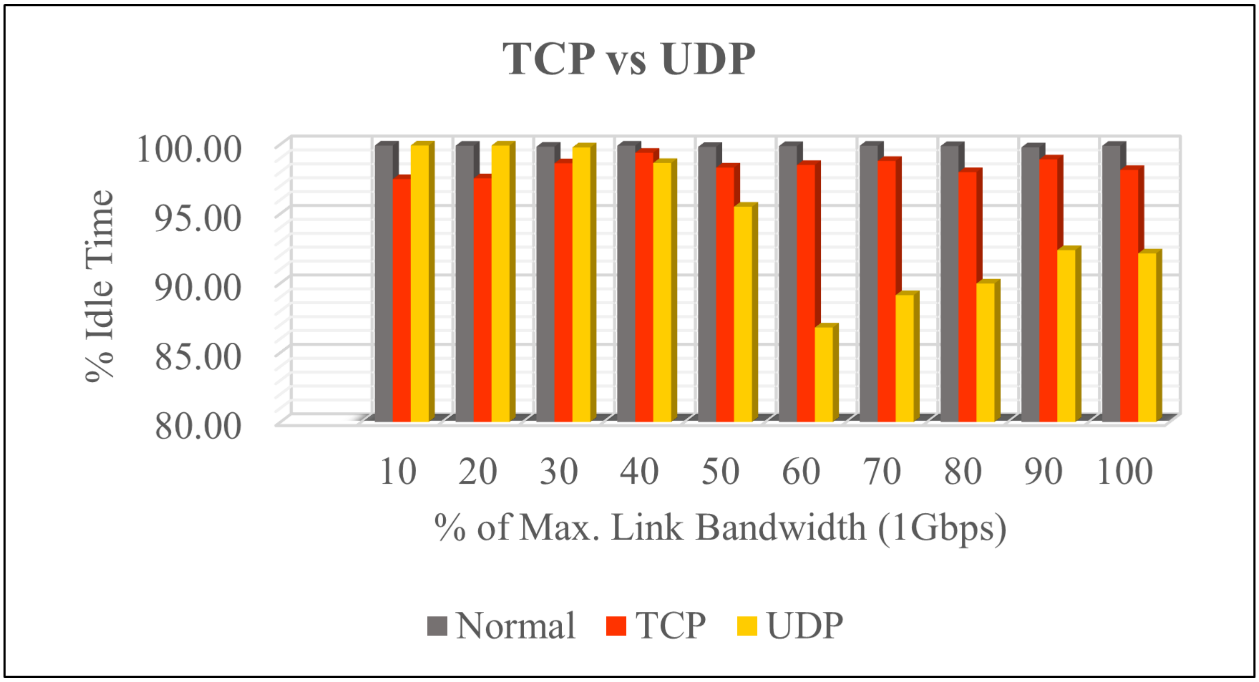


Figure 4 - 4 Logical Disk: Percentage Idle Time

> "% Idle Time reports the percentage of time during the sample interval that the disk was idle."

Figure 4 – 4 shows hardly any activity for Logical Disk before the cyberattacks, indicated by the gray-shaded bars marked Normal traffic, which represents the computer activity caused by legitimate traffic alone. During TCP attack the Logical Disk stayed idle about 98% of the time. And during the UDP attack, the logical disk activity started increasing at 40% of the attack intensity and reached 13% of activity before slowly decreasing the activity and reaching 8% of activity for 90-100% for the 1Gbps Link Bandwidth intensity. It would have been interesting to see what would have happened had we tested continuously for another hour at 100% intensity. Would the activity pick up or would it decrease to reach zero again?

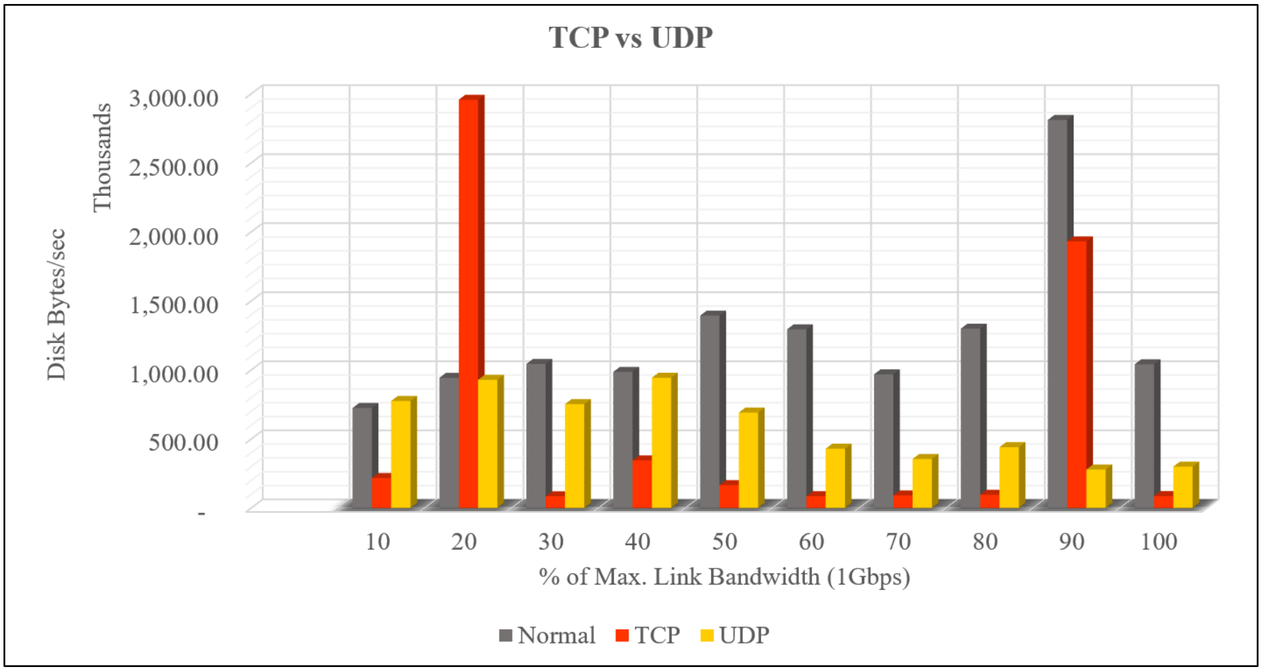


Figure 4 - 5 Logical Disk: Disk Bytes per second

“Disk Bytes/sec is the rate bytes are transferred to or from the disk during write or read operations.”

The graph, Disk Bytes per second in Figure 4 -5 does not show a pattern we can follow for the TCP attack. For UDP Attack we can see an immediate rise in the number of bytes being transferred back and forth, when the attack began at 10% of the bandwidth intensity, reaching a maximum close to one thousand bytes. Then, there is a steady decline in the migration of bytes, reaching a minimum at 90% of the bandwidth intensity with around 250 bytes per second. This counter could be studied farther. What happens if the bandwidth continues to increase in intensity to 200%? Would the bytes transferred per second increase, or would they stay below 500 disk bytes per second? On the other hand, we can see that for test when there was no traffic there was a tendency for bytes to increase. This may indicate that there is more communication with the disk when there is no attack than when there is one happening. This could be communication interference and could be due to saturation of processor resources.

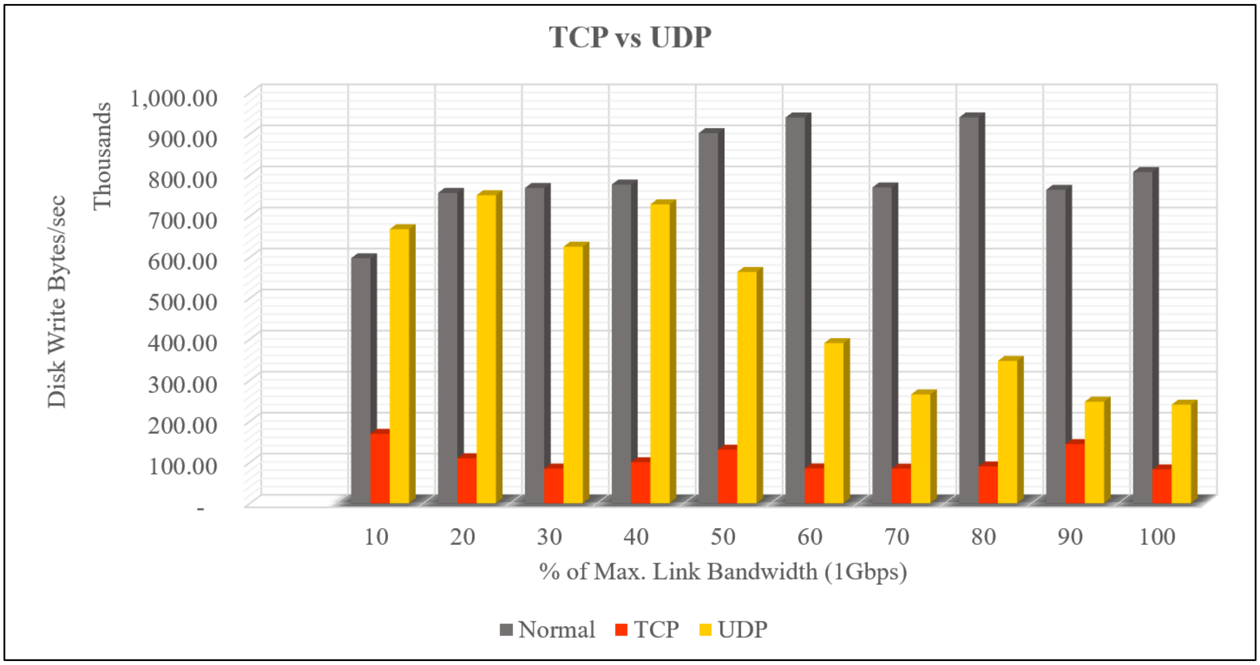


Figure 4 - 6 Logical Disk: Disk Write Bytes per second

“Disk Write Bytes/sec is rate at which bytes are transferred to the disk during write operations.”

In Figure 5-6 we can see that when a TCP attack is happening disk write operations are drastically affected, more so than during a UDP attack. We can see this by comparing these graphs to the gray graph, which represents the Disk Write Bytes/sec activity while the normal network activity is happening. The UDP attack did not affect until the attack intensity reached

30%. From there on, the write bytes per second operations were limited to 250 write bytes per second, on average. From this graph we can conclude that TCP attack affects the disk write operations more than a UDP attack.

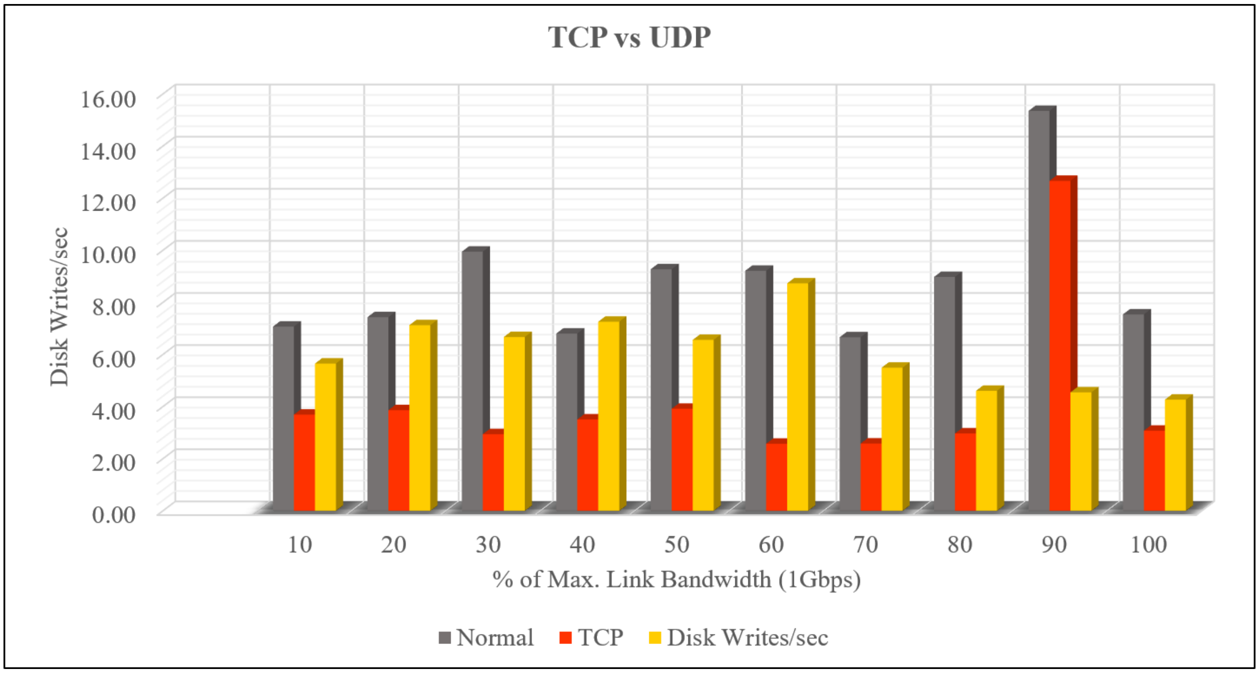


Figure 4 - 7 Logical Disk: Disk Writes per second

“Disk Writes/sec is the rate of write operations on the disk.”

Figure 4-7 is the Disk Writes per second that happen during TCP and UDP attack, compared to Normal traffic Writes per second, while not attack is going on. Here we can see that even thou the traffic generated by UDP attack is greater than that of TCP attack the difference is not much. The only noticeable pattern is that the TCP attack traffic was steadily under four writes per second throughout and the pronounced spike at 90% of the attack intensity that concurs with what happened during the Normal traffic. It would be interesting to analyze that more closely.

4.3.3 Memory

"The Memory performance object consists of counters that describe the behavior of physical and virtual memory on the computer. Physical memory is the amount of random access memory on the computer. Virtual memory consists of the space in physical memory and on disk. Many of the memory counters monitor paging, which is the movement of pages of code and data between disk and physical memory. Excessive paging, a symptom of a memory shortage, can cause delays which interfere with all system processes."

In Memory performance object we focus on % Committed Bytes In Use, Available Mbytes, and Page Faults/sec, and Pin Read Hits %. Each counter has its definition from Microsoft in quotation marks and our explanation of the results below its graph.

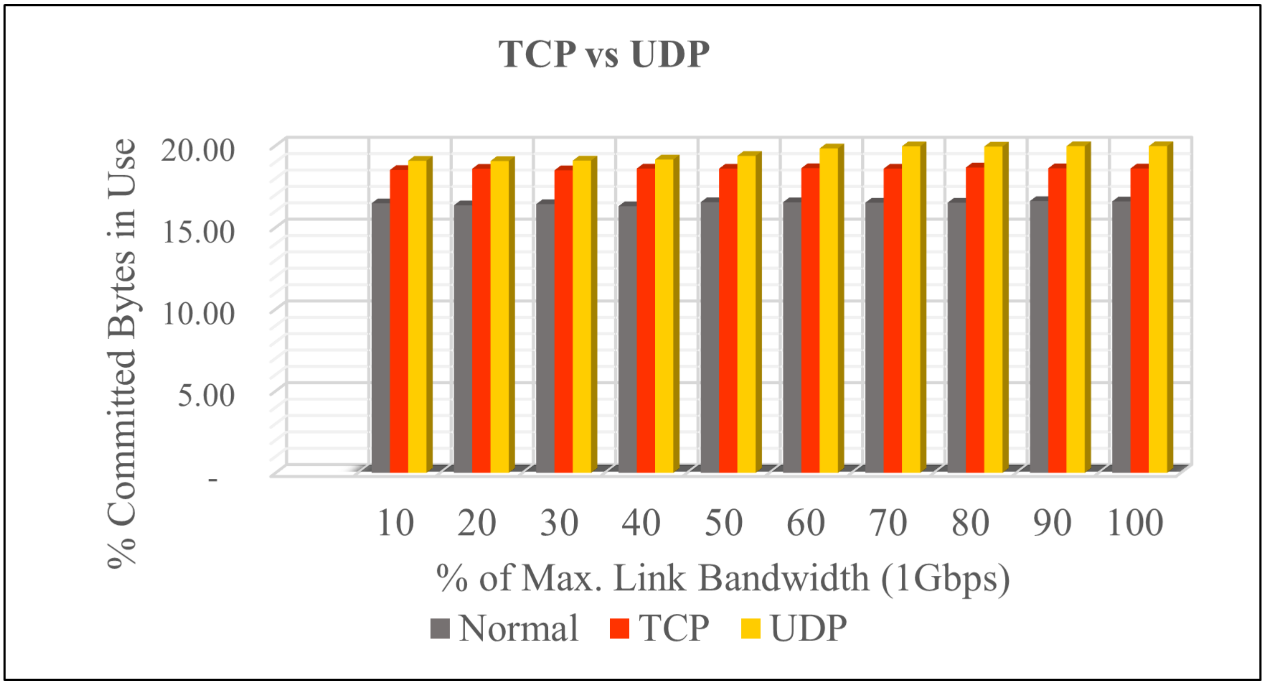


Figure 4 - 8 Memory: Percent Committed Bytes In Use

"% Committed Bytes In Use is the ratio of Memory\\Committed Bytes to the Memory\\Commit Limit. Committed memory is the physical memory in use for which space has been reserved in the paging file should it need to be written to disk. The commit limit is determined by the size of the paging file. If the paging file is enlarged, the commit limit increases, and the ratio is reduced). This counter displays the current percentage value only; it is not an average."

Something interesting about Figure 4-8 is that the graph shows 16% committed bytes while computer is running on idle and during normal traffic. This happens for both legitimate traffic, starting at 10% of the maximum link bandwidth, and persists until the test ends. Comparing the normal result against TCP attack and UDP attack we notice that there are more bytes from Memory committed during the UDP attack. This is evidence that these two forms of cyberattack affect memory of the computer. This comparison also helps us determine that a UDP attack does use more memory than a TCP attack.

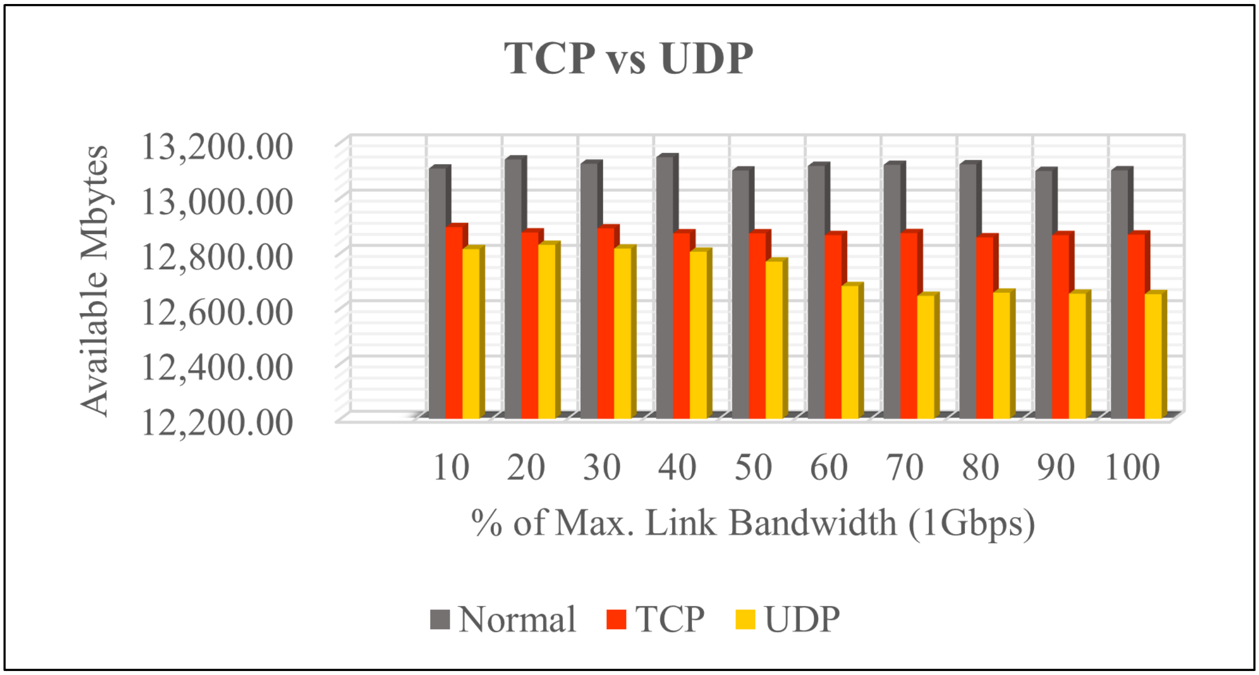


Figure 4 - 9 Memory: Available Megabytes in Memory

"Available MBytes is the amount of physical memory, in Megabytes, immediately available for allocation to a process or for system use. It is equal to the sum of memory assigned to the standby (cached), free and zero page lists."

Figure 4-9 above show the Megabytes available. Our victim computer has 16 Gigabytes of memory. Knowing that, we can deduct from this graph that the megabytes occupied during a UDP attack increases as the attack intensifies during both TCP and UDP attacks. For TCP attack the difference is not much. However, if we compare it to the base memory occupied during normal traffic, we can see an increment in memory usage during TCP attack. Also, for UDP attack, we can see the increase of megabytes after 40% of the attack intensity and continues to increase until 70% of the attack intensity of the 1 Gbps max link bandwidth. This is another clear evidence of the effect these cyberattack have on main memory. UDP attack, again, affects memory more than TCP attack.

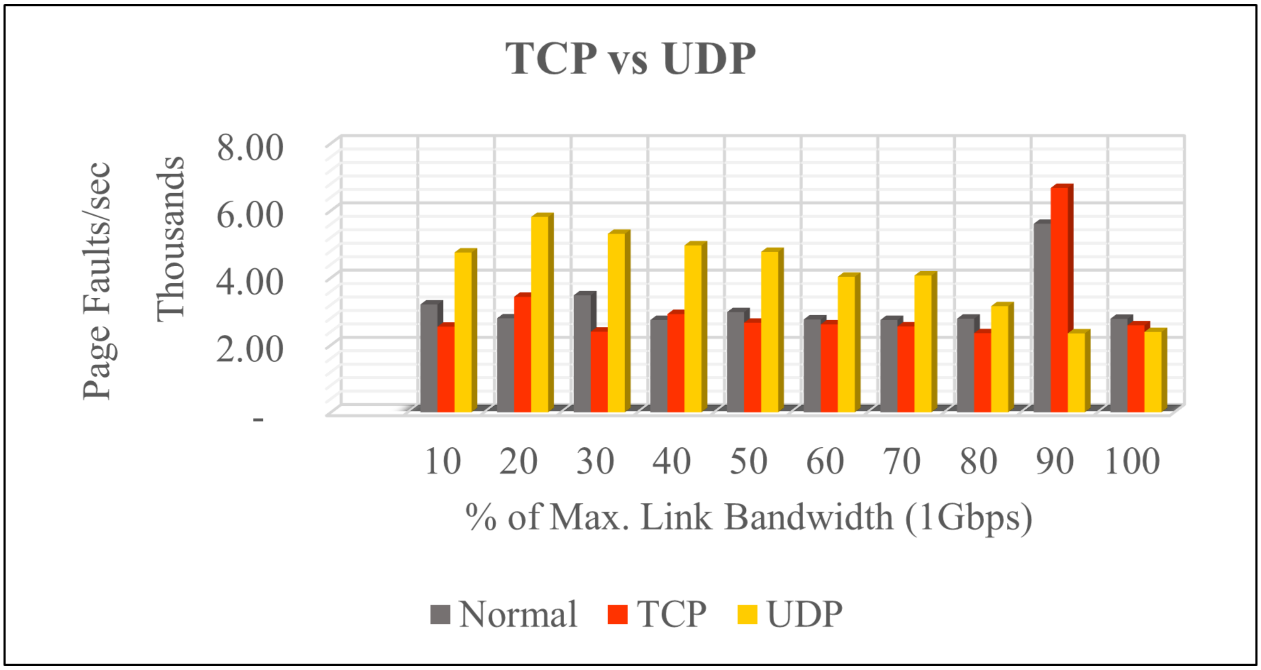


Figure 4 - 10 Memory: Page Faults per second

“Page Faults/sec is the average number of pages faulted per second. It is measured in number of pages faulted per second because only one page is faulted in each fault operation, hence this is also equal to the number of page fault operations. This counter includes both hard faults (those that require disk access) and soft faults (where the faulted page is found elsewhere in physical memory.) Most processors can handle large numbers of soft faults without significant consequence. However, hard faults, which require disk access, can cause significant delays."

In the graph above, Figure 4-10, we can observe the amount of page faults generated during normal traffic compared to traffic while a UDP and TCP attack are happening. Page faults happened during normal traffic on average of three thousand Page Faults per second. During a TCP attack the average of page faults declined but during the UDP attack the average of page faults per second rose. We can see in the graph this significant rise starting at 10% of the attack intensity. UDP attack almost reached six thousand Page Faults per second at 20% of the attack intensity, after which the decline start until reaching a stability at 90% of the attack intensity.

4.3.4 Physical Disk

In Physical Disk performance object, we focused on % Disk Time, % Disk Write Time, and % Idle Time. We present the definition as provided by Microsoft in quotation marks followed by our explanation of the results below its graph.

"The Physical Disk performance object consists of counters that monitor hard or fixed disk drive on a computer. Disks are used to store file, program, and paging data and are read to retrieve these items, and written to record changes to them. The values of physical disk counters are sums of the values of the logical disks (or partitions) into which they are divided."

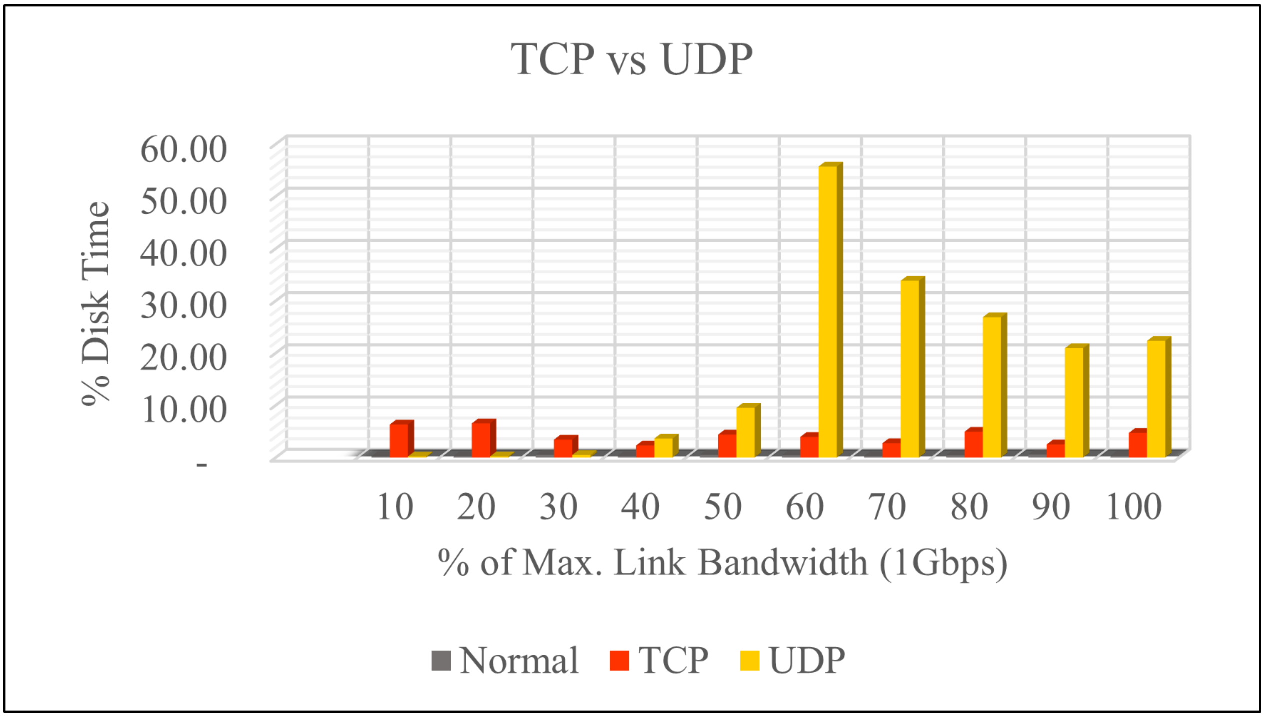


Figure 4 - 11 Physical Disk: Percent Disk Time

"% Disk Time is the percentage of elapsed time that the selected disk drive was busy servicing read or write requests."

Percent Disk Time graph in Figure 4-11 contains the % Disk Time counter of Physical Disk three times. The first counter is the base, where the computer ran on its own with simulated legitimate traffic. This graph can barely be recognized due to the low percentage of time it was busy; however, we can see a small difference during the TCP attack and even a more pronounced result during UDP attack.

During TCP Attack we can see the disk was busy up to 5% of the time throughout the duration of the attack. Throughout most of the other percent intensities of the attack the activity of the disk time remained way below the 10% mark. During the UDP attack, however, we can notice the activity of the disk increasing slowly from 0% to 50% of the activity. The disk time drastically jumps up to 55% in activity at 60% of the attack intensity. It then has a pronounced decline in activity until reaching 90% of the attack intensity where we see that it stabilizes. It would be interesting to see more results of this counter.

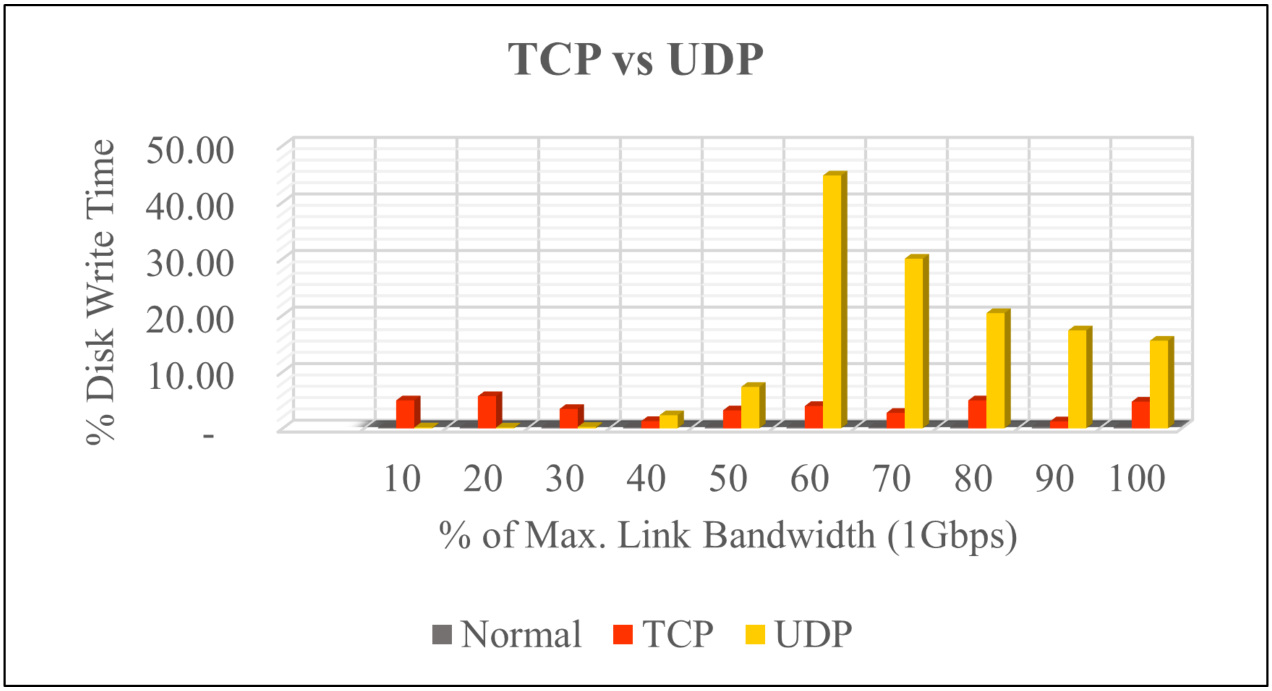


Figure 4 - 12 Physical Disk: Percent Disk Write Time

"% Disk Write Time is the percentage of elapsed time that the selected disk drive was busy servicing write requests."

Comparing Figure 4-12 to Figure 4-11 we can deduce that most of the time the disk was busy it was due to disk writes. As we did in Figure 4-11, we can see a very low percent of percent disk write time for the normal traffic in Figure 4-12 as well. We also see the percent disk write time rise to no more than 5% during the TCP attack. However, during the UDP attack the percent disk write time increased starting at 40% of the attack intensity reaching 44% disk write time at 60% of the attack intensity before declining to 15% disk write time when the attack intensity reached 100%. We can deduce that the decrement of the percent disk write time will stabilize at about 15% disk write time even if the attack continues to increase but more testing that involve letting the test run longer or doing more observations of higher traffic intensities.

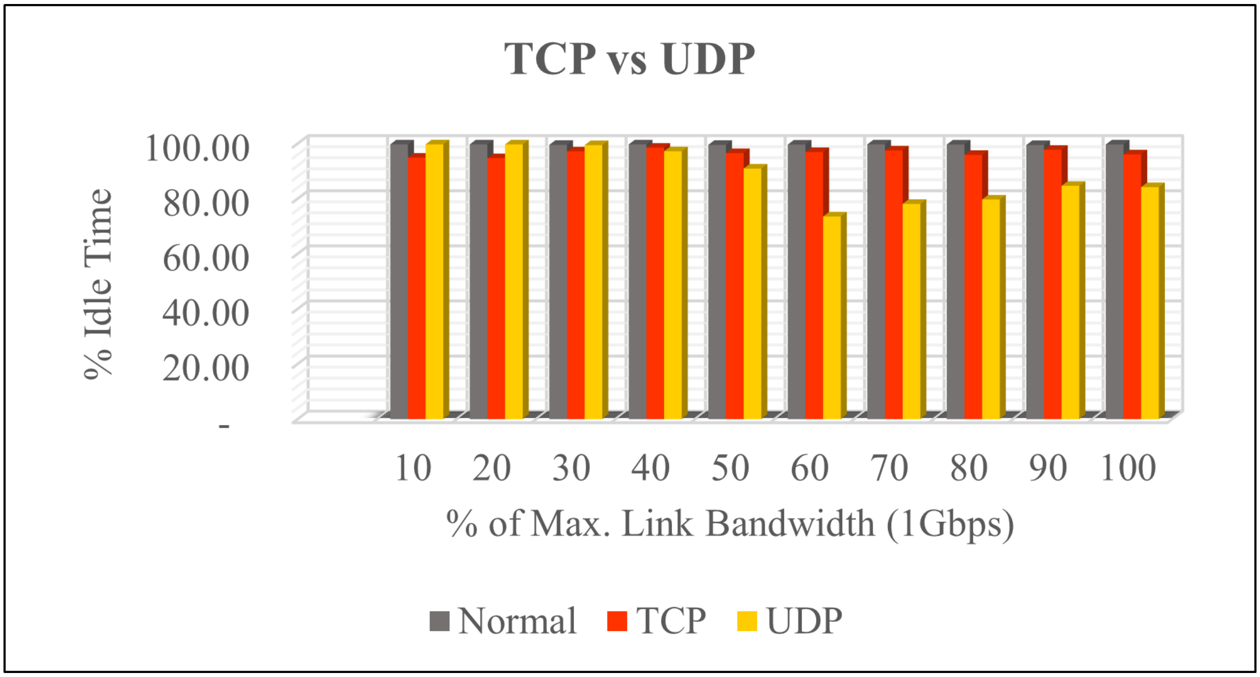


Figure 4 - 13 Physical Disk: Percent Idle Time

"% Idle Time reports the percentage of time during the sample interval that the disk was idle."

In Figure 4-13 we see the differences in the activity of Physical Disk. This graph shows that Physical Disk was active at most 30%. This agrees with the graphs in Figure 4-12 and Figure 4-11 related to Physical Disk.

4.3.5 Process

> "The Process performance object consists of counters that monitor running application program and system processes. All the threads in a process share the same address space and have access to the same data."

In Process performance object, we focused on % Privileged Time, % Processor Time, % User Time, IO Write Operations/sec and Page Faults/sec. We present the definition for each counter, as provided by Microsoft, in quotation marks followed by our explanation of the results below its graph.

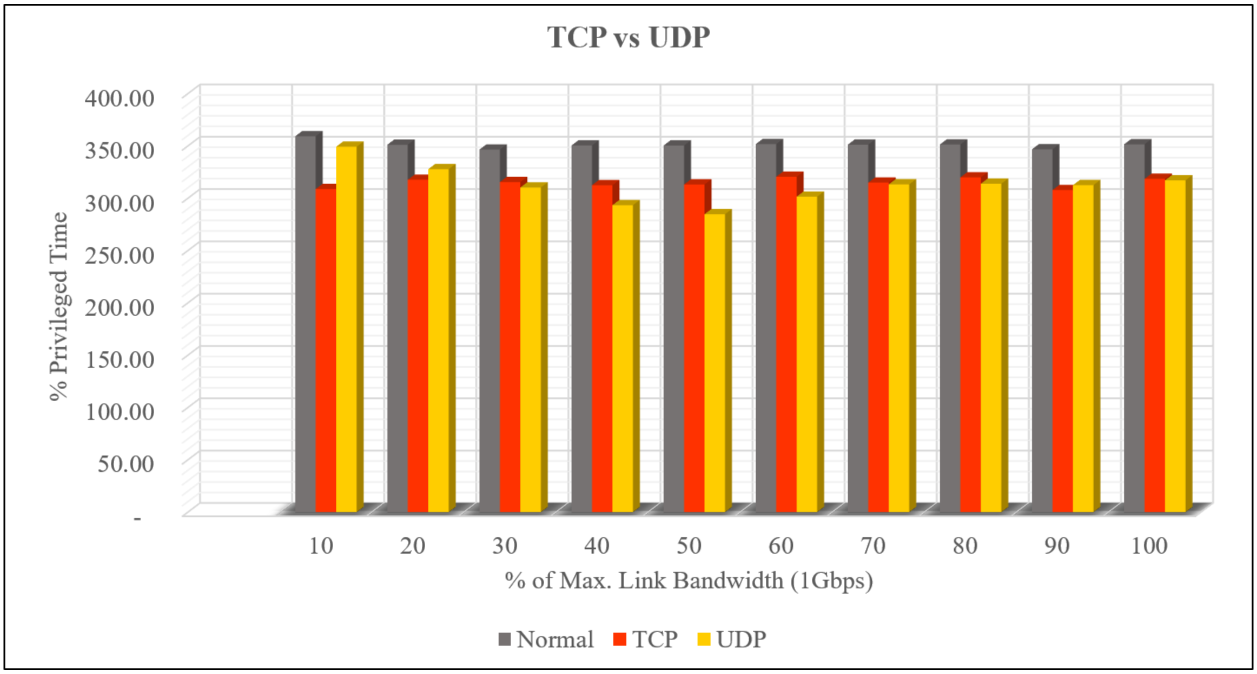


Figure 4 - 14 Process: Percent Privileged Time

"% Privileged Time is the percentage of elapsed time that the process threads spent executing code in privileged mode. When a Windows system service is called, the service will often run in privileged mode to gain access to system-private data. Such data is protected from access by threads executing in user mode. Calls to the system can be explicit or implicit, such as page faults or interrupts. Unlike some early operating systems, Windows uses process boundaries for subsystem protection in addition to the traditional protection of user and privileged modes. Some work done by Windows on behalf of the application might appear in other subsystem processes in addition to the privileged time in the process."

Figure 4-14 is the % Privileged Time. These results are somewhat puzzling as logical measurements should be limited to 100% but as we can see the limits are way above. This could be due to the overclocking capabilities of the processor. Several more tests could reveal some better conclusion, perhaps.

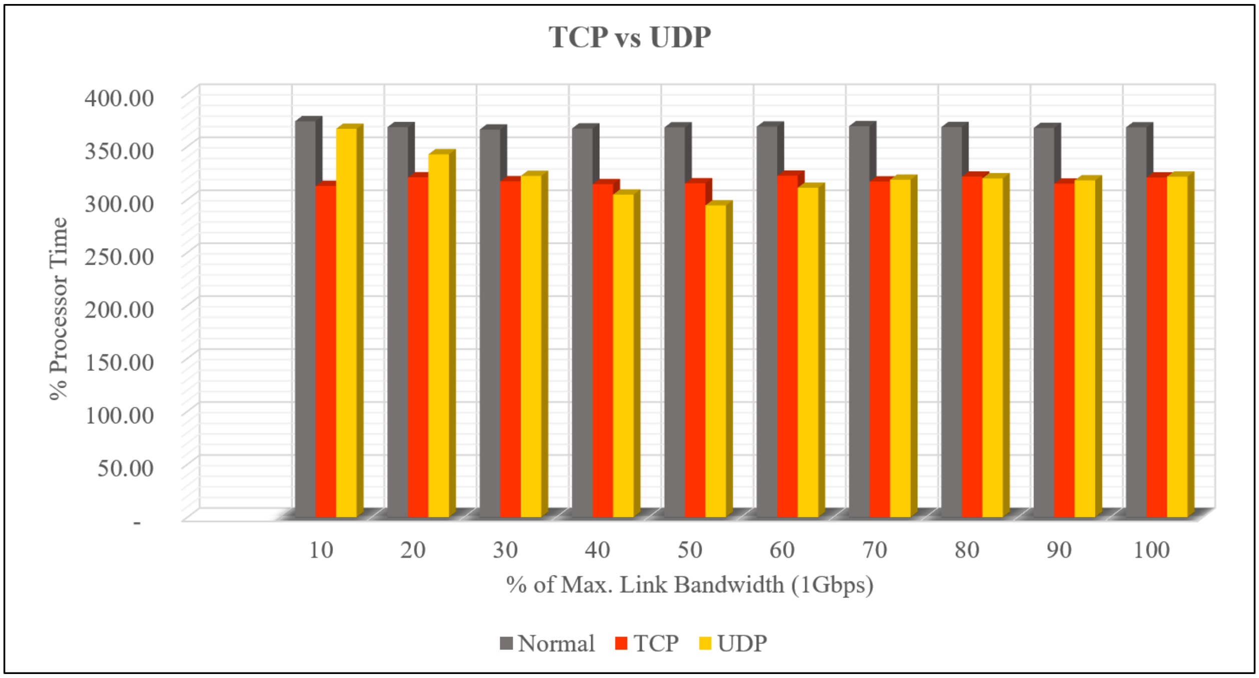


Figure 4 - 15 Process: Percent Processor Time

"% Processor Time is the percentage of elapsed time that all of process threads used the processor to execution instructions. An instruction is the basic unit of execution in a computer, a thread is the object that executes instructions, and a process is the object created when a program is run. Code executed to handle some hardware interrupts and trap conditions are included in this count."

% Processor Time graph above in Figure 4-15 has matching results to Figure 4-14. We can see that for both graphs above there is a similar situation where both graphs exceed the 100%. The question here is, after the 100% is reached, where does the other added percentage come from? These are similar results as the ones found for the Layer 3.5 attacks.

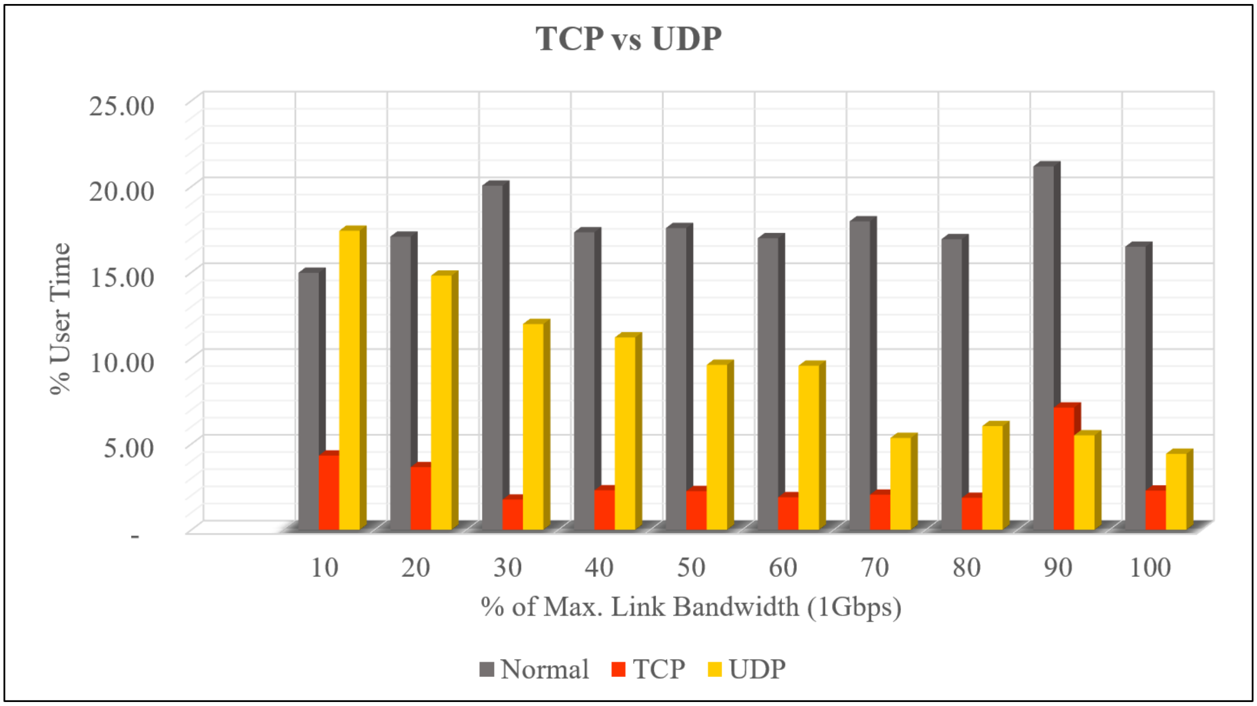


Figure 4 - 16 Process: Percent User Time

"% User Time is the percentage of elapsed time that the process threads spent executing code in user mode. Applications, environment subsystems, and integral subsystems execute in user mode. Code executing in user mode cannot damage the integrity of the Windows executive, kernel, and device drivers. Unlike some early operating systems, Windows uses process boundaries for subsystem protection in addition to the traditional protection of user and privileged modes. Some work done by Windows on behalf of the application might appear in other subsystem processes in addition to the privileged time in the process."

In Figure 4-16 we can see that the Percentage User Time when the computer receives normal traffic reaches 20%. We can tell by the graph that Layer 4 attacks suppress % Usage Time. This is clearly indicated by the prompt decline in user percentage at the 10% attack intensity during TCP attack. For UDP attack, the decline in user time is not as severe but a decline is observed, none the less.

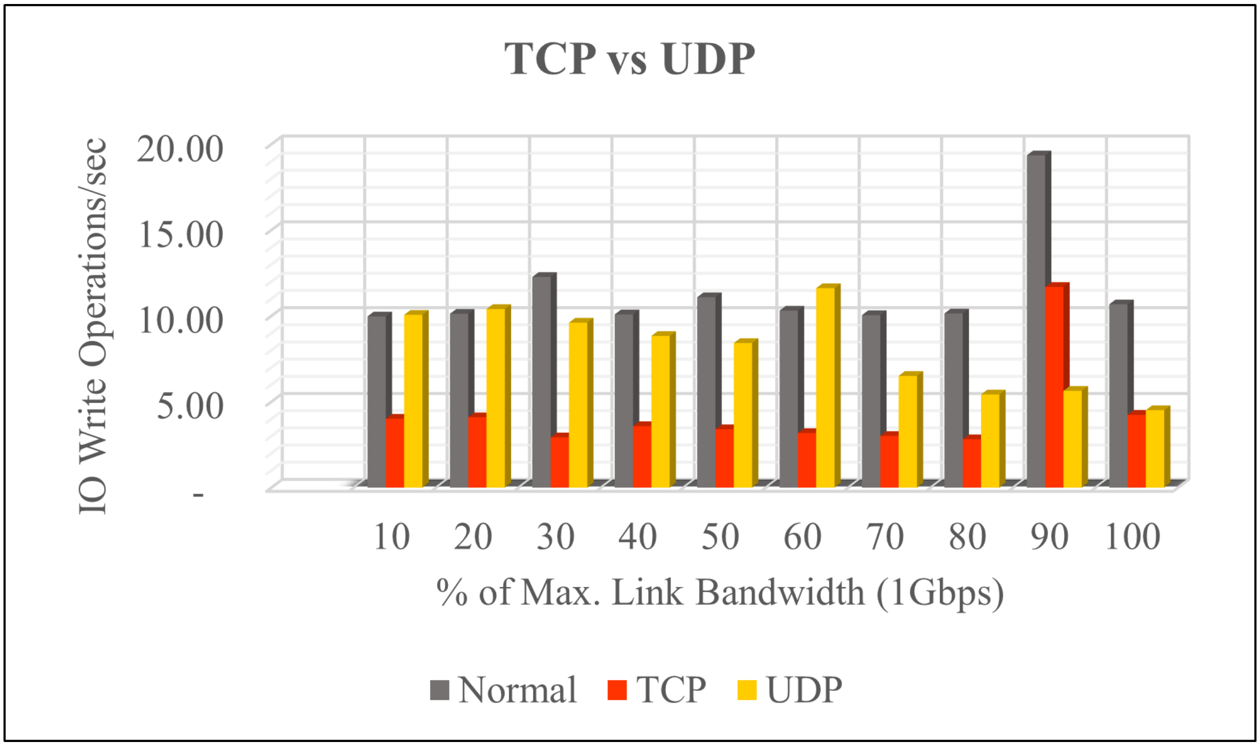


Figure 4 - 17 Process: Input/Output Write Operations per second

> "The rate at which the process is issuing write I/O operations. This counter counts all I/O activity generated by the process to include file, network and device I/Os."

Figure 4-17 presents the process activity from file, network and input/output devices. This graph shows that there is more activity when the computer is receiving not traffic than when Internet traffic is applied. A TCP attack suppresses the input/output activity more than a UDP attack as less than half the activity of UDP is observed.

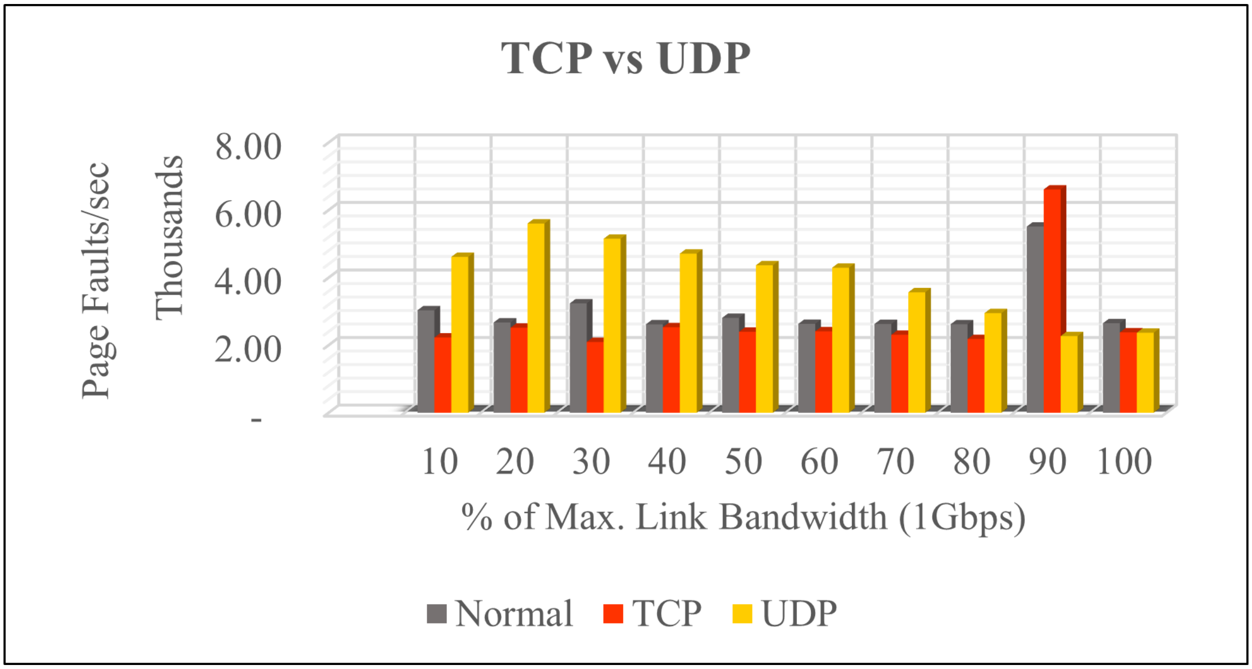


Figure 4 - 18 Process: Page Faults per second

> "Page Faults/sec is the rate at which page faults by the threads executing in this process are occurring. A page fault occurs when a thread refers to a virtual memory page that is not in its working set in main memory. This may not cause the page to be fetched from disk if it is on the standby list and hence already in main memory, or if it is in use by another process with whom the page is shared."

Figure 4-18 shows the Page Faults that occur per second when a TCP and a UDP attack is happening. The activity is not as pronounced in this Layer 4 attack as it is in the results for Layer 3.5 attack presented in chapter 3. Here, something to notices the activity of the UDP attack. As shown in the graph, the page faults increase once the attack begins. Then it decreases to the level of Normal Internet activity. Based on the description provided by Microsoft this could be due to the computer being able to find the pages needed in virtual memory.

4.3.6 Processor

“The Processor performance object consists of counters that measure aspects of processor activity. The processor is the part of the computer that performs arithmetic and logical computations, initiates operations on peripherals, and runs the threads of processes. A computer can have multiple processors. The processor object represents each processor as an instance of the object.”

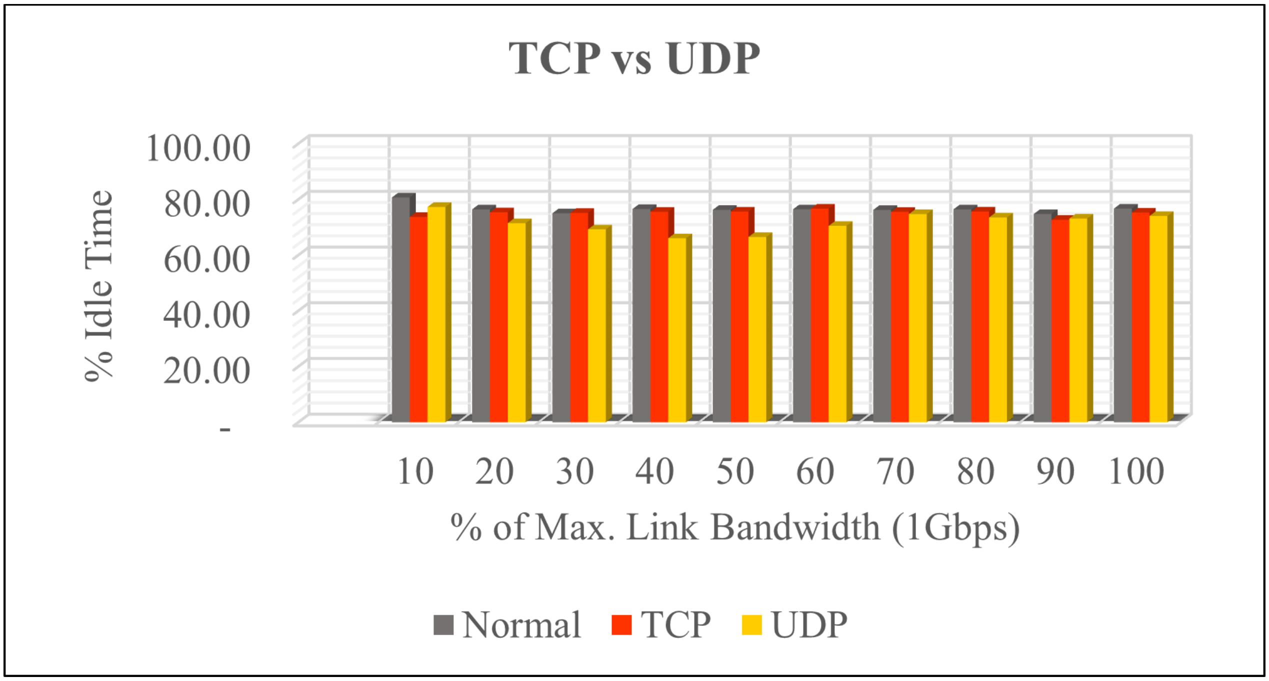


Figure 4 - 19 Processor: Percent Idle Time

“% Idle Time is the percentage of time the processor is idle during the sample interval”

In Figure 4-19 there are not many noticeable differences from attack to attack. On the 0% intensity both Normal and UDP attack displayed a similar idle time percentage. UDP idleness decreased as the intensity of the traffic increased, but the percentage kept about the same for these two counters. There was a slight difference for the TCP counter. We can see TCP attack’s % Idle Time kept a steady percentage throughout the test. From this graph we can say that the processor activity is grater during a UDP attack and that a UDP attack affects a computer processor more than during a TCP attack.

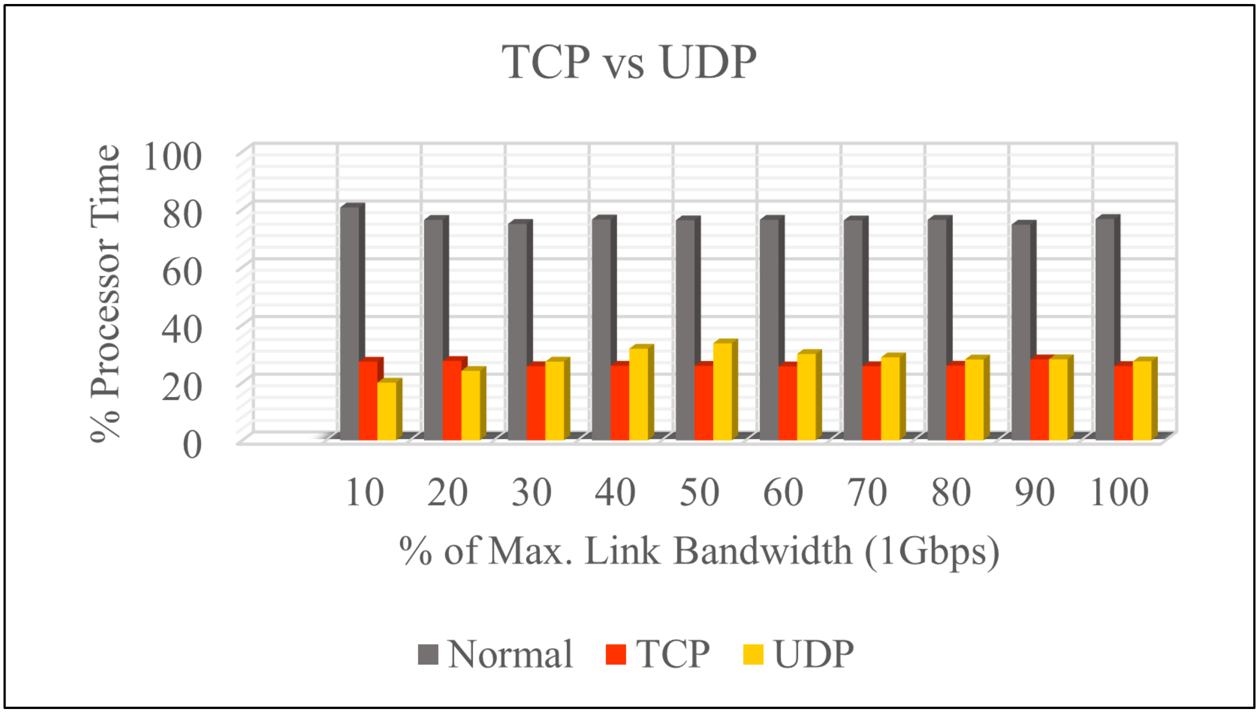


Figure 4 - 20 Processor: Percentage Processor Time

"% Processor Time is the percentage of elapsed time that the processor spends to execute a non-Idle thread. It is calculated by measuring the percentage of time that the processor spends executing the idle thread and then subtracting that value from 100%. (Each processor has an idle thread that consumes cycles when no other threads are ready to run). This counter is the primary indicator of processor activity, and displays the average percentage of busy time observed during the sample interval. It should be noted that the accounting calculation of whether the processor is idle is performed at an internal sampling interval of the system clock (10ms). On todays fast processors, % Processor Time can therefore underestimate the processor utilization as the processor may be spending a lot of time servicing threads between the system clock sampling interval. Workload based timer applications are one example of applications which are more likely to be measured inaccurately as timers are signaled just after the sample is taken."

Percentage Processor Time in Figure 4-20 show how the processor time get reduced when a TCP attack and UDP attack reached the computer device. Compared to the Normal traffic, the processor spent less time executing during the attacks than when there was no attack.

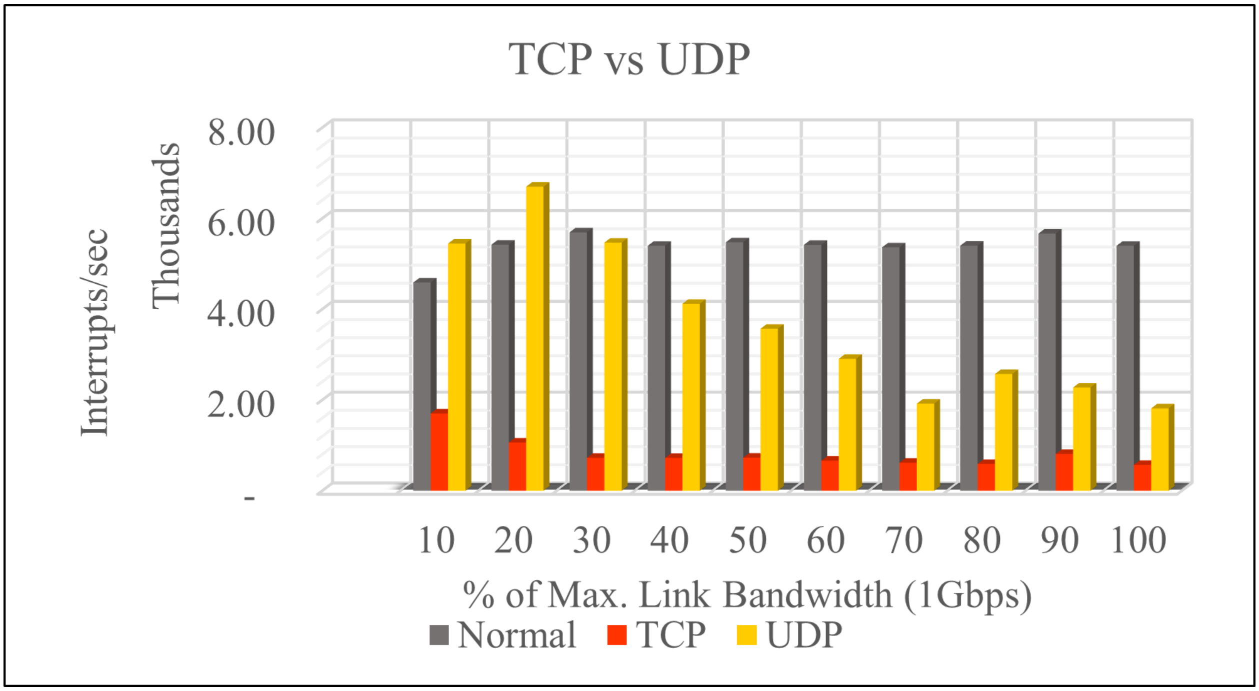


Figure 4 - 21 Processor: Interrupts per second

"Interrupts/sec is the average rate, in incidents per second, at which the processor received and serviced hardware interrupts. It does not include deferred procedure calls (DPCs), which are counted separately. This value is an indirect indicator of the activity of devices that generate interrupts, such as the system clock, the mouse, disk drivers, data communication lines, network interface cards, and other peripheral devices. These devices normally interrupt the processor when they have completed a task or require attention. Normal thread execution is suspended. The system clock typically interrupts the processor every 10 milliseconds, creating a background of interrupt activity. This counter displays the difference between the values observed in the last two samples, divided by the duration of the sample interval."

The graph in Figure 4-21 we can see the number of Interrupts per second. According to this graph, UDP attack generates the most interrupts per second than TCP attack. We can also observe that the Interrupts happen when there is Internet activity, as it's observe by the gray bars. After that we see the interrupts activity increases to four thousand and then peaks to above five thousand with Internet activity. We can see Interrupts activity at the beginning of each attack, but that activity declines for both, at different periods. For UDP attack the number of Interrupts peaks at twenty percent of the attack intensity. Then it slowly declines to below the two thousand mark. This activity is lower than that generated by Normal traffic. Interrupt activity for TCP attack shows a more pronounced decline. It only rises to twenty-five hundred Interrupts per second and from there the number of Interrupts keep declining until the end of the test. However, the difference from intensity to intensity is not too noticeable for TCP attack.

4.3.7 System

System performance object is analyzed, in this paper, using the following counters: % Registry Quota In Use, File Write Bytes/sec, and File Write Operations/sec and System Calls/sec.

"The System performance object consists of counters that apply to more than one instance of a component processors on the computer."

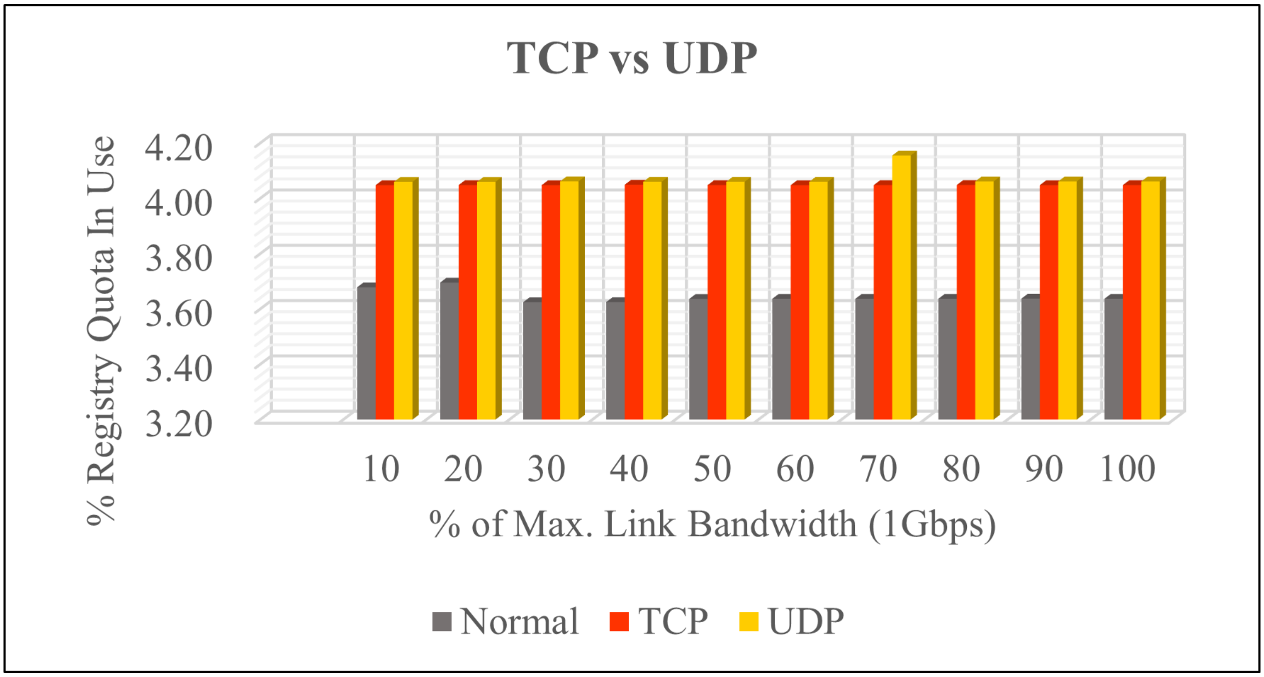


Figure 4 - 22 System: Percentage Registry Quota In Use

"% Registry Quota In Use is the percentage of the Total Registry Quota Allowed that is currently being used by the system. This counter displays the current percentage value only; it is not an average."

The registry, as explained in the book Internet Information Services (IIS) 6.0 by Microsoft Press, holds all type of information related to the computer, from what type of fonts the system uses to decryption classes, and component information that make the computer [13]. The description of this counter says it is not 'an average' so we took samples every two seconds. Then, we averaged those samples to get this graph averaging those them. Here we see the % Registry Quota in Use stays consistent throughout most of attacks intensities. The difference between the effects of UDP versus TCP is minimum. At 70% of the maximum link bandwidth of 1 Gbps the % Registry Quota In Use we found a very defined difference where UDP peaks up some 0.10%. We do see, however, that the registry is affected by TCP and UDP attacks by about 1.0%.

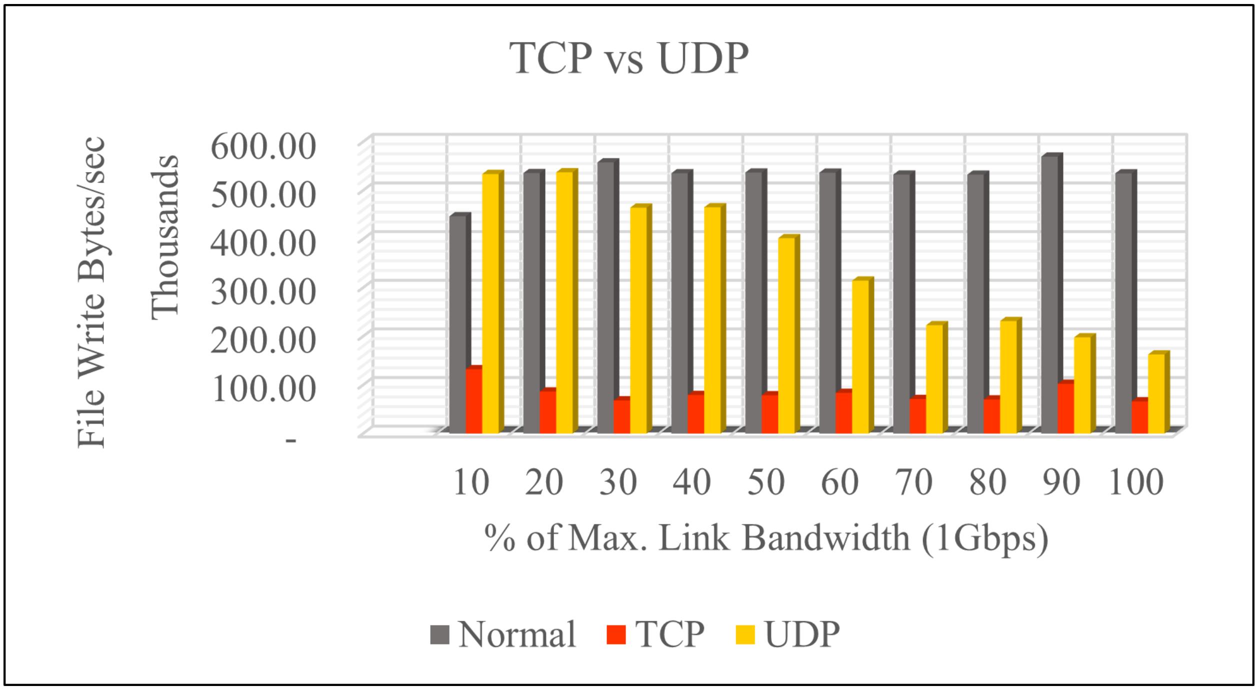


Figure 4 - 23 System: File Write Bytes per second

"File Write Bytes/sec is the overall rate at which bytes are written to satisfy file system write requests to all devices on the computer, including writes to the file system cache. It is measured in number of bytes per second. This counter displays the difference between the values observed in the last two samples, divided by the duration of the sample interval."

Figure 4-23 shows how File Write Bytes are suppressed by TCP and UDP attacks. This is evident by observing and comparing the gray bars, the Normal Internet traffic to the red bars, the TCP attack and UDP attack traffic. We can see how, for zero percent of the Maximum Link Bandwidth, File Write Bytes per second reaches the same level as that for Normal traffic, which is Internet traffic. Then, declines once TCP attack is introduced. We observe the same scenario for UDP attack but much later, when the attack intensity reaches thirty percent. Then again at seventy percent until the one hundred percent attack intensity is reached.

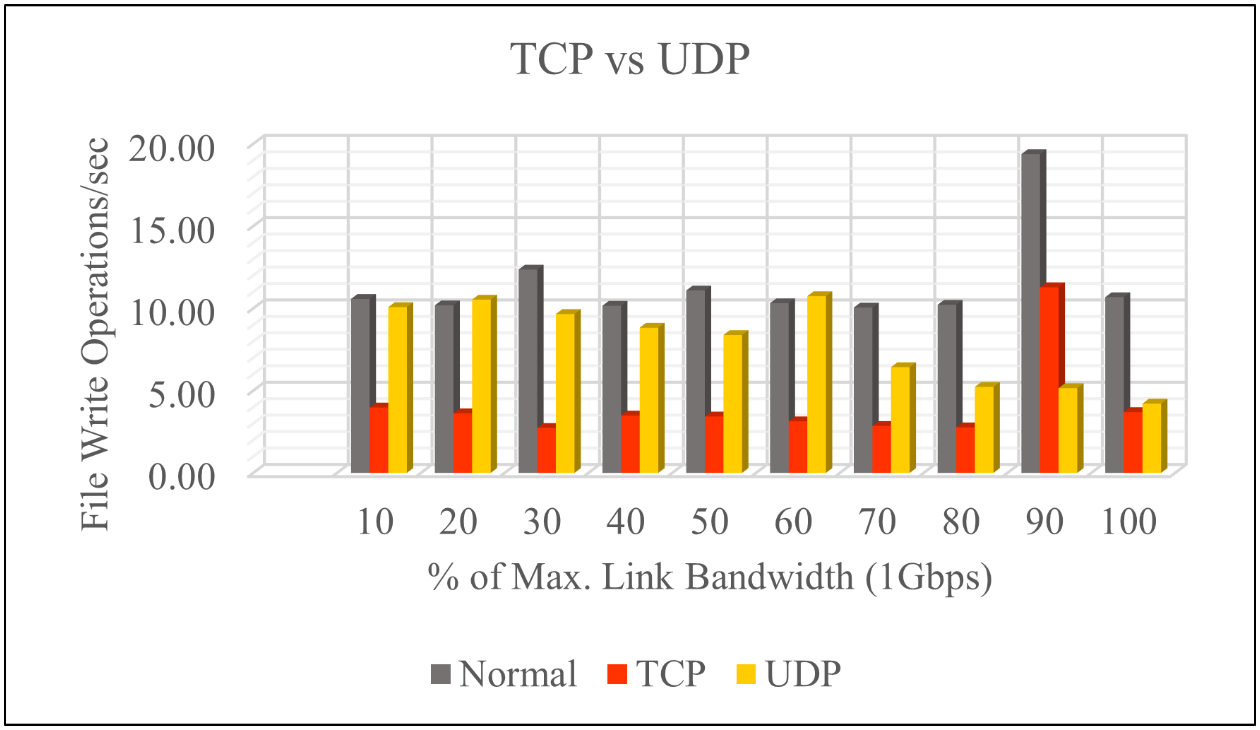


Figure 4 - 24 System: File Write Operations per second

> “File Write Operations/sec is the combined rate of the file system write requests to all devices on the computer, including requests to write to data in the file system cache. It is measured in numbers of writes. This counter displays the difference between the values observed in the last two samples, divided by the duration of the sample interval.”

From Figure 4-24 we can observe that the File Write Operations per second is not very high for neither of the attacks. If we look at the effect displayed by Normal traffic, which is when the computer is receiving legitimate traffic, we can determine that there is a suppression of this type of activity during Internet activity and much more during UDP and TCP attacks.

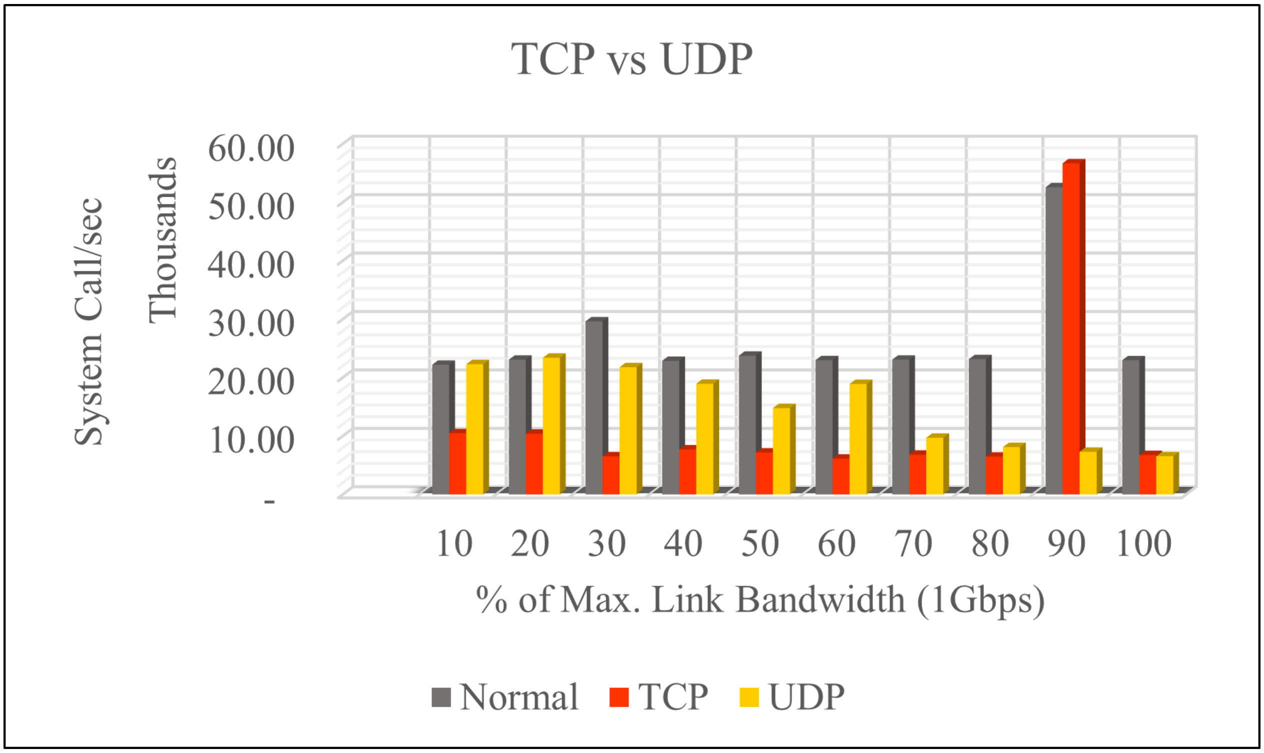


Figure 4 - 25 System: Calls pers second

> “System Calls/sec is the combined rate of calls to operating system service routines by all processes running on the computer. These routines perform all of the basic scheduling and synchronization of activities on the computer, and provide access to non-graphic devices, memory management, and name space management. This counter displays the difference between the values observed in the last two samples, divided by the duration of the sample interval.”

System calls per second in Figure 4-25 show how UDP attack causes activity of the operating system. Here we can observe that the Normal traffic keeps a steady relationship with the operating system, but when a UDP attack or TCP attack are incorporated that relationship gets affected. In a TCP attack the interaction with the operating system is limited to ten thousand call per second and it is decreased as the intensity of the attack is increased. For a UDP attack this is not affected as much at the beginning, but we do see a decline in the interaction of the processing resources with operating system.

4.4 CONCLUDING REMARKS

In this chapter we presented an in-depth look at what happens inside a computer when a Layer 4 cyberattack is affecting it. As in chapter 3, here too, we observed seven computer performance objects to get a better understanding of what happens to the computer. Here, we focus on the counters that show activity that could give us a better understanding to compare the results to a Layer 4 type of attack. The results for a Layer 4 type of attack are presented in chapter 4. For Cache, we looked at three elements that gave us the most activity: Dirty Pages, Dirty page threshold and Pin Read Hits %. In Logic Disk we looked at % Idle Time, Disk Bytes/sec, Disk Write Bytes/sec and Disk Writes/sec. The Memory object is analyzed using % Committed Bytes In Use, Available Mbytes, and Page Faults/sec. In Physical Disk performance object we looked into % Disk Time, Disk Write Time, % Idle Time object counters. The Process performance object, in our test, is composed of the following performance counters: % Privileged Time, % Processor Time, % User Time, I/O Write Operations/sec, Page Faults/sec. % Idle Time, % Processor Time, and Interrupts/sec are counters selected to represent the activity of the Processor performance object. Lastly, System performance object is analyzed in the paper using the following counters: % Registry Quota In Use, File Write Bytes/sec, and File Write Operations/sec and System Calls/sec.

Our observation of the computer activity using the above-mentioned performance objects showed how a computer’s activity is affected when it is attack by commonly known DDoS attacks such as TCP attack and UDP attack. We conclude that a UDP attack is more damaging to a Microsoft Surface Pro 4 computer than a TCP attack in the Layer 4. In this chapter we compare TCP attack with UDP attack. Two commonly known cyberattacks in Layer 4 that affect networks around the world. In Chapter 5 we will compare the most damaging of this, Layer 4 attacks, to the most damaging attack from Layer 3 in chapter 3. In Chapter 5, will give some suggestions experts recommend for keeping a safe computer with mitigations tools and methods that can be implemented.

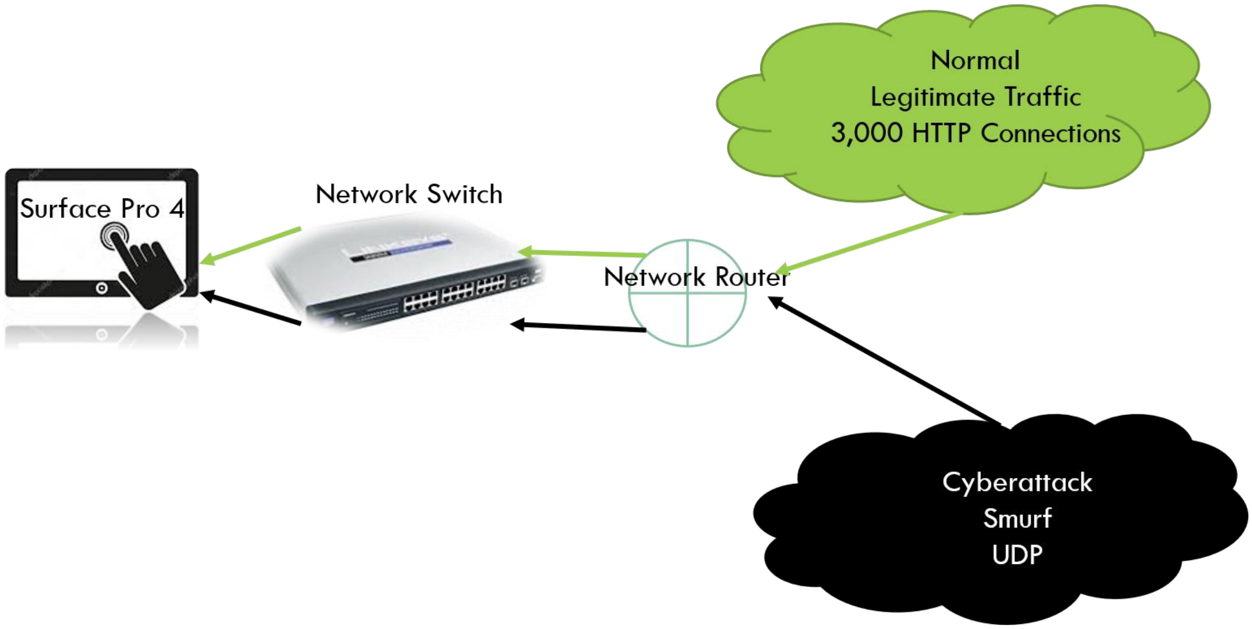
CHAPTER V

LAYER 3.5 VS LAYER 4 DDOS ATTACK RESULTS

5.1 Experimental Setup

In this chapter we cover the elements we used to perform our Layer 3.5 versus Layer 4 Cyberattack, testing the differences of the most damaging cyberattacks presenting in chapters 3 and 4. In the first section we include the hardware and software that was necessary to run our tests. Then, we explain our results in section two. Section two is divided into definition of the object, both definitions of the counters are in quotation marks as given by Microsoft, and the graph representing the results obtained for that part of our test. The graph is followed by our description of the results. In section three, we conclude with a synthesis of the experiment we talk about in this chapter.

In Schematic 5-1, we can see the setup of the experiment we ran to analyze the performance of SP4 under Smurf attack and UDP attack. Going from left to right, the two clouds represent the Internet. The green cloud represents legitimate user connections. These connections represent a simulation of 3,000 HTTP connections. The black cloud represents simulated cyberattack. This attack traffic is added to the legitimate traffic simulating what happens in a network daily. The traffic from these two cloud networks comes into the network through the router, which in this case is simulated and represented by the round figure with two lines going through the center. The Network Switch simply connects multiple devices in our network. The device on the far left represents the SP4 device we tested. This is how we test how well a computer or server performs under cyberattacks. Below is a list of the actual devices we used and some of their features.



Schematic 5 - 1: Smurf Attack and UDP Attack Experimental Setup

5.2 Hardware and Software tools

The tools we used in this project are the same the tools we used in chapters 3 and chapter 4. We used Microsoft Surface Pro 4 as our victim device. It is built with an Intel Core i7 processor, which runs at 2.2 GHz [4]. We used Ethernet-to-USB 3.0 adaptor from Freegene to transfer the traffic generated for our test on the SP4 as these devices are not equipped with an Ethernet port. Data was collected using Performance Monitor, a feature embedded in Windows operating systems to analyze the performance of many computer parameters, including Cache, Logic Disk, Memory, Physical Disk, Process, Processor and System. The data comes in a comma separated value (CSV) file, so we used Microsoft Office Excel to analyze our data and convert it to graphs that can be easier to read. We learned that CSV files are not very reliable when you need to make changes to it. Many times, the data was lost when we tried graphing the data directly from a CSV file. To not go through the same issue any more we converted the extracted CSV file to excel file. This work out better because we were able to create our graphs and we preserved the integrity of the original data. We worked from the excel file and kept the CSV file untouched for the remaining of the project.

5.3 Performance Objects of Interest

The computer parameters we use in our evaluation are presented in this chapter. We used ten different traffic intensities from 10% to 100% in our comparisons. We grouped these intensities, comparing the base traffic with the cyberattacks to see the distinction and effects the intensities of the attacks have on the computer. Base 10% to 100% represented in gray bars, we let the computer run with simulated legitimate traffic combined to capture a representation of it, to then compare it to that of cyberattacks. Smurf attack traffic is represented by blue bars and UDP attack traffic is represented by yellow bars.

5.3.1 Cache

"The Cache performance object consists of counters that monitor the file system cache, an area of physical memory that stores recently used data as long as possible to permit access to the data without having to read from the disk. Because applications typically use the cache, the cache is monitored as an indicator of application I/O operations. When memory is plentiful, the cache can grow, but when memory is scarce, the cache can become too small to be effective."

In Cache performance object we focus on Dirty Page Threshold, Dirty Pages, and Pin Read Hits %. Each counter has its definition from Microsoft in quotation marks and our explanation of the results below its graph.

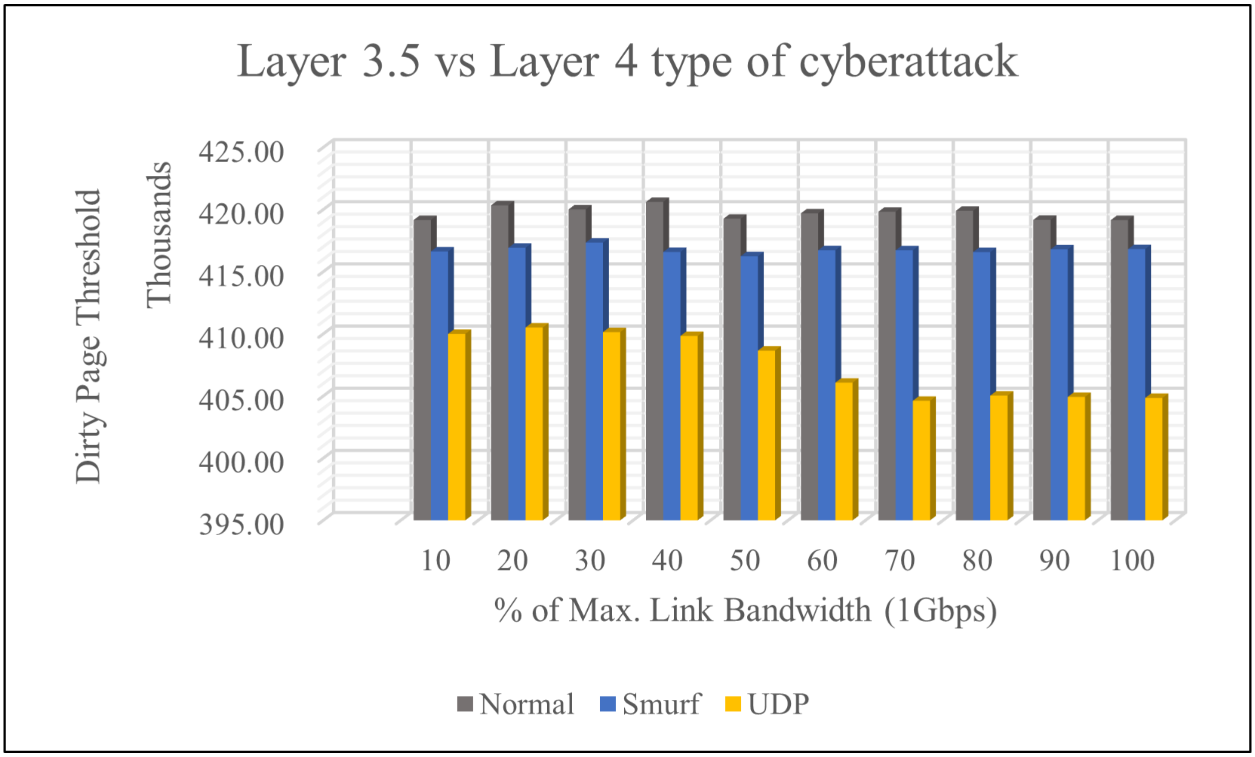


Figure 5 - 1 Cache: Dirty Page Threshold

“Threshold for number of dirty pages on system cache”

In Figure 5-1 the Dirty Page Threshold is affected the most during a Smurf attack in comparison to a UDP attack. The threshold of dirty pages remains very steady, slowly increasing throughout the different attack intensities of Smurf attack, while UDP’s threshold is reduced as the attack intensifies. As mentioned in previous chapters, according to [10] and [11] this is due to operating system strategy of reducing the threshold to get rid of the dirty pages in the cache quicker to keep the system from failing, which would affect the performance of the computer and integrity and availability of data or any output needed.

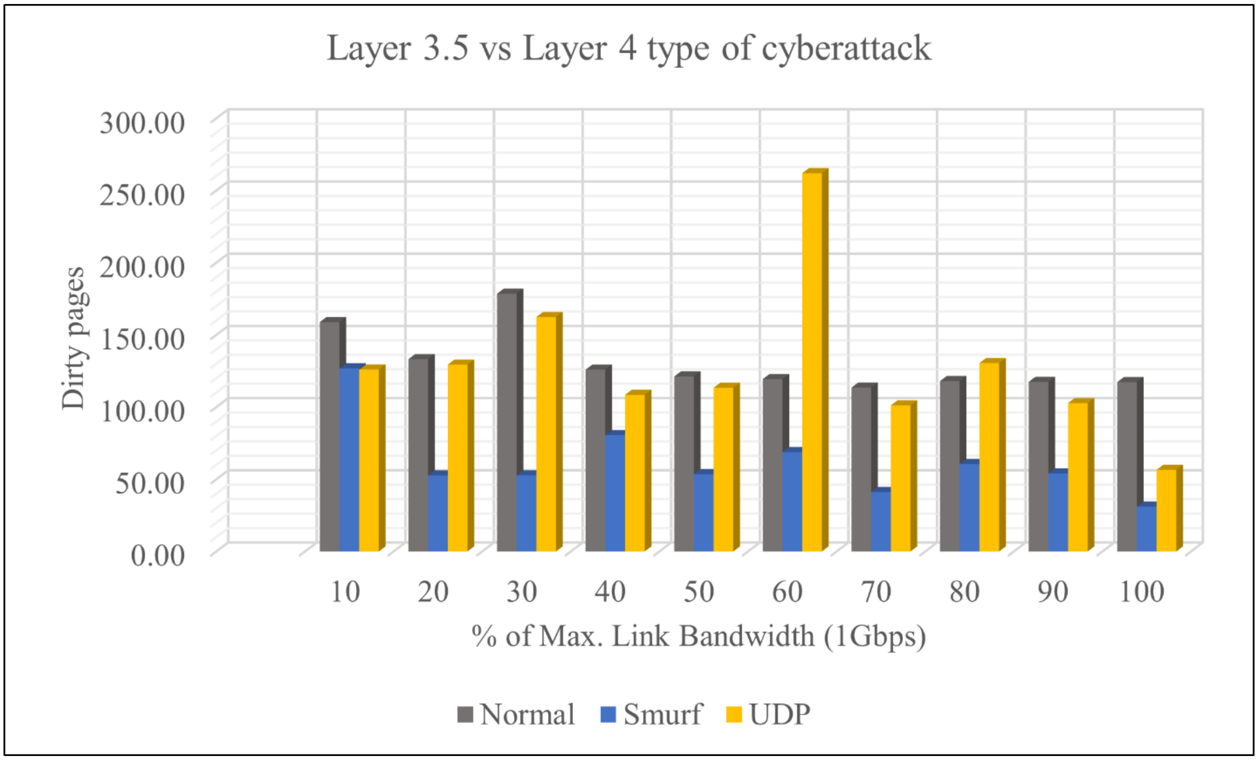


Figure 5 - 2 Cache: Dirty Pages

“Total number of dirty pages on the system cache”

Figure 5 – 2 shows the effect a Smurf attack has on Dirty Pages compared to UDP. It is not very clear what is going on here. From the graph we can tell the number of dirty pages during a UDP attack are more than during the Smurf attack. However, there are instances where a more in-depth analysis needs to be done. For example, at 60% of the attack intensity of UDP attack suffers a spike increase in dirty pages, while during the Smurf attack the number of dirty pages stay close to the trend. One thing that is for sure is that, since the cache’s purpose is to gather most frequently used information to keep it close to the processor, that could be reason why we see higher bars at the beginning of the attack. We must remember that these bars display the activity of cache and it makes sense that the bars will be higher at the beginning of the attack showing that data is being gather from other parts of memory to provide easier access by the processor.

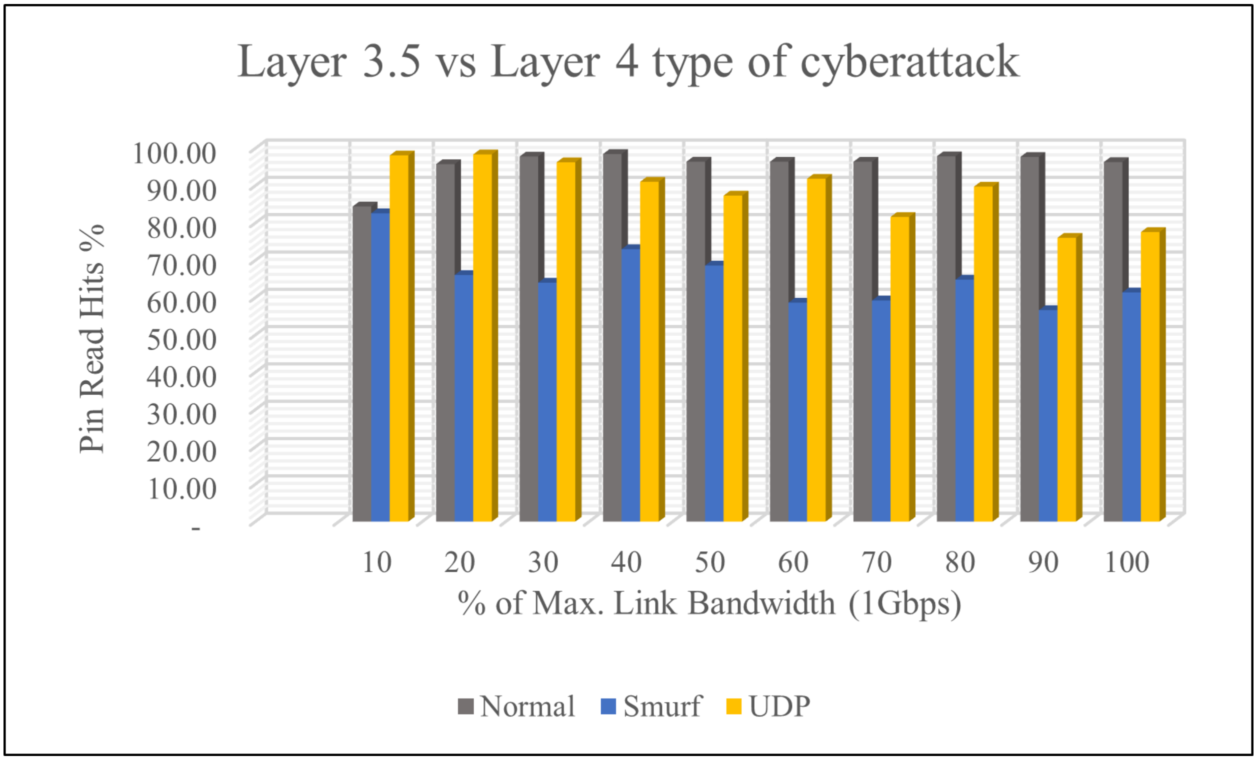


Figure 5 - 3 Cache: Pin Read Hits Percent

“Pin Read Hits is the percentage of pin read requests that hit the file system cache, i.e., did not require a disk read in order to provide access to the page in the file system cache. While pinned, a page's physical address in the file system cache will not be altered. The LAN Redirector uses this method for retrieving data from the cache, as does the LAN Server for small transfers. This is usually the method used by the disk file systems as well.”

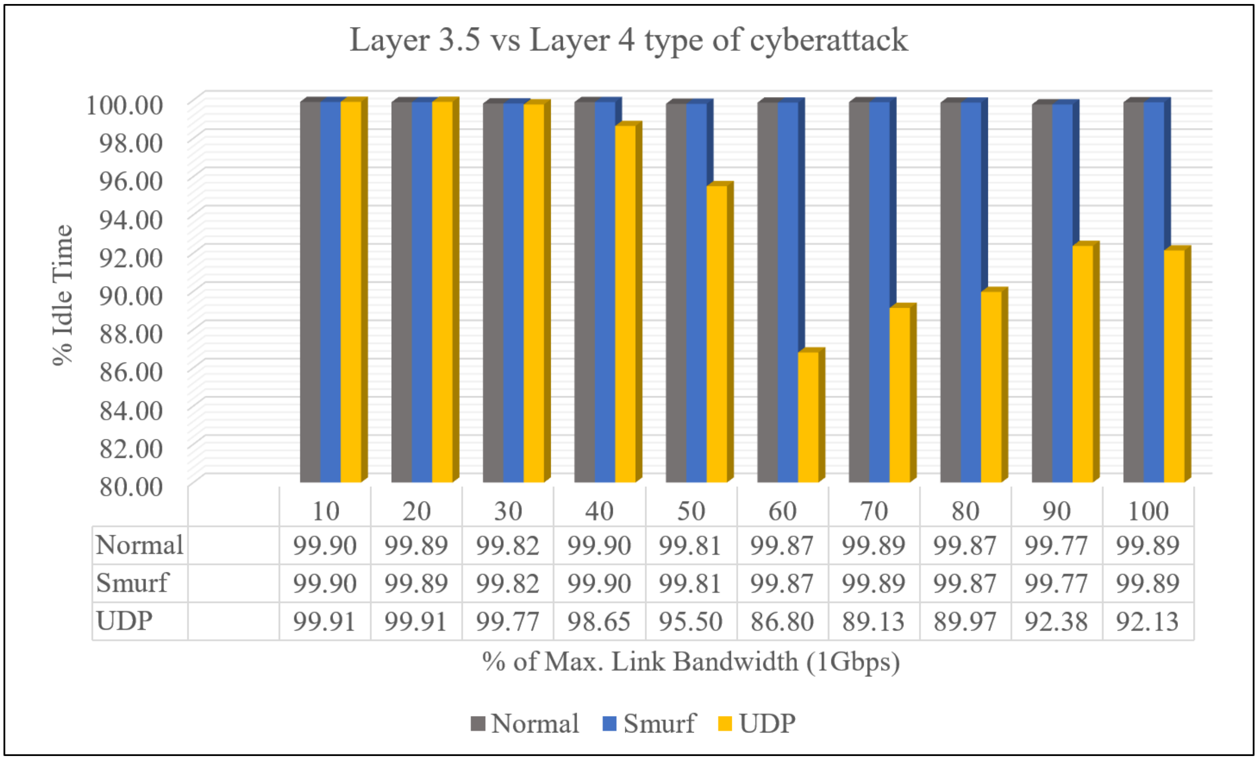
The Pin Read Hits Percent graph in Figure 5 – 3 indicates that during layer four attacks, UDP attack, the request for disk reads were more than during a Layer 3.5, Smurf attack. The disk read request is increased during UDP attack as the intensity of the attacks increments. The blue bars is networking simulated traffic of Smurf attack and the yellow bar is simulated traffic of UDP attack. The simulated attack does not come into play until bars at 10% of the attack where we see the highest number of Pin Read Hits %. The Pin Read Hits % then slowly decreased until reaching 75%. Again, this trend follows what we have explained for previous slides, that this is due to the way cache works.

5.3.2 Logical Disk

At first, it was unclear how much disk time DDOS attacks were taking -- we thought it would be minimum since in previous papers it was hardly mentioned. However, when we decided to isolate some counters related to disk, we found some interesting results.

> “The Logical Disk performance object consists of counters that monitor logical partitions of a hard or fixed disk drives. Performance Monitor identifies logical disks by their a drive letter, such as C.”

In Logical Disk performance object, we focused on % Idle Time, Disk Bytes/sec, Disk Write Bytes/sec and Disk Writes/sec. Each counter has its definition from Microsoft in quotation marks and our explanation of the results below its graph. Here we focus on these counters to get a better understanding of how the Logical Disk performance is affected by Smurf and UDP attacks, as we found these attacks to be the most damaging.

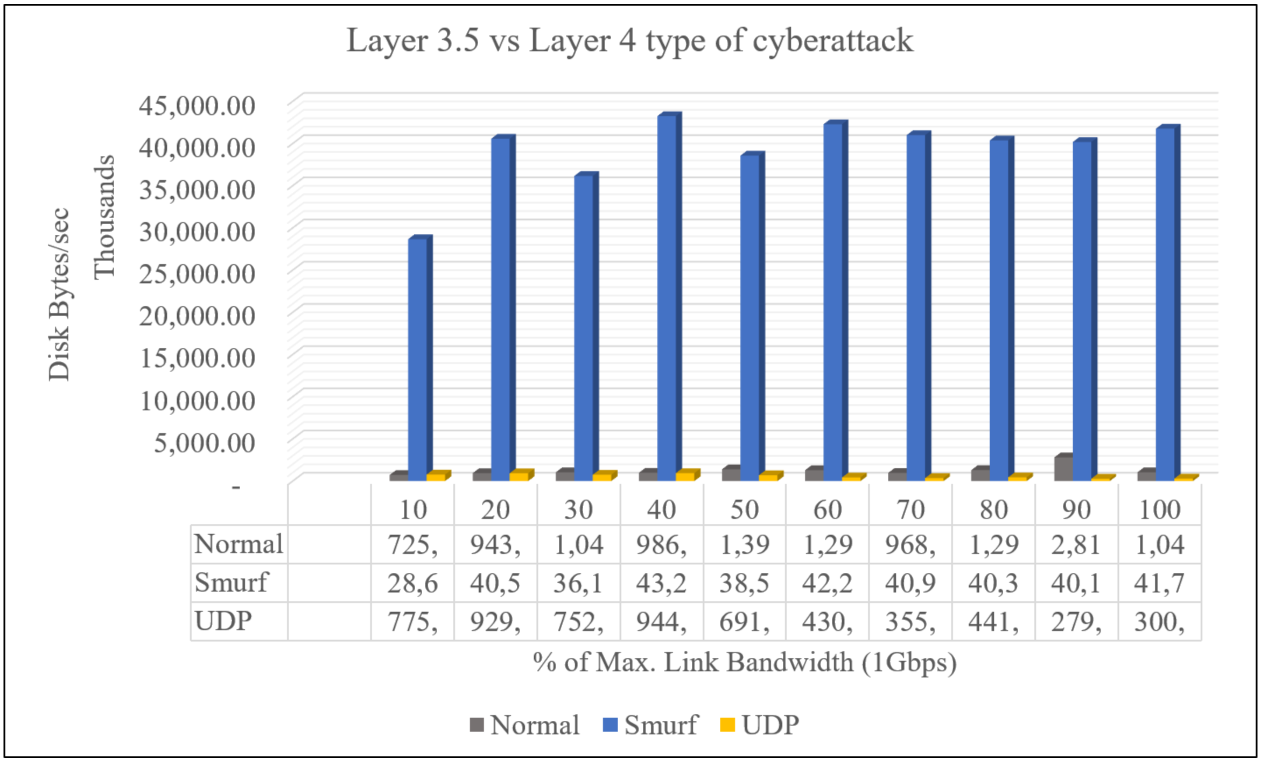


	10	20	30	40	50	60	70	80	90	100
Normal	99.90	99.89	99.82	99.90	99.81	99.87	99.89	99.87	99.77	99.89
Smurf	99.90	99.89	99.82	99.90	99.81	99.87	99.89	99.87	99.77	99.89
UDP	99.91	99.91	99.77	98.65	95.50	86.80	89.13	89.97	92.38	92.13

Figure 5 - 4 Logical Disk: Percentage Idle Time

“% Idle Time reports the percentage of time during the sample interval that the disk was idle.”

Figure 5 – 5 shows hardly any activity for Logical Disk before the cyberattacks, indicated by the gray-shaded bars marked Normal traffic, which represents the computer activity caused by legitimate traffic alone. During UDP attack the Logical Disk stayed idle about 98% of the time. And during the UDP attack, the logical disk activity started increasing at 40% of the attack intensity and reached 13% of activity before slowly decreasing the activity and reaching 8% of activity for 90-100% for the 1Gbps Link Bandwidth intensity. It would have been interesting to see what would have happened had we tested continuously for another hour at 100% intensity. Would the activity pick up or would it decrease to reach zero again?

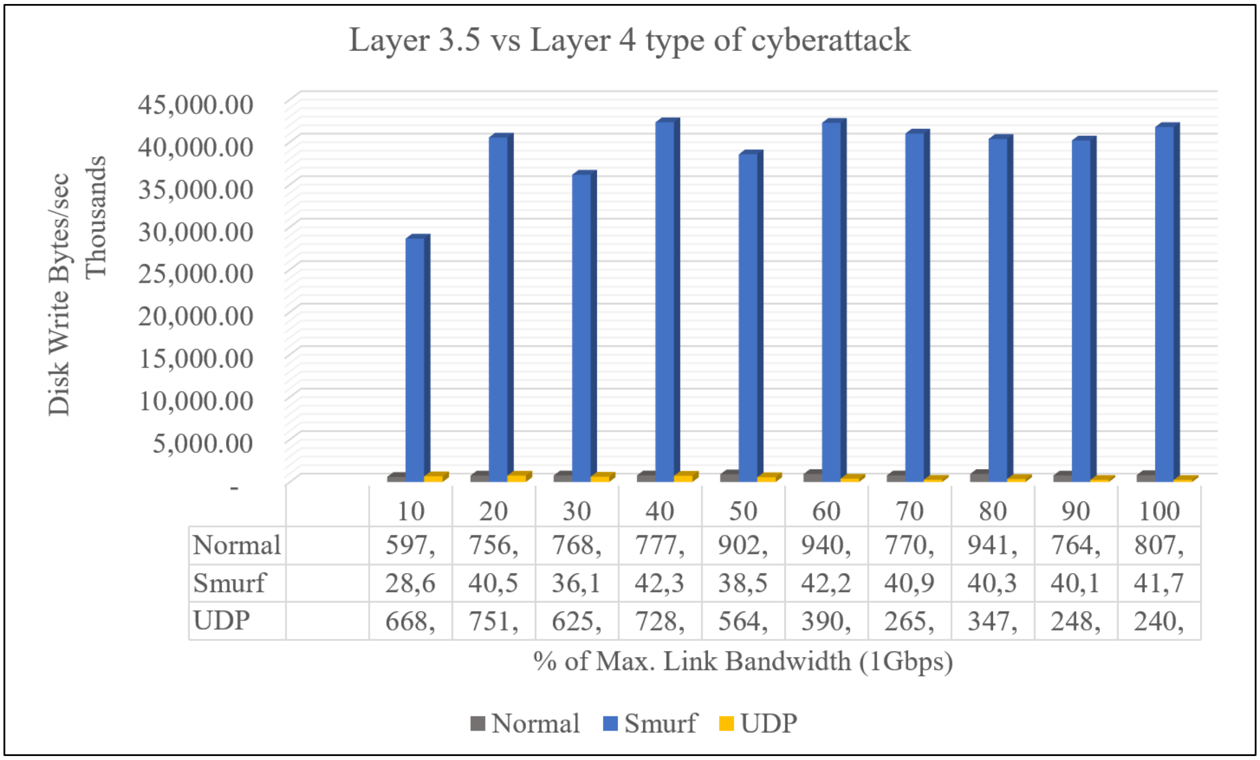


	10	20	30	40	50	60	70	80	90	100
Normal	725,	943,	1,04	986,	1,39	1,29	968,	1,29	2,81	1,04
Smurf	28,6	40,5	36,1	43,2	38,5	42,2	40,9	40,3	40,1	41,7
UDP	775,	929,	752,	944,	691,	430,	355,	441,	279,	300,

Figure 5 - 5 Logical Disk: Disk Bytes per second

> "Disk Bytes/sec is the rate bytes are transferred to or from the disk during write or read operations."

The graph, Disk Bytes per second in Figure 5 -5 shows a very high activity in the Logical Disk during Smurf attack. Here we hardly see any yellow bars but it does make evident that during Smurf attack the Logical Disk is far more active than during the UDP attack. For UDP Attack we can see, in chapter 4, an immediate rise in the number of bytes being transferred back and forth, when the attack began at 10% of the bandwidth intensity, reaching a maximum close to one thousand bytes. Then, there is a steady decline in the migration of bytes, reaching a minimum at 90% of the bandwidth intensity with around 250 bytes per second. This counter could be studied farther. What happens if the bandwidth continues to increase in intensity to 200%? Would the bytes transferred per second increase, or would they stay below 500 disk bytes per second? On the other hand, we can see that for test when there was no traffic there was a tendency for bytes to increase. This may indicate that there is more communication with the disk when there is no attack than when there is one happening. This could be communication interference and could be due to saturation of processor resources.

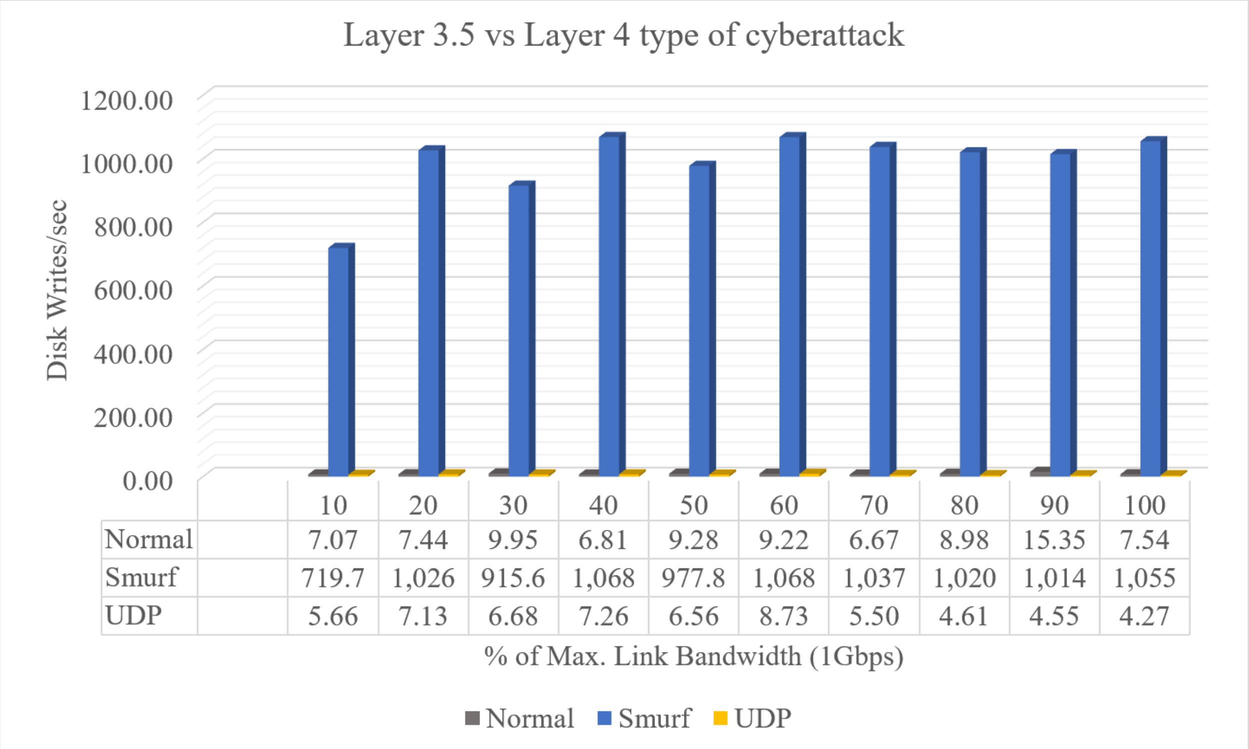


		10	20	30	40	50	60	70	80	90	100
Normal		597,	756,	768,	777,	902,	940,	770,	941,	764,	807,
Smurf		28,6	40,5	36,1	42,3	38,5	42,2	40,9	40,3	40,1	41,7
UDP		668,	751,	625,	728,	564,	390,	265,	347,	248,	240,

Figure 5 - 6 Logical Disk: Disk Write Bytes per second

"Disk Write Bytes/sec is rate at which bytes are transferred to the disk during write operations."

In Figure 5-6 we can see that when a Smurf attack is happening disk write operations are drastically affected, more so than during a UDP attack. We can see this by comparing these graphs to the gray graph bars, which represents the Disk Write Bytes/sec activity while the normal network activity is happening. Even though we hardly see the UDP attack bars, from chapter 4 we know that UDP attack did not affect until the attack intensity reached 30%. From there on, the write bytes per second operations were limited to 250 write bytes per second, on average. From this graph we can conclude that Smurf attack affects the disk write operations far more than a UDP attack.



	10	20	30	40	50	60	70	80	90	100
Normal	7.07	7.44	9.95	6.81	9.28	9.22	6.67	8.98	15.35	7.54
Smurf	719.7	1,026	915.6	1,068	977.8	1,068	1,037	1,020	1,014	1,055
UDP	5.66	7.13	6.68	7.26	6.56	8.73	5.50	4.61	4.55	4.27

Figure 5 - 7 Logical Disk: Disk Writes per second

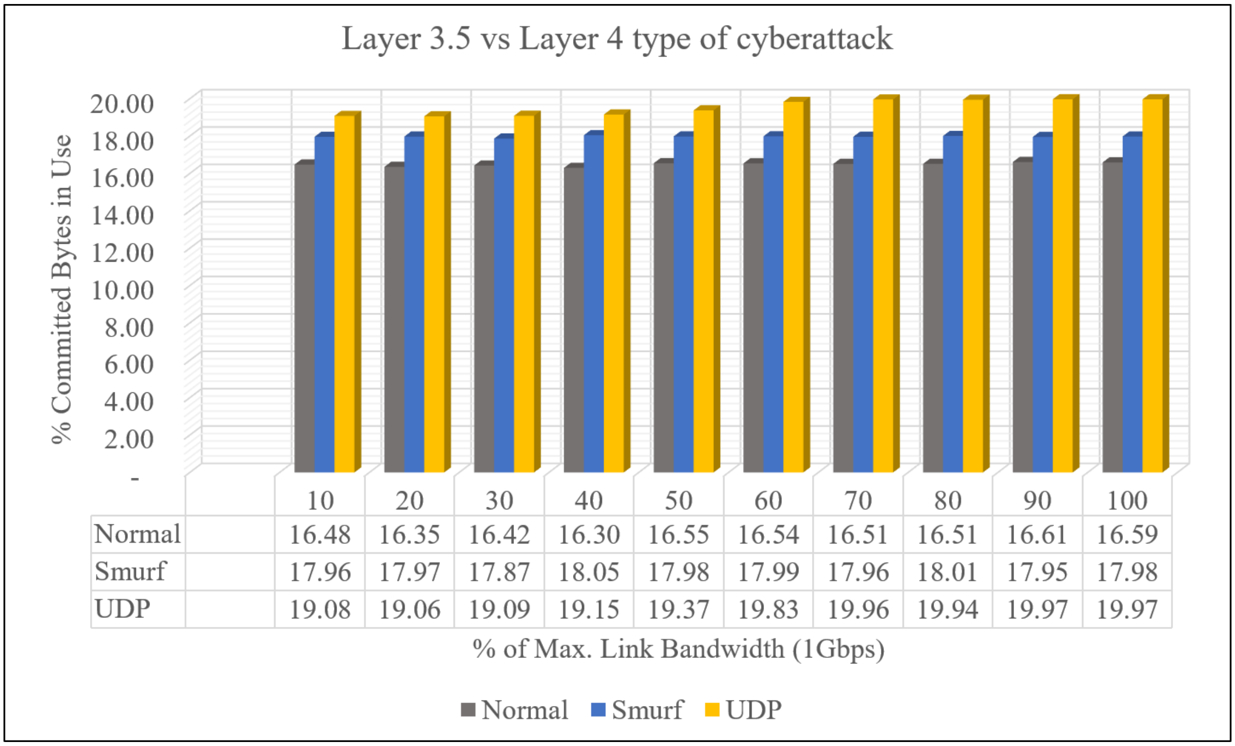
"Disk Writes/sec is the rate of write operations on the disk."

Figure 5-7 is the Disk Writes per second that happen during Smurf attack and UDP attack, compared to Normal traffic Writes per second. Here we can see that the traffic generated by Smurf attack is far greater than that of UDP attack. The difference is very noticeable, as the activity from the UDP attack is barely noticeable. It would be interesting to analyze this activity more closely.

5.3.3 Memory

"The Memory performance object consists of counters that describe the behavior of physical and virtual memory on the computer. Physical memory is the amount of random access memory on the computer. Virtual memory consists of the space in physical memory and on disk. Many of the memory counters monitor paging, which is the movement of pages of code and data between disk and physical memory. Excessive paging, a symptom of a memory shortage, can cause delays which interfere with all system processes."

In Memory performance object we focus on % Committed Bytes In Use, Available Mbytes, and Page Faults/sec, and Pin Read Hits %. Each counter has its definition from Microsoft in quotation marks and our explanation of the results below its graph.

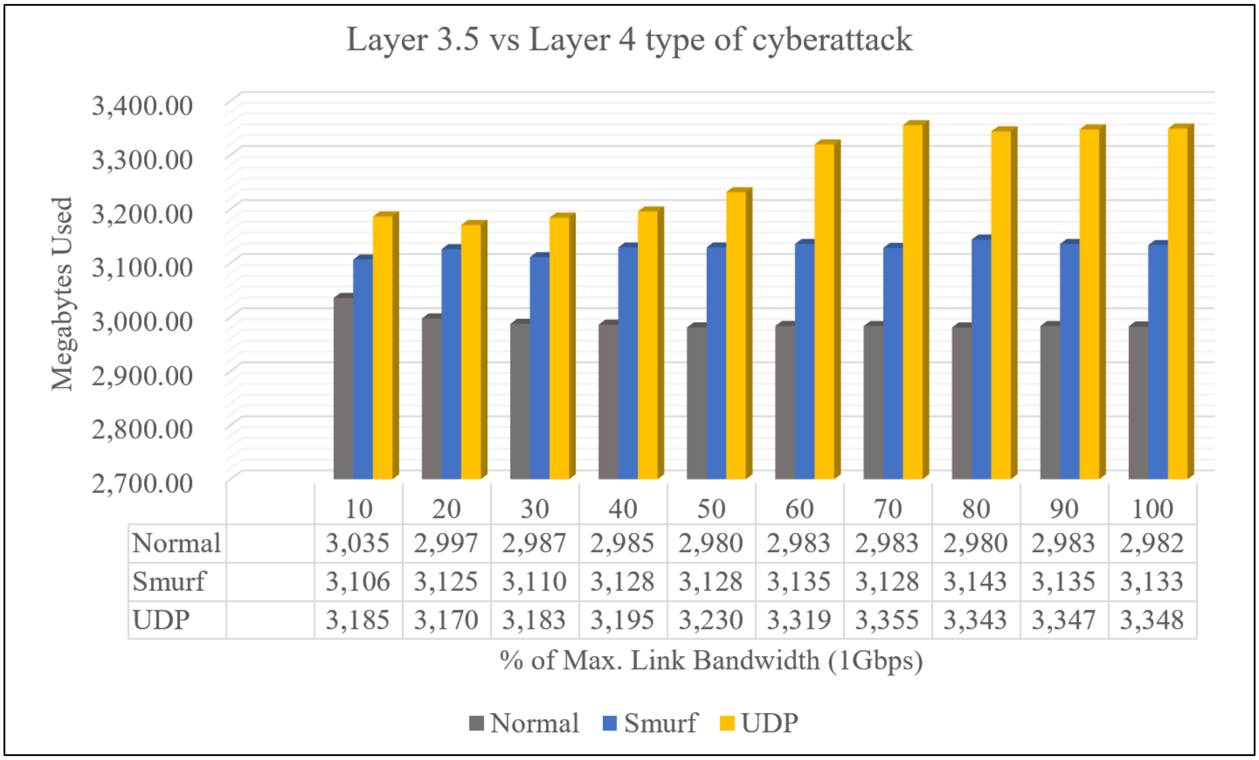


	10	20	30	40	50	60	70	80	90	100
Normal	16.48	16.35	16.42	16.30	16.55	16.54	16.51	16.51	16.61	16.59
Smurf	17.96	17.97	17.87	18.05	17.98	17.99	17.96	18.01	17.95	17.98
UDP	19.08	19.06	19.09	19.15	19.37	19.83	19.96	19.94	19.97	19.97

Figure 5 - 8 Memory: Percent Committed Bytes In Use

"% Committed Bytes In Use is the ratio of Memory\\Committed Bytes to the Memory\\Commit Limit. Committed memory is the physical memory in use for which space has been reserved in the paging file should it need to be written to disk. The commit limit is determined by the size of the paging file. If the paging file is enlarged, the commit limit increases, and the ratio is reduced). This counter displays the current percentage value only; it is not an average.”

Something interesting about Figure 5-8 is that the graph shows 16% committed bytes while computer is running during normal traffic. This happens for both legitimate traffic, starting at 10% of the maximum link bandwidth, and persists until the test ends. Comparing the normal result against Smurf attack and UDP attack traffic we notice that there are more bytes from Memory committed during the UDP attack than Smurf attack traffic. This is evidence that these two forms of cyberattack affect memory of the computer. This comparison also helps us determine that a UDP attack does use more memory than a Smurf attack.

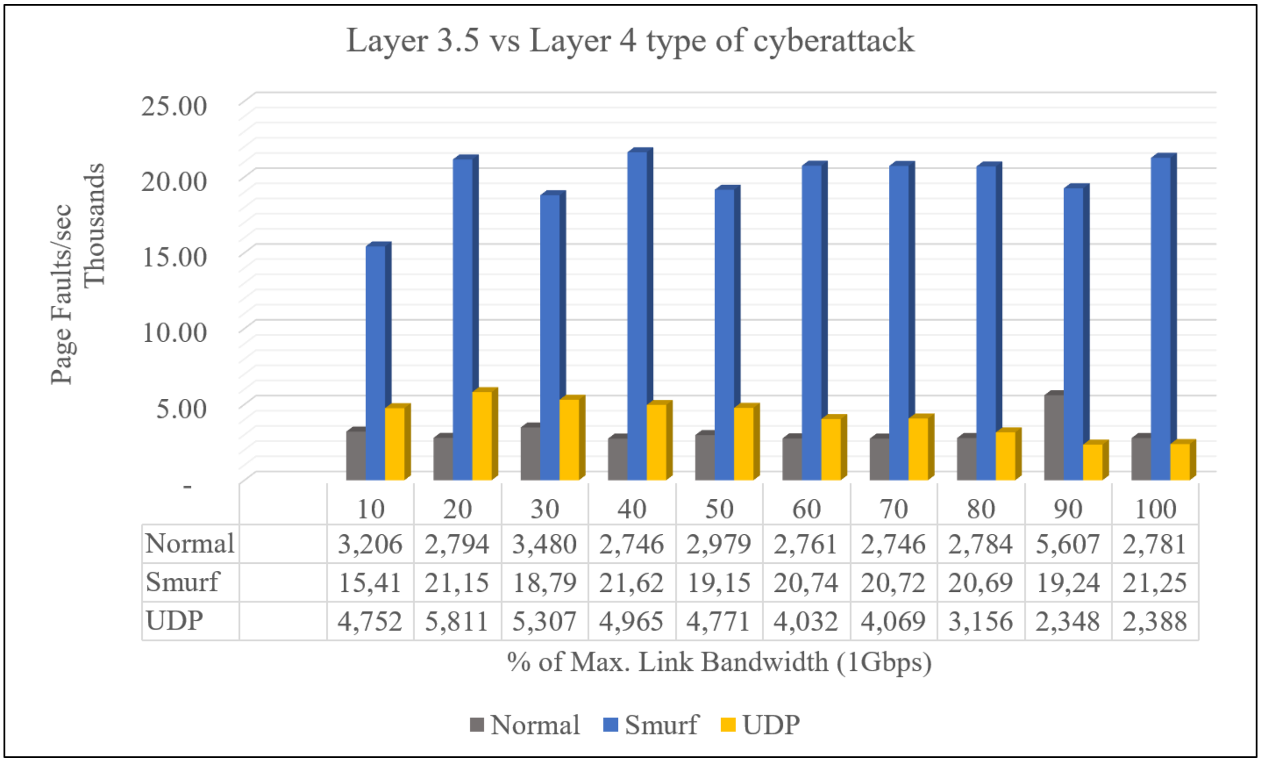


	10	20	30	40	50	60	70	80	90	100
Normal	3,035	2,997	2,987	2,985	2,980	2,983	2,983	2,980	2,983	2,982
Smurf	3,106	3,125	3,110	3,128	3,128	3,135	3,128	3,143	3,135	3,133
UDP	3,185	3,170	3,183	3,195	3,230	3,319	3,355	3,343	3,347	3,348

Figure 5 - 9 Memory: Available Megabytes

“Available MBytes is the amount of physical memory, in Megabytes, immediately available for allocation to a process or for system use. It is equal to the sum of memory assigned to the standby (cached), free and zero page lists.”

Figure 5-9 above show the Megabytes available. Our victim computer has 16 Gigabytes of memory. Knowing that, we can deduct from this graph that the megabytes occupied during a UDP attack increases as the attack intensifies during both Smurf and UDP attacks. For Smurf attack the difference is not much. However, if we compare it to the base memory occupied during normal traffic, we can see an increment in memory usage during Smurf attack. Also, for UDP attack, we can see the increase of megabytes after 40% of the attack intensity and continues to increase until 70% of the attack intensity of the 1 Gbps max link bandwidth. This is another clear evidence of the effect these cyberattack have on main memory. UDP attack, in this case, affects memory more so than Smurf attack.



	10	20	30	40	50	60	70	80	90	100
Normal	3,206	2,794	3,480	2,746	2,979	2,761	2,746	2,784	5,607	2,781
Smurf	15,41	21,15	18,79	21,62	19,15	20,74	20,72	20,69	19,24	21,25
UDP	4,752	5,811	5,307	4,965	4,771	4,032	4,069	3,156	2,348	2,388

Figure 5 - 10 Memory: Page Faults per second

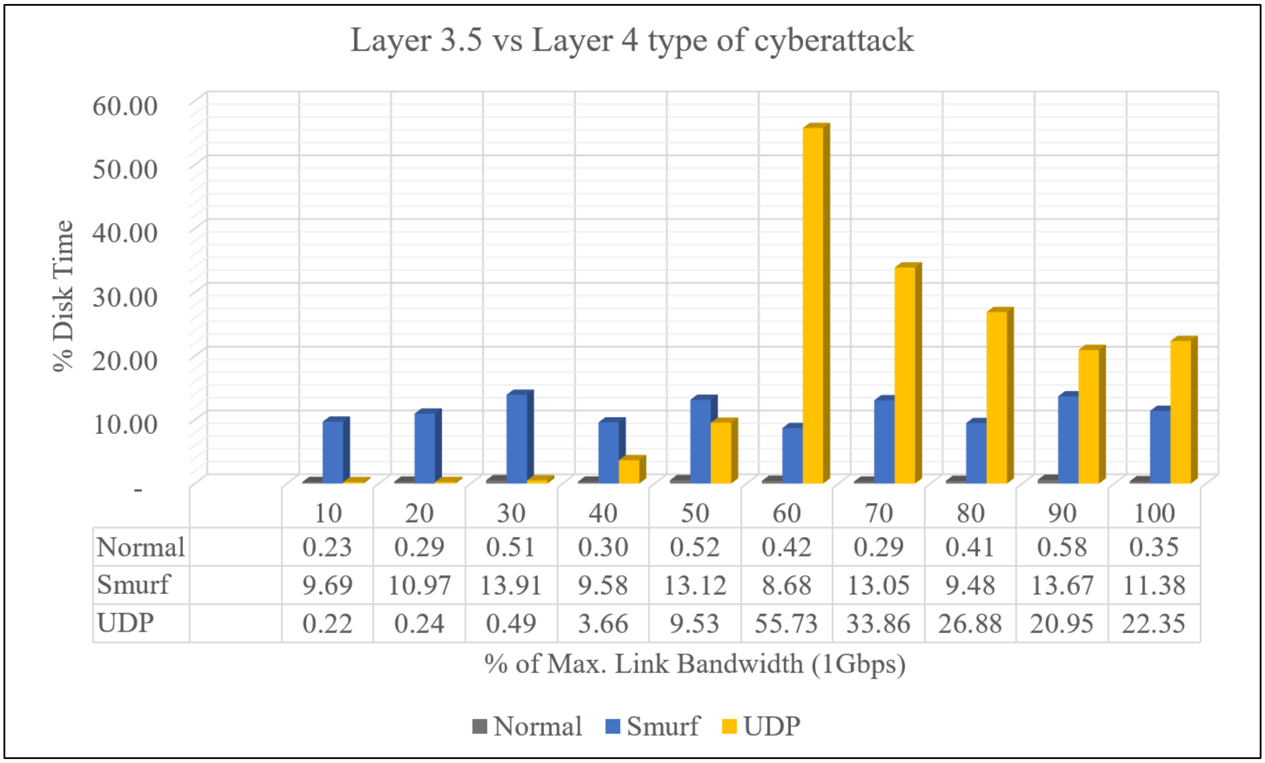
“Page Faults/sec is the average number of pages faulted per second. It is measured in number of pages faulted per second because only one page is faulted in each fault operation, hence this is also equal to the number of page fault operations. This counter includes both hard faults (those that require disk access) and soft faults (where the faulted page is found elsewhere in physical memory.) Most processors can handle large numbers of soft faults without significant consequence. However, hard faults, which require disk access, can cause significant delays."

In the graph above, Figure 4-10, we can observe the amount of page faults generated during normal traffic compared to traffic while a UDP attack and Smurf attack are happening. Page faults happened during normal traffic on average of three thousand Page Faults per second during UDP attack. During a Smurf attack the average of page faults further increase reaching more than 25 thousand faults per second. We can see in the graph this significant rise starting at 10% of the attack intensity. UDP attack almost reached six thousand Page Faults per second at 20% of the attack intensity, after which the decline starts until reaching a stability at 90% of the attack intensity.

5.3.4 Physical Disk

In Physical Disk performance object, we focused on % Disk Time, % Disk Write Time, and % Idle Time. We present the definition as provided by Microsoft in quotation marks followed by our explanation of the results below its graph.

"The Physical Disk performance object consists of counters that monitor hard or fixed disk drive on a computer. Disks are used to store file, program, and paging data and are read to retrieve these items, and written to record changes to them. The values of physical disk counters are sums of the values of the logical disks (or partitions) into which they are divided."



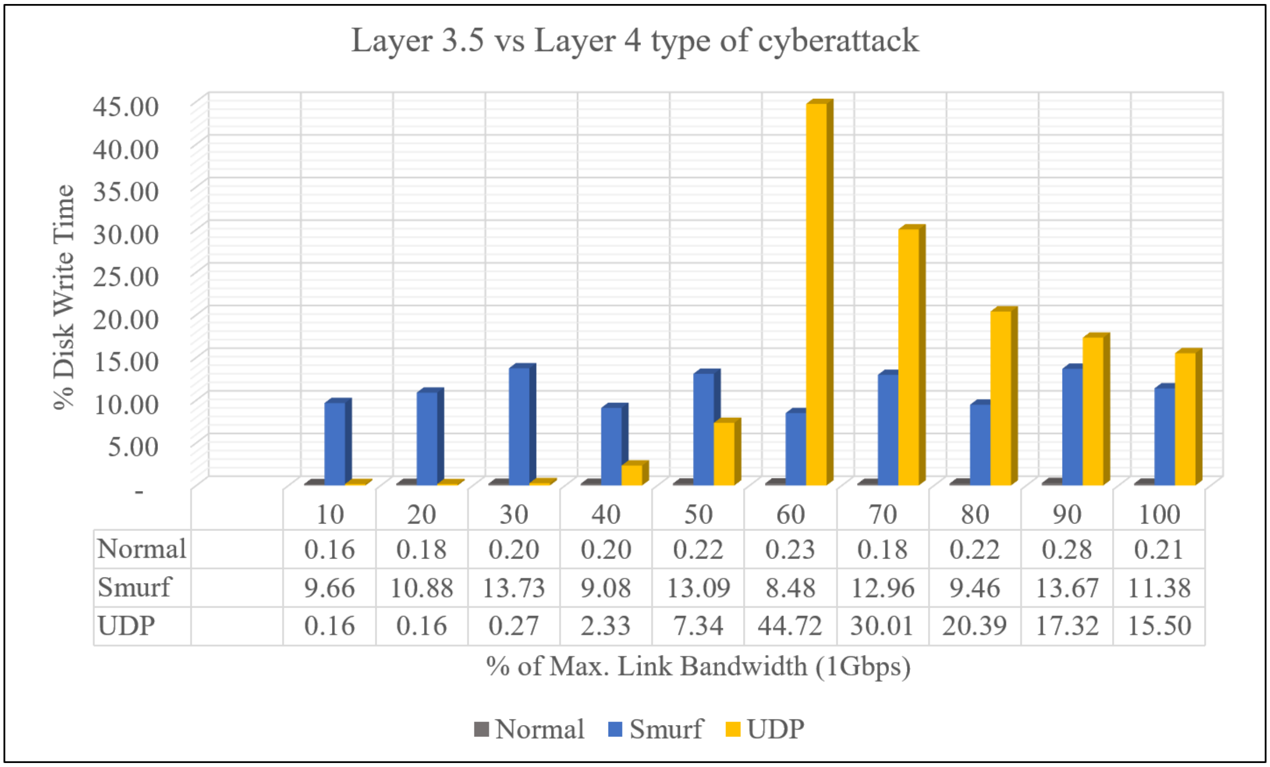
	10	20	30	40	50	60	70	80	90	100
Normal	0.23	0.29	0.51	0.30	0.52	0.42	0.29	0.41	0.58	0.35
Smurf	9.69	10.97	13.91	9.58	13.12	8.68	13.05	9.48	13.67	11.38
UDP	0.22	0.24	0.49	3.66	9.53	55.73	33.86	26.88	20.95	22.35

Figure 5 - 11 Physical Disk: Percent Disk Time

"% Disk Time is the percentage of elapsed time that the selected disk drive was busy servicing read or write requests."

Percent Disk Time graph in Figure 5-11 contains the % Disk Time counter of Physical Disk three times. In this graph we can barely recognized the traffic from Normal traffic due to the low percentage of time it was busy; however, we can see a difference when the Smurf attack and even a more pronounced result during UDP attack.

During Smurf Attack we can see the disk was busy up to 10% of the time throughout the duration of the attack. During the UDP attack, however, we can notice the activity of the disk increasing slowly from 10% to 50% of the activity. The disk time drastically jumps up to 55% in activity at 60% of the attack intensity. It then has a pronounced decline in activity until reaching 90% of the attack intensity where we see that it stabilizes. It would be interesting to see more results of this counter.

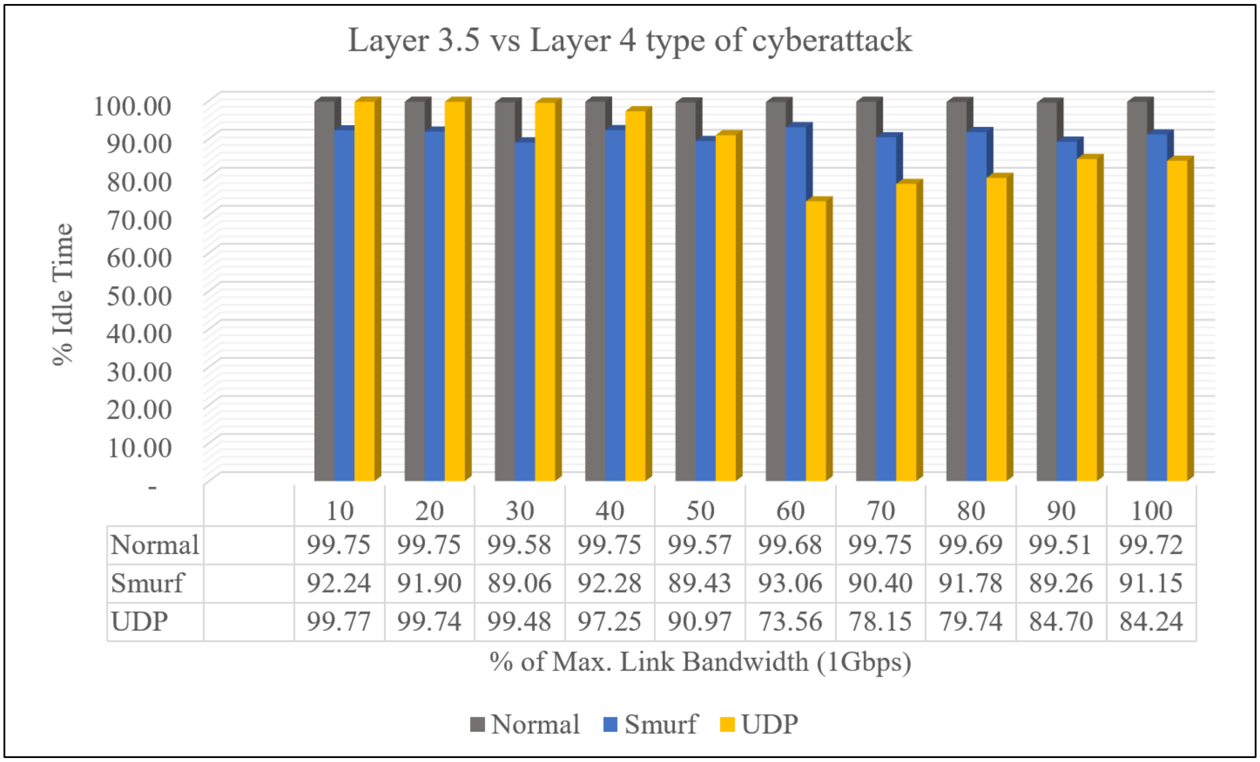


	10	20	30	40	50	60	70	80	90	100
Normal	0.16	0.18	0.20	0.20	0.22	0.23	0.18	0.22	0.28	0.21
Smurf	9.66	10.88	13.73	9.08	13.09	8.48	12.96	9.46	13.67	11.38
UDP	0.16	0.16	0.27	2.33	7.34	44.72	30.01	20.39	17.32	15.50

Figure 5 - 12 Physical Disk: Percent Disk Write Time

> “% Disk Write Time is the percentage of elapsed time that the selected disk drive was busy servicing write requests.”

Comparing Figure 5-12 to Figure 5-11 we can deduce that most of the time the disk was busy it was due to disk writes. As we did in Figure 5-11, we can see a very low percent of percent disk write time for the normal traffic in Figure 5-12 as well. We also see the percent disk write time rise to no more than 5% during the TCP attack. However, during the UDP attack the percent disk write time increased starting at 40% of the attack intensity reaching 44% disk write time at 60% of the attack intensity before declining to 15% disk write time when the attack intensity reached 100%. We can deduce that the decrement of the percent disk write time will stabilize at about 15% disk write time even if the attack continues to increase but more testing that involve letting the test run longer or doing more observations of higher traffic intensities would be interesting to observe.



	10	20	30	40	50	60	70	80	90	100
Normal	99.75	99.75	99.58	99.75	99.57	99.68	99.75	99.69	99.51	99.72
Smurf	92.24	91.90	89.06	92.28	89.43	93.06	90.40	91.78	89.26	91.15
UDP	99.77	99.74	99.48	97.25	90.97	73.56	78.15	79.74	84.70	84.24

Figure 5 - 3 Physical Disk: Percent Idle Time

"% Idle Time reports the percentage of time during the sample interval that the disk was idle."

5.2.5. Process

"The Process performance object consists of counters that monitor running application program and system processes. All the threads in a process share the same address space and have access to the same data."

In Process performance object, we focused on % Privileged Time, % Processor Time, % User Time, IO Write Operations/sec and Page Faults/sec. We present the definition for each counter, as provided by Microsoft, in quotation marks followed by our explanation of the results below its graph.

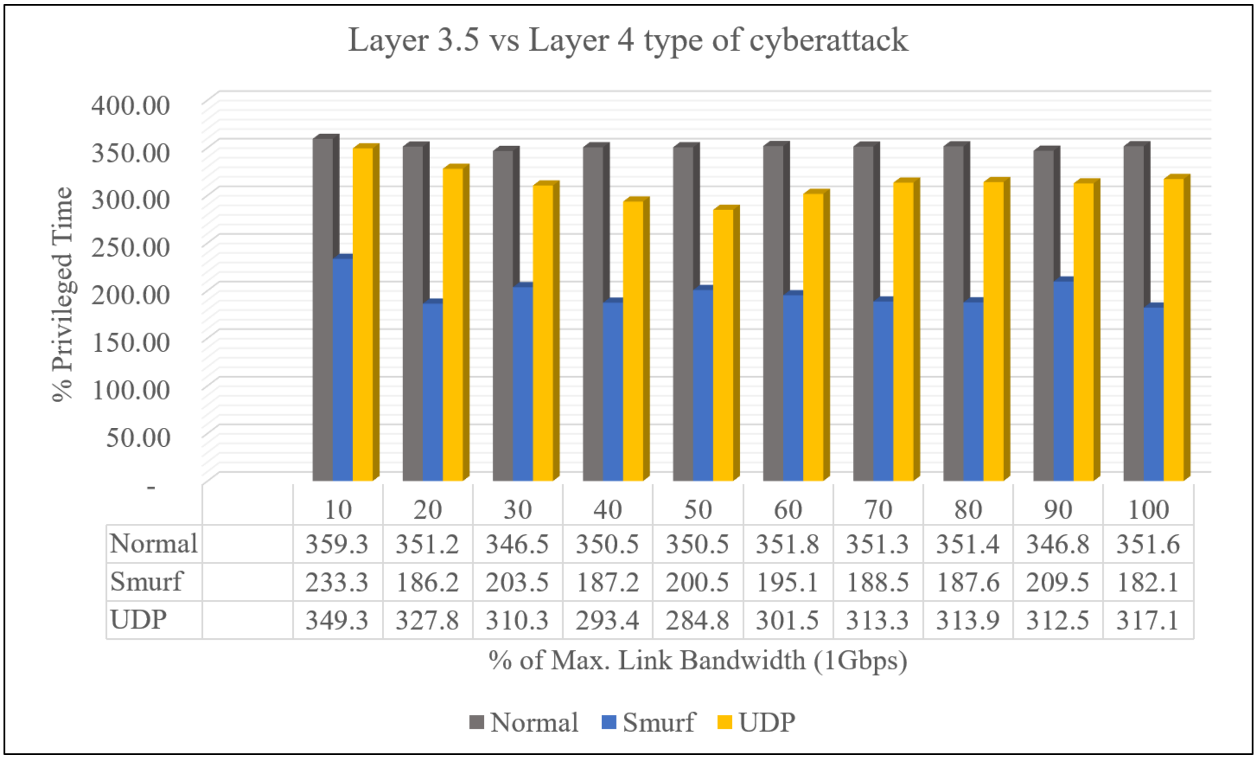
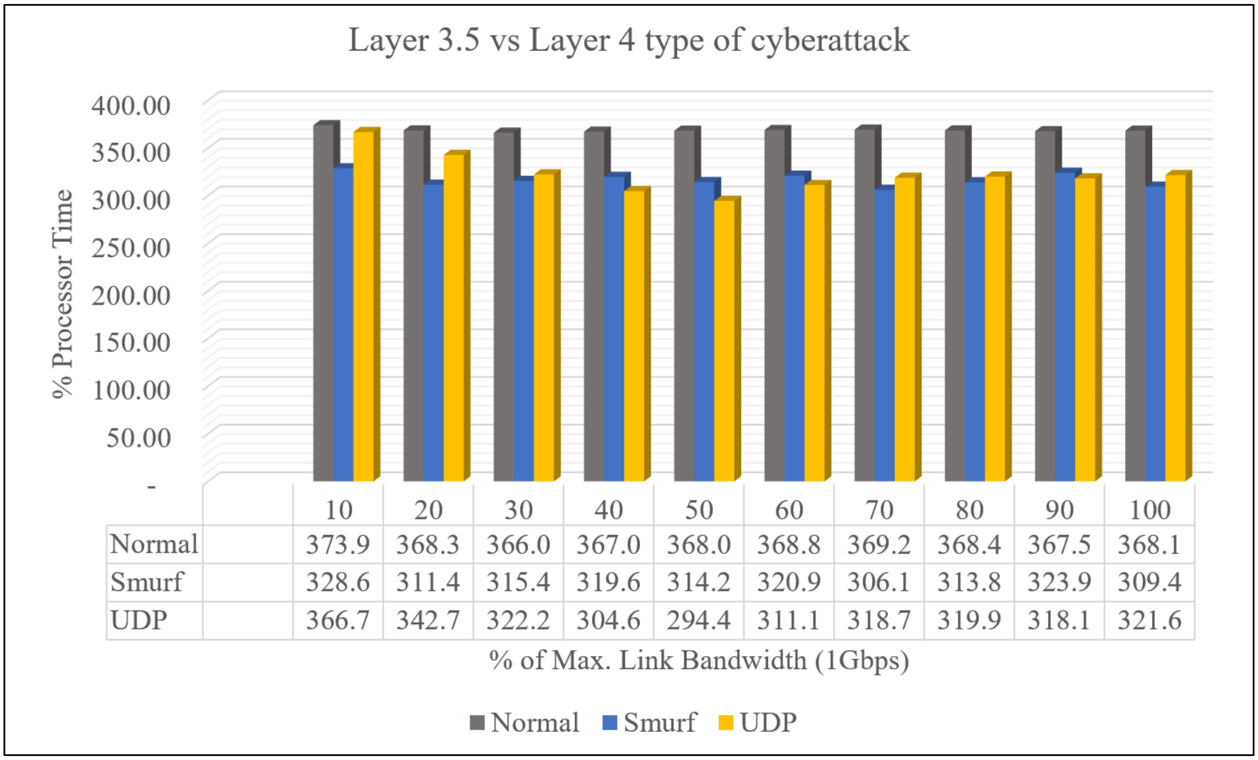


Figure 5 - 14 Process: Percent Privileged Time

"% Privileged Time is the percentage of elapsed time that the process threads spent executing code in privileged mode. When a Windows system service is called, the service will often run in privileged mode to gain access to system-private data. Such data is protected from access by threads executing in user mode. Calls to the system can be explicit or implicit, such as page faults or interrupts. Unlike some early operating systems, Windows uses process boundaries for subsystem protection in addition to the traditional protection of user and privileged modes. Some work done by Windows on behalf of the application might appear in other subsystem processes in addition to the privileged time in the process."

Figure 5-14 is the % Privileged Time. Here we can see how Privileged Time is being used by a Smurf attack versus how a UDP attack uses it. As we compare these results, we can see both UDP attack and Smurf attack reducing the Privileged Time percentage of the system. This shows that these two attacks surpass some of the protections found in the operating system.

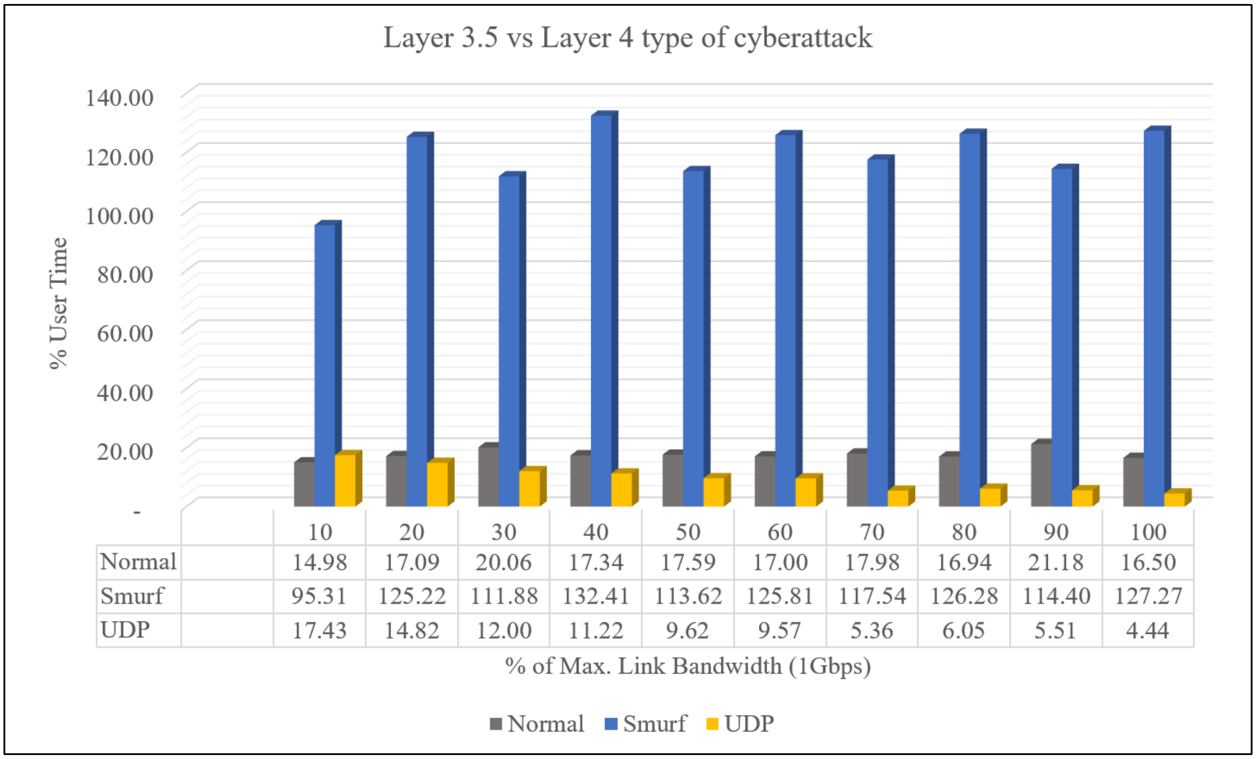


	10	20	30	40	50	60	70	80	90	100
Normal	373.9	368.3	366.0	367.0	368.0	368.8	369.2	368.4	367.5	368.1
Smurf	328.6	311.4	315.4	319.6	314.2	320.9	306.1	313.8	323.9	309.4
UDP	366.7	342.7	322.2	304.6	294.4	311.1	318.7	319.9	318.1	321.6

Figure 5 - 15 Process: Percent Processor Time

"% Processor Time is the percentage of elapsed time that all of process threads used the processor to execution instructions. An instruction is the basic unit of execution in a computer, a thread is the object that executes instructions, and a process is the object created when a program is run. Code executed to handle some hardware interrupts and trap conditions are included in this count."

% Processor Time graph above in Figure 5-15 has matching results to Figure 5-14.

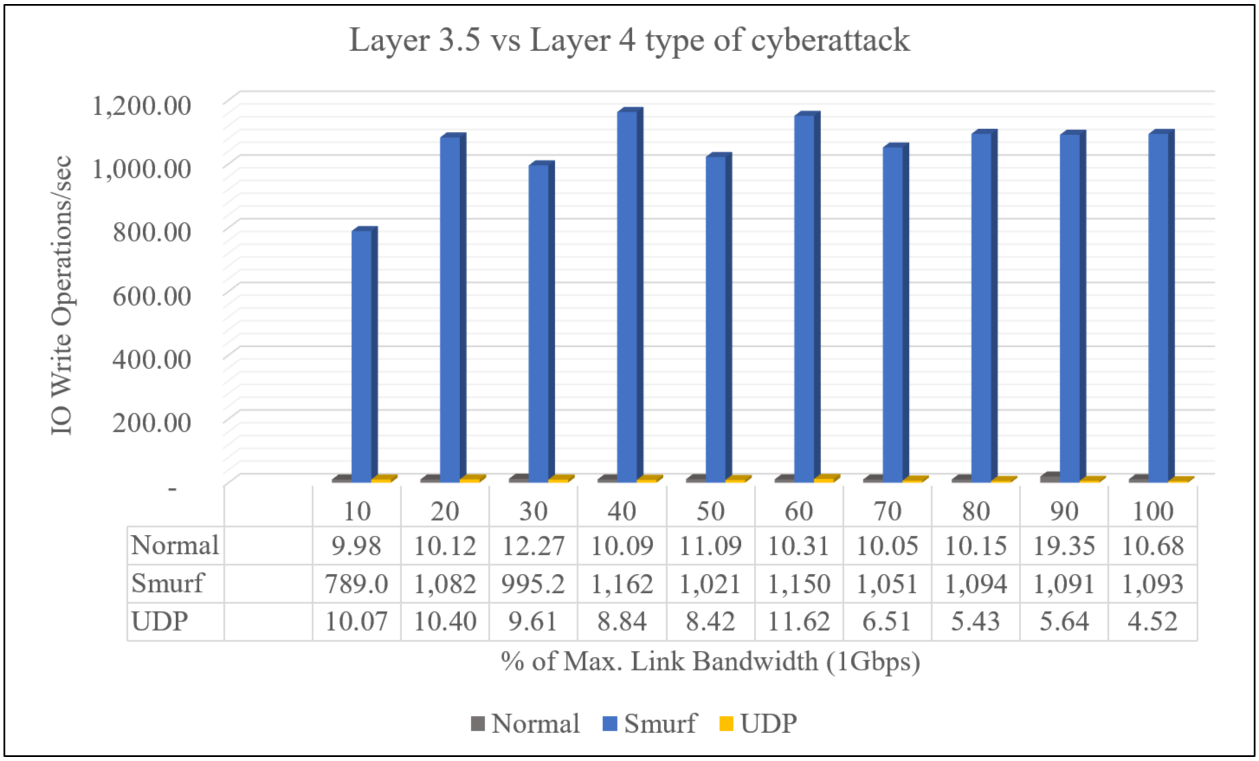


	10	20	30	40	50	60	70	80	90	100
Normal	14.98	17.09	20.06	17.34	17.59	17.00	17.98	16.94	21.18	16.50
Smurf	95.31	125.22	111.88	132.41	113.62	125.81	117.54	126.28	114.40	127.27
UDP	17.43	14.82	12.00	11.22	9.62	9.57	5.36	6.05	5.51	4.44

Figure 5 - 16 Process: Percent User Time

"% User Time is the percentage of elapsed time that the process threads spent executing code in user mode. Applications, environment subsystems, and integral subsystems execute in user mode. Code executing in user mode cannot damage the integrity of the Windows executive, kernel, and device drivers. Unlike some early operating systems, Windows uses process boundaries for subsystem protection in addition to the traditional protection of user and privileged modes. Some work done by Windows on behalf of the application might appear in other subsystem processes in addition to the privileged time in the process."

In Figure 5-16 we can see that the Percentage User Time when the computer receives normal traffic reaches 20%. However, we can tell by the graph that Layer 3.5 attack, % User Time heavily increases. This is clearly indicated by the prompt increment in user percentage at the 10% attack intensity during Smurf attack. This increment is kept steadily throughout the duration of the attack. For UDP attack, the decline in user time is not as severe but a decline is observed, none the less.

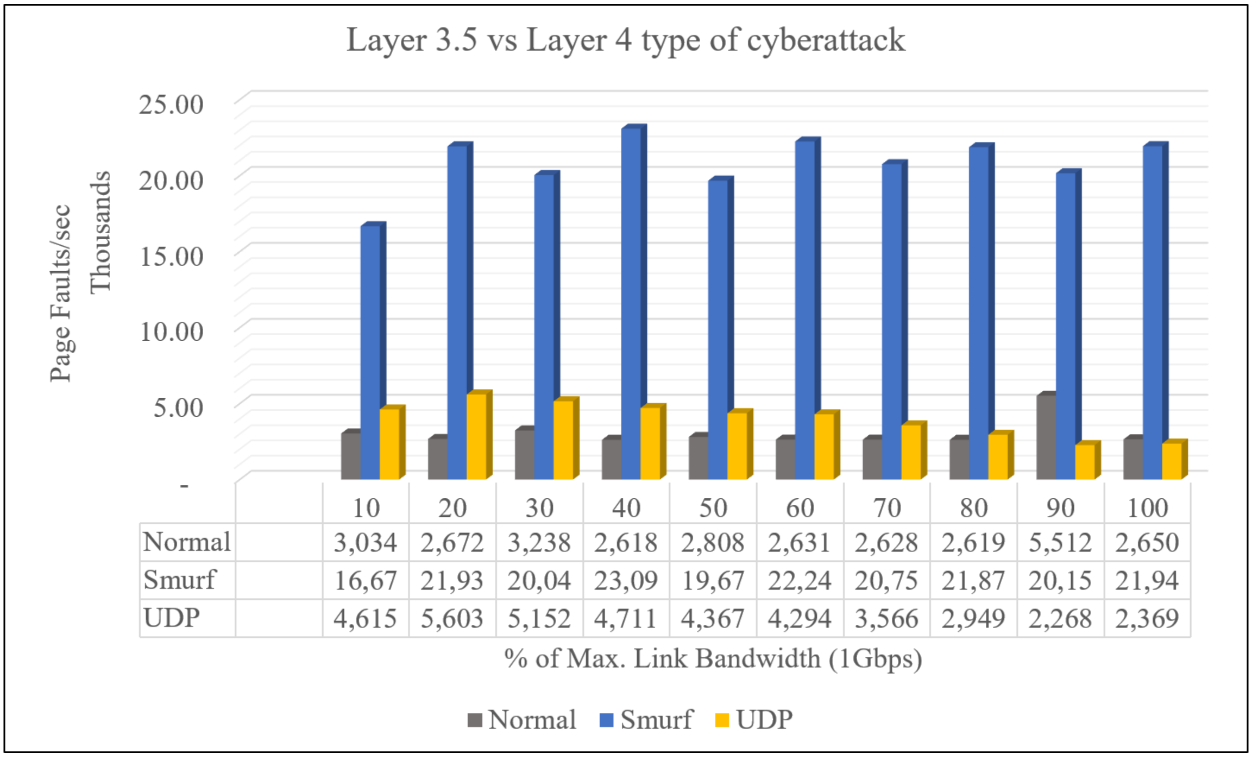


	10	20	30	40	50	60	70	80	90	100
Normal	9.98	10.12	12.27	10.09	11.09	10.31	10.05	10.15	19.35	10.68
Smurf	789.0	1,082	995.2	1,162	1,021	1,150	1,051	1,094	1,091	1,093
UDP	10.07	10.40	9.61	8.84	8.42	11.62	6.51	5.43	5.64	4.52

Figure 5 - 17 Process: Input/Output Write Operations per second

> "The rate at which the process is issuing write I/O operations. This counter counts all I/O activity generated by the process to include file, network and device I/Os."

Figure 5-16 presents the process activity from file, network, and input/output devices. This graph shows that there is more activity during Normal traffic than for a UDP attack. A Smurf attack increases the input/output activity more than a UDP attack.



	10	20	30	40	50	60	70	80	90	100
Normal	3,034	2,672	3,238	2,618	2,808	2,631	2,628	2,619	5,512	2,650
Smurf	16,67	21,93	20,04	23,09	19,67	22,24	20,75	21,87	20,15	21,94
UDP	4,615	5,603	5,152	4,711	4,367	4,294	3,566	2,949	2,268	2,369

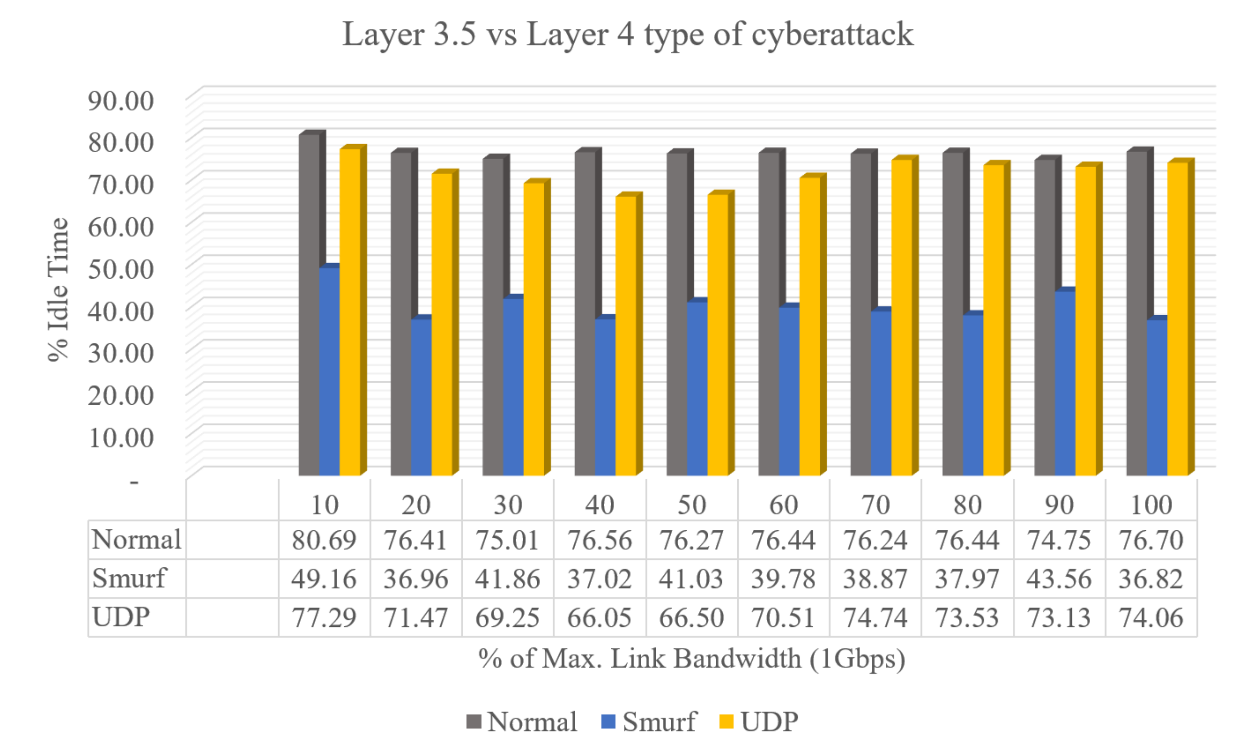
Figure 5 - 18 Process: Page Faults per second

> “Page Faults/sec is the rate at which page faults by the threads executing in this process are occurring. A page fault occurs when a thread refers to a virtual memory page that is not in its working set in main memory. This may not cause the page to be fetched from disk if it is on the standby list and hence already in main memory, or if it is in use by another process with whom the page is shared.”

Figure 5-18 shows the Page Faults that occur per second when a Smurf attack and a UDP attack is happening. The activity is not as pronounced in this Layer 4 attack as it is in the results for Layer 3.5 attack presented in chapter 3. Here, something to notice is the activity of the UDP attack. As shown in the graph, the page faults increase once the attack begins. Then it decreases to the level of Normal Internet activity. Based on the description provided by Microsoft this could be due to the computer being able to find the pages needed in virtual memory. We see a activity in page faults when a Smurf Attack is introduced. Here is a clear difference that can be observe to differentiate the two layers.

5.3.6 Processor

“The Processor performance object consists of counters that measure aspects of processor activity. The processor is the part of the computer that performs arithmetic and logical computations, initiates operations on peripherals, and runs the threads of processes. A computer can have multiple processors. The processor object represents each processor as an instance of the object.”

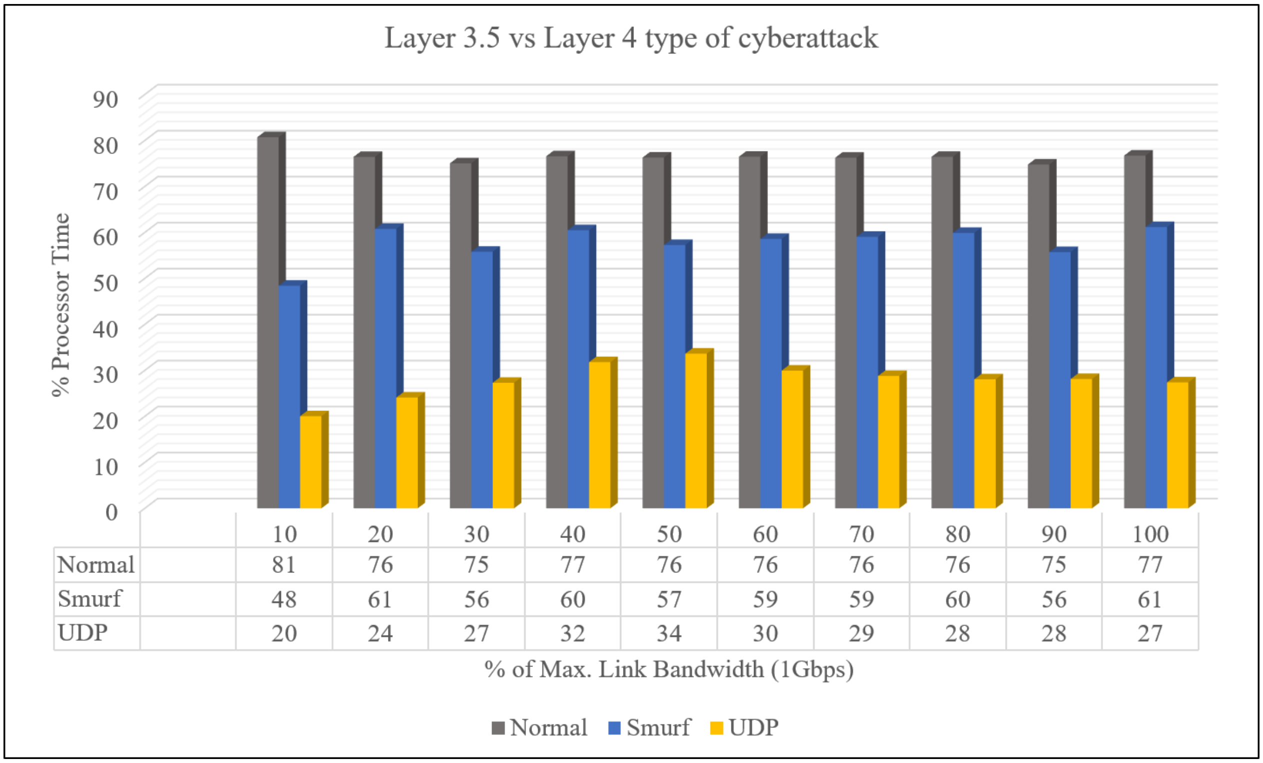


	10	20	30	40	50	60	70	80	90	100
Normal	80.69	76.41	75.01	76.56	76.27	76.44	76.24	76.44	74.75	76.70
Smurf	49.16	36.96	41.86	37.02	41.03	39.78	38.87	37.97	43.56	36.82
UDP	77.29	71.47	69.25	66.05	66.50	70.51	74.74	73.53	73.13	74.06

Figure 5 - 19 Processor: Percent Idle Time

“% Idle Time is the percentage of time the processor is idle during the sample interval”

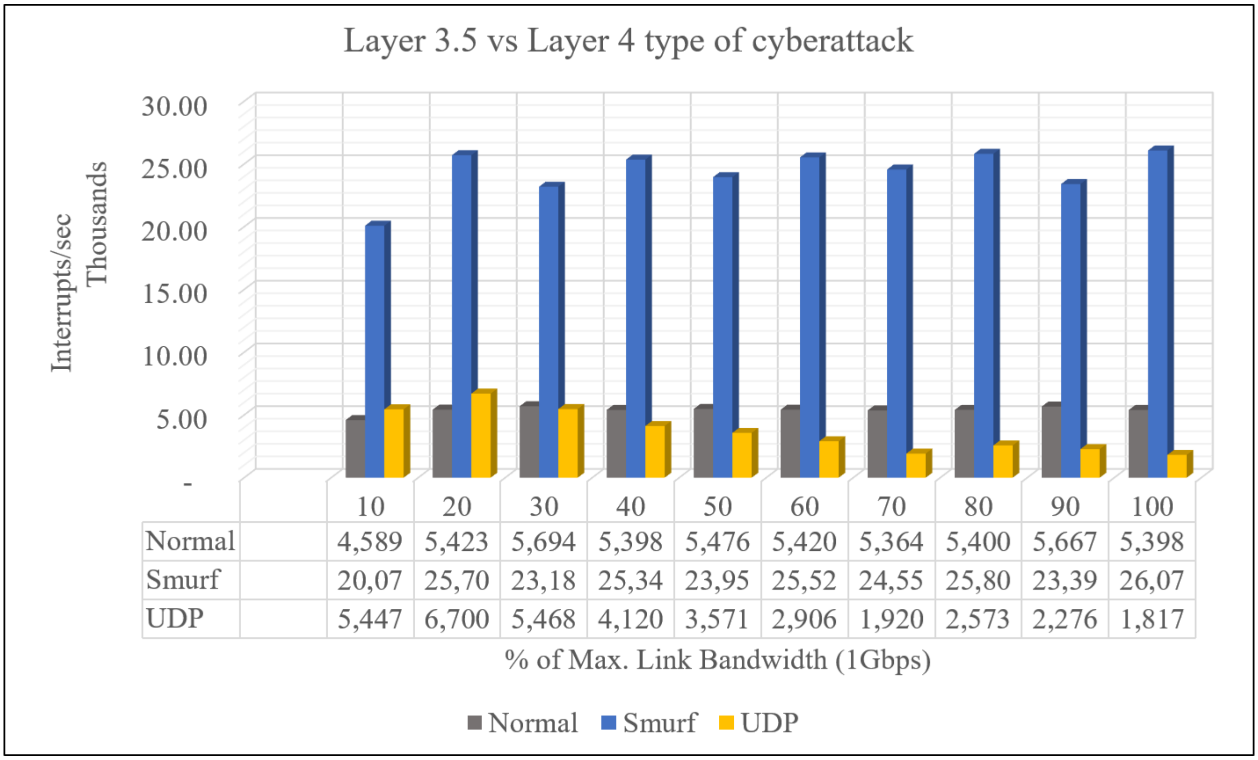
In Figure 5-19 we can notice how the computer is more active during a Smurf attack than during a UDP attack. On the 10% intensity both Normal and UDP attack displayed a similar idle time percentage. UDP idleness decreased as the intensity of the traffic increased, but the percentage kept about the same for these two counters. There was a bigger difference for the Smurf counter. We can see Smurf attack's % Idle Time kept a steady percentage throughout the test. From this graph we can say that the processor activity is less during a UDP attack and that a Smurf attack affects a computer processor more than during a UDP attack.



	10	20	30	40	50	60	70	80	90	100
Normal	81	76	75	77	76	76	76	76	75	77
Smurf	48	61	56	60	57	59	59	60	56	61
UDP	20	24	27	32	34	30	29	28	28	27

Figure 5 - 20 Processor: Percentage Processor Time

"% Processor Time is the percentage of elapsed time that the processor spends to execute a non-Idle thread. It is calculated by measuring the percentage of time that the processor spends executing the idle thread and then subtracting that value from 100%. (Each processor has an idle thread that consumes cycles when no other threads are ready to run). This counter is the primary indicator of processor activity, and displays the average percentage of busy time observed during the sample interval. It should be noted that the accounting calculation of whether the processor is idle is performed at an internal sampling interval of the system clock (10ms). On todays fast processors, % Processor Time can therefore underestimate the processor utilization as the processor may be spending a lot of time servicing threads between the system clock sampling interval. Workload based timer applications are one example of applications which are more likely to be measured inaccurately as timers are signaled just after the sample is taken."



	10	20	30	40	50	60	70	80	90	100
Normal	4,589	5,423	5,694	5,398	5,476	5,420	5,364	5,400	5,667	5,398
Smurf	20,07	25,70	23,18	25,34	23,95	25,52	24,55	25,80	23,39	26,07
UDP	5,447	6,700	5,468	4,120	3,571	2,906	1,920	2,573	2,276	1,817

Figure 5 - 21 Processor: Interrupts per second

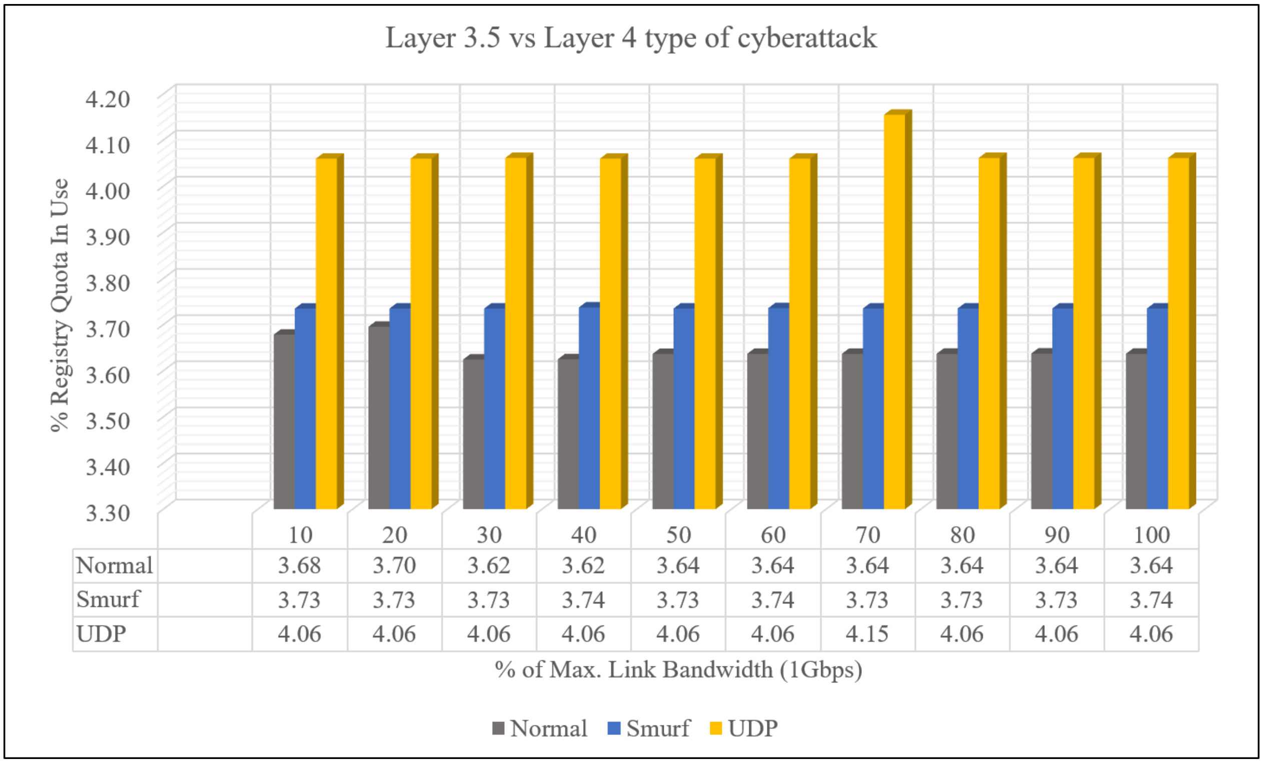
"Interrupts/sec is the average rate, in incidents per second, at which the processor received and serviced hardware interrupts. It does not include deferred procedure calls (DPCs), which are counted separately. This value is an indirect indicator of the activity of devices that generate interrupts, such as the system clock, the mouse, disk drivers, data communication lines, network interface cards, and other peripheral devices. These devices normally interrupt the processor when they have completed a task or require attention. Normal thread execution is suspended. The system clock typically interrupts the processor every 10 milliseconds, creating a background of interrupt activity. This counter displays the difference between the values observed in the last two samples, divided by the duration of the sample interval."

The graph in Figure 5-21 we can see the number of Interrupts per second, according to this graph UDP attack generates the most interrupts per second than TCP attack. We can also observe that the Interrupts happen when there is Internet activity. The first gray bar shows about one thousand interrupts per second. That is when the computer is idle. After that we see the interrupts activity increases to four thousand and then peaks to above five thousand with Internet activity. We can see Interrupts activity at the beginning of each attack, but that activity declines for both, at different periods. For UDP attack the number of Interrupts peaks at twenty percent of the attack intensity. Then it slowly declines to below the two thousand mark. This activity is lower than that generated by Normal traffic. Interrupt activity for TCP attack shows a more pronounced decline. It only rises to twenty five hundred Interrupts per second and from there the number of Interrupts keep declining until the end of the test, however, the difference from intensity to intensity is not too noticeable.

5.3.7 System

System performance object is analyzed, in this paper, using the following counters: % Registry Quota In Use, File Write Bytes/sec, and File Write Operations/sec and System Calls/sec.

"The System performance object consists of counters that apply to more than one instance of a component processors on the computer."



	10	20	30	40	50	60	70	80	90	100
Normal	3.68	3.70	3.62	3.62	3.64	3.64	3.64	3.64	3.64	3.64
Smurf	3.73	3.73	3.73	3.74	3.73	3.74	3.73	3.73	3.73	3.74
UDP	4.06	4.06	4.06	4.06	4.06	4.06	4.15	4.06	4.06	4.06

Figure 5 - 22 System: Percentage Registry Quota In Use

"% Registry Quota In Use is the percentage of the Total Registry Quota Allowed that is currently being used by the system. This counter displays the current percentage value only; it is not an average."

The registry, as explained in the book Internet Information Services (IIS) 6.0 by Microsoft Press, holds all type of information related to the computer, from what type of fonts the system uses to decryption classes, and component information that make the computer. The description of this counter says it is not 'an average' but we took samples every two seconds. Then, we averaged those samples to get this graph. Here we see the % Registry Quota in Use stays consistent throughout most of attacks intensities. The difference between the effects of UDP versus Smurf is very pronounced. Starting at 10% of the maximum link bandwidth of 1 Gbps the % Registry Quota In Use we found a very defined difference where Smurf attack starts going up to above 3.70%. We do see, however, that the Registry is affected by Smurf and UDP attacks. Here UDP attack has more effect on the registry quota than Smurf attack.

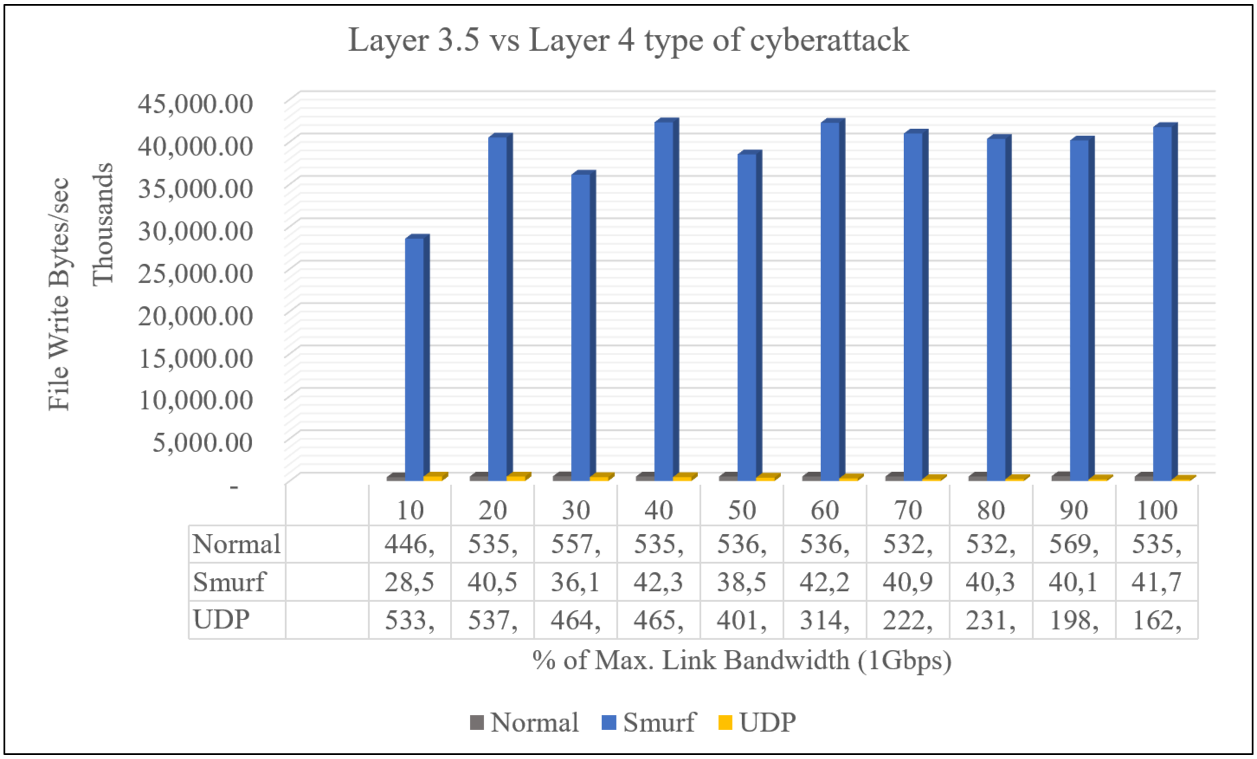
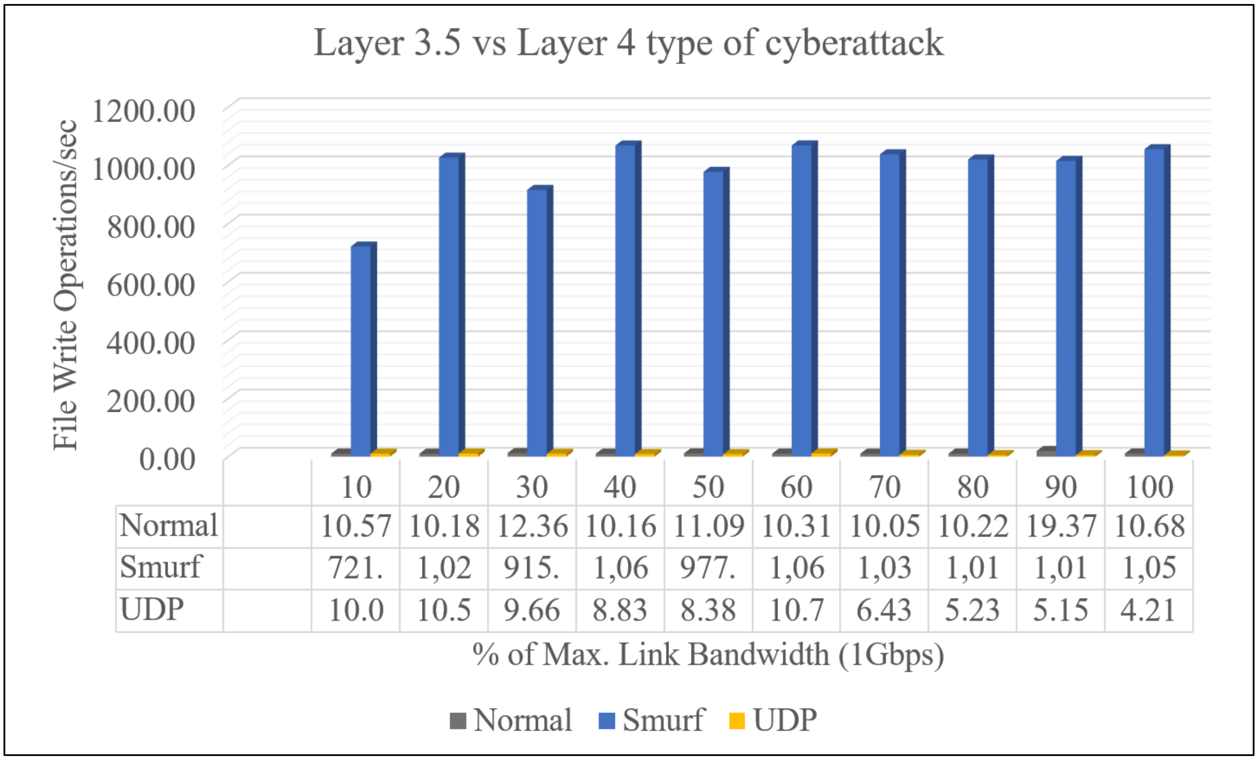


Figure 5 - 23 System: File Write Bytes per second

> "File Write Bytes/sec is the overall rate at which bytes are written to satisfy file system write requests to all devices on the computer, including writes to the file system cache. It is measured in number of bytes per second. This counter displays the difference between the values observed in the last two samples, divided by the duration of the sample interval."

Figure 5-23 shows how File Write Bytes are suppressed by UDP attacks. This is evident by observing and comparing the gray bars, which are shown in chapter 4, to the yellow bars, the UDP attack traffic. Here we can see how the Smurf attack overwhelmingly overpowers what happens during a UDP attack. The yellow bars representing the UDP attack are barely noticeable. From the results shown in chapter 4 for UDP, we know that the UDP attack activity barely reached the activity reached by a Normal traffic. Here, the computer activity is literally taken into hyper drive when Smurf attack is introduced. Smurf attack causes up to 43 thousand File Write Bytes per second, which is far more than the activity caused by the UDP attack.

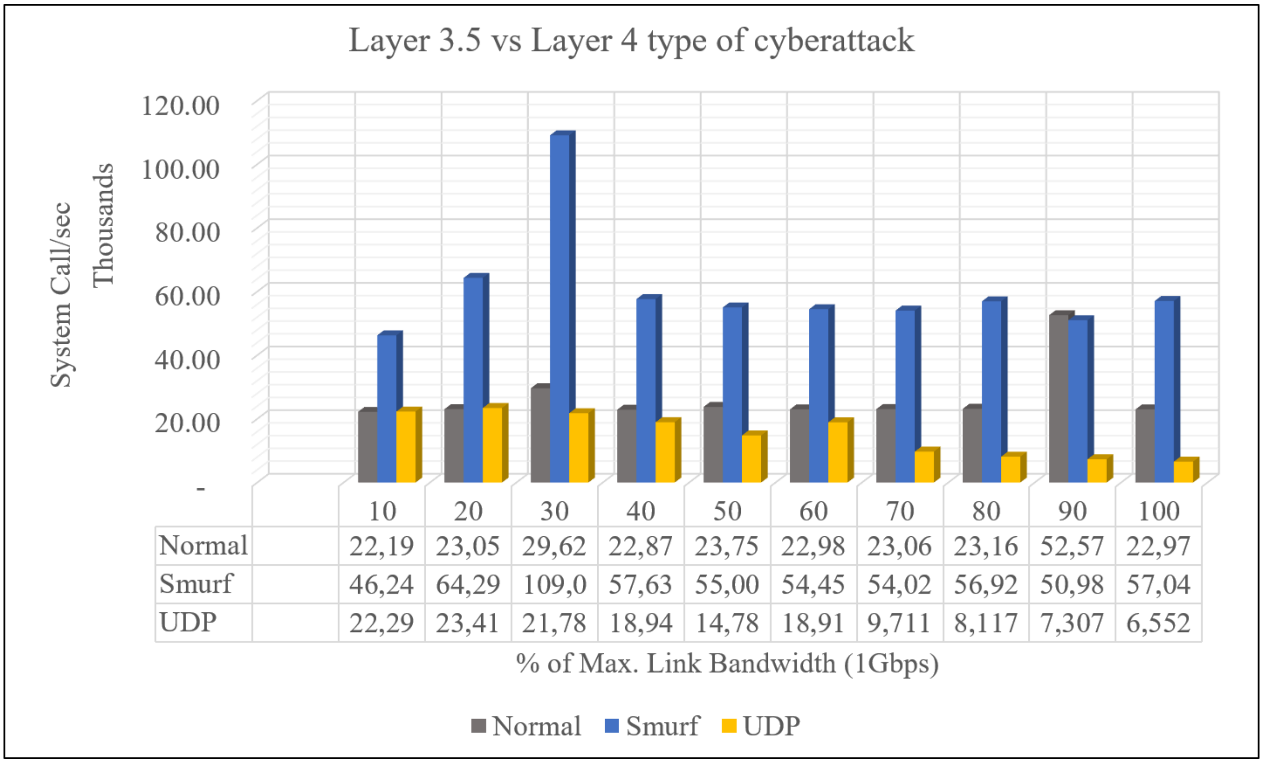


	10	20	30	40	50	60	70	80	90	100
Normal	10.57	10.18	12.36	10.16	11.09	10.31	10.05	10.22	19.37	10.68
Smurf	721.	1,02	915.	1,06	977.	1,06	1,03	1,01	1,01	1,05
UDP	10.0	10.5	9.66	8.83	8.38	10.7	6.43	5.23	5.15	4.21

Figure 5 - 24 System: File Write Operations per second

"File Write Operations/sec is the combined rate of the file system write requests to all devices on the computer, including requests to write to data in the file system cache. It is measured in numbers of writes. This counter displays the difference between the values observed in the last two samples, divided by the duration of the sample interval."

From Figure 5-24 we can observe that the File Write Operations per second is very high for Smurf attack. If we look at the effects displayed by Normal traffic, which is when the computer is receiving legitimate traffic, the 3,000 connections, we can determine that there is a suppression of this type of file write operation activity during Internet activity and much more during UDP attack. During a Smurf attack we can see the activity rises to close to 1500 file operations per second.



	10	20	30	40	50	60	70	80	90	100
Normal	22,19	23,05	29,62	22,87	23,75	22,98	23,06	23,16	52,57	22,97
Smurf	46,24	64,29	109,0	57,63	55,00	54,45	54,02	56,92	50,98	57,04
UDP	22,29	23,41	21,78	18,94	14,78	18,91	9,711	8,117	7,307	6,552

Figure 5 - 25 System: System Call per second

"System Calls/sec is the combined rate of calls to operating system service routines by all processes running on the computer. These routines perform all of the basic scheduling and synchronization of activities on the computer, and provide access to non-graphic devices, memory management, and name space management. This counter displays the difference between the values observed in the last two samples, divided by the duration of the sample interval."

System call per second in Figure 5-25 show how a Smurf attack requested more operating system services than a UDP attack.

5.4 Concluding Remarks

In this chapter we presented an in-depth look at what happens inside a computer when a Layer 4 cyberattack, and a Layer 3.5 cyberattack is launched. As in chapter 3 and chapter 4, here too, we observed seven computer performance objects to get a better understanding of what happens to the computer. Here, we focus on the counters that show activity that could give us a better understanding to compare the results to a Layer 4 type of attack and Layer 3.5 type of attack. The results for a Layer 4 type of attack are presented in chapter 4 and the results for Smurf are presented in chapter 3. Here we compare the results of what we consider the most damaging of the two attacks presented in each chapter. For Cache, we looked at three elements that gave use the most activity: Dirty Pages, Dirty page threshold and Pin Read Hits %. In Logic Disk we looked at % Idle Time, Disk Bytes/sec, Disk Write Bytes/sec and Disk Writes/sec. The Memory object is analyzed using % Committed Bytes In Use, Available Mbytes, and Page Faults/sec. In Physical Disk performance object we looked into % Disk Time, Disk Write Time, % Idle Time object counters. The Process performance object, in our test, is composed of the following performance counters: % Privileged Time, % Processor Time, % User Time, I/O Write Operations/sec, Page Faults/sec. % Idle Time, % Processor Time, and Interrupts/sec are counters selected to represent the activity of the Processor performance object. Lastly, System performance object is analyzed in the paper using the following counters: % Registry Quota In Use, File Write Bytes/sec, and File Write Operations/sec and System Calls/sec.

Our observation of the computer activity using the above-mentioned performance objects during the mentioned cyberattacks give us a very comprehensive account of how a computer's activity is affected when it is attack by commonly known DDoS attacks.

In the book Internet Information Services (IIS) 6.0, it is suggested that to improve a computer's performance the processor needs to be updated. Upgrading the L2 cache helps improve the processing performance. Since the cache is a small piece of memory, finding better algorithms to manage this part of memory more efficiently should improve the processing performance as well. As far as the improvement of servers some suggestions to mitigate against damaging traffic is to use multiprocessor server. Since cyberattacks like Smurf, UDP, Ping, and TCP, work by targeting internet traffic, utilizing http components, the web administrators can use server scripts like ASP, PHP, JASON, among others. Limiting connections, adjusting the number of threads, and controlling the type of traffic of the computer can ensure the availability of data resources to legitimate users [13].

REFERENCES

1. Brant, Tom, et al. “The Best 2-in-1 Convertible and Hybrid Laptops for 2019.” *PCMAG*, 7 Dec. 2018, www.pcmag.com/roundup/346226/the-best-2-in-1-convertible-and-hybrid-laptops.

2. Buzzi, Matthew. “Microsoft Surface Pro 6.” *PCMAG*, 29 Oct. 2018, www.pcmag.com/review/364401/microsoft-surface-pro-6.

3. Osborne, Joe. “Microsoft Surface Pro 4 Review.” *TechRadar*, TechRadar The Source for Tech Buying Advice, 4 Mar. 2019, www.techradar.com/reviews/pc-mac/tablets/microsoft-surface-pro-4-1290285/review.

4. “Intel® Core™ i7-6650U Processor (4M Cache, up to 3.40 GHz) Product Specifications.” *(4M Cache, up to 3.40 GHz) Product Specifications*, ark.intel.com/content/www/us/en/ark/products/91497/intel-core-i7-6650u-processor-4m-cache-up-to-3-40-ghz.html.

5. MAD\ttian. “Software Techniques for Shared-Cache Multi-Core Systems.” *Intel® Software*, Intel, 8 Mar. 2012, software.intel.com/en-us/articles/software-techniques-for-shared-cache-multi-core-systems/?wapkw=smart+cache.

6. Wang, Weixun, et al. “Dynamic Cache Reconfiguration and Partitioning for Energy Optimization in Real-Time Multi-Core Systems.” *Proceedings of the 48th Design Automation Conference on - DAC '11*, 2011, doi:10.1145/2024724.202493

7. Hemmendinger, David. “Operating system.” Encyclopedia Britannica. Encyclopedia Britannica, inc, November 14, 2019, https://www.britannica.com/technology/operating-system Accessed: September 16, 2020.

8. Kreit, Alice, Wanke, Jessica. “Timeline: Bill Gates-From Geek to Gazillionaire to Do-Gooder.” PCMAG, 7 Dec. 2018, https://legacy.npr.org/news/graphics/2008/june/bill_gates/gates_timeline_04.html.

9. Baez, R., Jr. (2015). Evaluation of security vulnerabilities of popular computer and server operating systems under cyber attacks (Order No. 1592483). Available from ProQuest Dissertations & Theses Global. (1706876951). Retrieved from

http://ezhost.utrgv.edu:2048/login?url=https://www.proquest.com/docview/1706876951?accountid=7119

10. Alessandro. (2014, May 13). File System Cache-Dirty Pages Threshold. Retrieved July 14, 2020 from https://smartwindows.wordpress.com/2014/05/13/file-system-cachedirty-pages-threshold/

11. Satran, M. (2018, May 31). File Caching. Retrieved on July 14, 2020 from https://docs.microsoft.com/en-us/windows/win32/fileio/file-caching#:~:text=By%20default%2C%20Windows%20caches%20file,than%20from%20the%20physical%20disk.&text=Caching%20is%20managed%20per%20file%20object.

12. File System Cache-Dirty Pages Threshold. https://smartwindows.wordpress.com/2014/05/13/file-system-cachedirty-pages-threshold/

13. Internet Information Services (IIS) 6.0. Microsoft Press, 2004.

14. Ganesh Gunnam and Sanjeev Kumar, "Do ICMP Security Attacks have same Impact on Servers?" Journal of Information Security, Vol.8, No.3, pp. 274-283, July 2017 (750 Text Views) - citations tracked by Web of Science, Google based Impact Factor = 2.23

15. Koushicaa Sundar and Sanjeev Kumar, "Blue Screen of Death observed for the Microsoft's Server 2012 R2 under Denial of Service Attacks," Journal of Information Security, vol. 7, pp. 225-231, July 2016. (1805 Text views since publication); citations tracked by Web of Science, Google based Impact Factor = 2.23

16. Rodolfo Baez Jr., Sanjeev Kumar, "Apple's Lion Vs. Microsoft's Windows 7: Comparing Built-In Protection against ICMP Flood Attacks," Journal of Information Security, vol. 5, no.3, pp. 123-135, July 2014, 4807 Text views; citations tracked by Web of Science, Google Scholar, Impact Factor = 2.23

BIOGRAPHICAL SKETCH

Adrian Guerrero was born on November 4, 1979 to immigrant parents from Mexico. At the age of two, him and family moved to Mexico where he learned to raise all kinds of farm animals and where he and his siblings learned the meaning of contributing to a family through hard work. In 1993 he moved to Waco, Texas, where he graduated from La Vega High School in 1999.

In 2004 he graduated from McLennan Community College with an Associates in Computer Programming. During those years he was part of the Internationals Students Club and learned to play basic guitar. In 2007 he moved to Edinburg, Texas, where he received his bachelor's degree in Computer Engineering from The University of Texas – Pan American (UTPA). One of the most meaningful experiences was when he was awarded The Cultural Immersion Program – China Scholarship with which he was able to travel and stay in China for one month in June 2009 and June 2010. Adrian received his master's degree in Electrical Engineering in 2020 and considers it a lifetime achievement.

Permanent Address:

4315 Harrison St, Waco, Texas 76705

Publications:

[1] Microsoft Surface Pro 4 Performance Under Denial of Service

The 2nd International Conference on Data Intelligence and Security, June 28-30, 2018