# Part A

As a computer science student with no prior experience in cybersecurity, I am well aware of the growing shortage of security personnel worldwide. This shortage can be attributed to a number of factors, including the rapid pace of technological innovation, the increasing number of cyberattacks, and the lack of qualified candidates.

Because of the rapid pace of technological advancement, the amount of sensitive data saved and communicated online has increased. As a result, there is a growing demand for cybersecurity specialists who can protect people and businesses from online threats. There is a shortage of security professionals because the number of qualified candidates has not kept up with the demand.

Furthermore, the growing number of cyberattacks has highlighted the importance of putting in place a strong and effective security infrastructure. This has increased the demand for security personnel, making it more difficult for businesses to find qualified candidates.

There are numerous steps and actions that can be taken to solve this problem. One of the most important tasks is to invest in education and training programs to develop more trained security professionals. This can be accomplished by creating specialized cybersecurity programs at colleges and other educational institutions, as well as training programs for current IT workers who want to transition to the cybersecurity industry.

Another important step is to offer competitive compensation and benefits to attract and retain top talent. This will make it more appealing for individuals to pursue a career in cybersecurity and will help to ensure that the best and brightest minds are working to protect our online assets.

Because I am a computer science student with no prior experience in cybersecurity, I lack the knowledge and expertise required to be regarded as a security expert. However, I have a strong background in computer science and programming, which will be useful as I learn more about security. If I want to become a security expert, I will need to invest in additional education and training in the subject, as well as gain practical experience through internships and other opportunities. In addition, I'll need to stay current with industry developments and be open to lifelong learning and development.

In conclusion, the shortage of security personnel worldwide is a serious concern that needs to be addressed. Investing in education and training, offering competitive compensation, and encouraging more people to pursue careers in security are all important steps that can help to address this shortage and ensure that we have the skills and expertise necessary to protect our online assets.

# Part B

Code used to test:

function userAction() {

    let searchString = document.getElementById("myInput").value;

    let txt = "";

    console.log(searchString);

    let webhook\_url =

        "https://eu-west-1.aws.data.mongodb-api.com/app/fyp-bffpf/endpoint/NameSearchWebhook";

    let url = webhook\_url + "?arg=" + searchString;

    fetch(url)

        .then(function (response) {

            return response.json();

        })

        .then(function (data) {

            document.getElementById("results").innerHTML =

                "<pre>" +

                JSON.stringify(data, undefined, 2)

                    .replace(/[&\\\#,+()$~%.'"\*?<>{}]/g, "")

                    .replace(/[\[\]']+/g, "") +

                "</pre>";

            console.log(data);

            if (data.length == 0) {

                document.getElementById("results").innerHTML = "No results";

            }

        })

        .catch(function (err) {

            console.log(err);

        });

    document.getElementById("myInput").value = "";

JSLint Results:

Report: Warnings (11)

*2: 24*

*1. Undeclared 'document'.* let searchString = document.getElementById("myInput").value;

*4: 5*

*2. Undeclared 'console'.* console.log(searchString);

*7: 1*

*3. Line is longer than 80 characters.* "https://eu-west-1.aws.data.mongodb-api.com/app/fyp-bffpf/endpoint/NameSearchWebhook";

*11: 5*

*4. Undeclared 'fetch'.* fetch(url)

*16: 13*

*5. Undeclared 'document'.* document.getElementById("results").innerHTML =

*19: 36*

*6. Unexpected '\' before '#'.* .replace(/[&\\\#,+()$~%.'"\*?<>{}]/g, "")

*22: 13*

*7. Undeclared 'console'.* console.log(data);

*23: 29*

*8. Expected '===' and instead saw '=='.* if (data.length == 0) {

*24: 17*

*9. Undeclared 'document'.* document.getElementById("results").innerHTML = "No results";

*28: 13*

*10. Undeclared 'console'.* console.log(err);

*30: 5*

*11. Undeclared 'document'.* document.getElementById("myInput").value = "";

Report: Properties (10)

/\*property

catch, getElementById, innerHTML, json, length, log, replace, stringify,

then, value

\*/

Report: Functions (4)

global

JSON, userAction

*1: 1*

*userAction()*

variable

searchString, txt, url, webhook\_url

*12: 15*

*«then»(response)*

parameter

response

*15: 15*

*«then»(data)*

parameter

data

global

JSON

*27: 16*

*«catch»(err)*

parameter

err

JSHint Results:

Metrics

There are **4** functions in this file.

Function with the largest signature take **1** arguments, while the median is **1**.

Largest function has **7** statements in it, while the median is **2.5**.

The most complex function has a cyclomatic complexity value of **2** while the median is **1**.

|  |  |
| --- | --- |
| Four warnings | |
| 14 | 'let' is available in ES6 (use 'esversion: 6') or Mozilla JS extensions (use moz). |
| 15 | 'let' is available in ES6 (use 'esversion: 6') or Mozilla JS extensions (use moz). |
| 18 | 'let' is available in ES6 (use 'esversion: 6') or Mozilla JS extensions (use moz). |
| 21 | 'let' is available in ES6 (use 'esversion: 6') or Mozilla JS extensions (use moz). |

|  |
| --- |
|  |

|  |  |
| --- | --- |
| Two unused variables | |
| 15 | txt |
| 13 | userAction |

# Part C

Basic Operators:

1. +

* Usage: chocolate +cake
* Explanation: The plus sign is used to include search terms in the results. In this example, the search results will include pages that mention both "chocolate" and "cake".

1. \*

* Usage: best \* recipe
* Explanation: The asterisk is used as a wildcard to represent any word or phrase. In this example, the search results will include pages that contain "best" and "recipe", with any word or phrase in between.

1. -

* Usage: pizza -dominos
* Explanation: The minus sign is used to exclude a search term from the results. In this example, the search results will include pages that mention "pizza" but not "dominos".

1. ""

* Usage: "best pizza in new york"
* Explanation: Quotation marks are used to search for an exact phrase. In this example, the search results will include pages that contain the exact phrase "best pizza in new york".

1. ~

* Usage: ~coffee
* Explanation: The tilde is used to include synonyms of a search term in the results. In this example, the search results will include pages that mention "coffee" as well as its synonyms, such as "java" and "brew".

1. |

* Usage: pizza | pasta
* Explanation: The vertical bar is used to search for pages that include either of two terms. In this example, the search results will include pages that mention either "pizza" or "pasta".

1. .

* Usage: how to make a pizza.italian style
* Explanation: The period is used as a placeholder for a single word in a phrase. In this example, the search results will include pages that contain the phrase "how to make a pizza" followed by a single word, which in this case is "italian style".

1. OR

* Usage: pizza OR pasta
* Explanation: The OR operator is used to search for pages that include either of two terms. In this example, the search results will include pages that mention either "pizza" or "pasta". Note that OR must be capitalized in order to function as an operator in Google search.

Advanced Operators:

1. allintext

* Usage alone: allintext:"apple banana"
* Usage in combination with another operator: allintext:"apple banana" site:nytimes.com

1. define

* Usage alone: define:algorithm
* Usage in combination with another operator: define:algorithm site:wikipedia.org

1. intitle

* Usage alone: intitle:"best books 2022"
* Usage in combination with another operator: intitle:"best books 2022" site:goodreads.com

1. site

* Usage alone: site:wikipedia.org
* Usage in combination with another operator: "artificial intelligence" site:wikipedia.org

1. allintitle

* Usage alone: allintitle:"best movies 2021"
* Usage in combination with another operator: allintitle:"best movies 2021" site:imdb.com

1. filetype

* Usage alone: filetype:pdf
* Usage in combination with another operator: "artificial intelligence" filetype:pdf site:ieee.org

1. inurl

* Usage alone: inurl:login
* Usage in combination with another operator: inurl:login site:facebook.com

1. numrange

* Usage alone: numrange:100-200
* Usage in combination with another operator: numrange:100-200 site:amazon.com

1. allinurl

* Usage alone: allinurl:"google search operators"
* Usage in combination with another operator: allinurl:"google search operators" site:moz.com

1. info

* Usage alone: info:nytimes.com
* Usage in combination with another operator: info:nytimes.com site:wikipedia.org

1. link

* Usage alone: link:nytimes.com
* Usage in combination with another operator: link:nytimes.com site:washingtonpost.com

1. daterange

* Usage alone: daterange:2459493-2459497
* Usage in combination with another operator: "climate change" site:nytimes.com daterange:20220101-20220201

1. cache

* Usage alone: cache:nytimes.com
* Usage in combination with another operator: cache:nytimes.com/best-sellers site:amazon.com

1. intext

* Usage alone: intext:"best chocolate cake recipe"
* Usage in combination with another operator: site:allrecipes.com intext:"best chocolate cake recipe"

1. related

* Usage alone: related:nytimes.com
* Usage in combination with another operator: related:nytimes.com site:wikipedia.org

Many of the operators listed above for Google search also work in Bing.

1. site: - Same in Bing.
2. filetype: - Same in Bing.
3. inurl: - Same in Bing.
4. intitle: - Same in Bing.
5. intext: - Same in Bing.
6. allinurl: - Same in Bing.
7. allintitle: - Same in Bing.
8. allintext: - Same in Bing.
9. link: - Same in Bing.
10. define: - Same in Bing.
11. related: - Not applicable to Bing.
12. numrange: - Not applicable to Bing.
13. cache: - Not applicable to Bing.
14. info: - Not applicable to Bing.
15. OR - Same in Bing.
16. + - Same in Bing.
17. \* - Same in Bing.
18. - - Same in Bing.
19. “” - Not applicable to Bing.
20. ~ - Not applicable to Bing.
21. | - Not applicable to Bing.
22. . - Not applicable to Bing.

Ten new search engines giving their advantage(s) or disadvantages over Google or Bing

1. DuckDuckGo

Advantages:

* Focuses on privacy, does not track or store user data

1. Yahoo!

Advantages:

* Large index, similar to Google's
* Provides access to Yahoo Mail, Yahoo Finance, Yahoo News, and other Yahoo services

1. Ask Jeeves

Advantages:

* Emphasizes natural language queries, allowing users to ask questions in plain English

Disadvantages:

* Not as widely used as Google or Bing, may not have as comprehensive results

1. WolframAlpha

Advantages:

* Uses sophisticated algorithms to generate in-depth reports and analyses

Disadvantages:

* May not be as useful for everyday web searches

1. Baidu

Advantages:

* Primarily used in China, with a large index of Chinese-language web pages

Disadvantages:

* May be subject to Chinese government censorship and restrictions

1. Yandex

Advantages:

* Large index of Russian-language websites

Disadvantages:

* May not be as useful for non-Russian speakers

1. CC Search

Advantages:

* Focuses on searching Creative Commons-licensed media, including images and music

Disadvantages:

* May not be as useful for general web searches

1. Dogpile

Advantages:

* Aggregates results from multiple search engines, including Google, Yahoo, and Bing

Disadvantages:

* May return redundant or low-quality results

1. Qwant

Advantages:

* Focuses on user privacy, does not track or store user data

Disadvantages:

* May not return as comprehensive or relevant results

1. Ecosia

Advantages:

* Focuses on sustainability, uses ad revenue to plant trees

Disadvantages:

* May not return as comprehensive or relevant results

# Part D

Examples of 10 vulnerabilities:

1. Man-in-the-middle (MITM) attack: An attacker intercepts communication between two parties, potentially stealing sensitive information or modifying the contents of the communication.
2. Denial-of-service (DoS) attack: An attacker floods a network with traffic or requests, causing the network to become unresponsive or crash.
3. SSL/TLS Vulnerabilities: Attackers can exploit vulnerabilities in SSL/TLS protocols to intercept or modify encrypted communication.
4. SMTP Vulnerabilities: Attackers can exploit vulnerabilities in Simple Mail Transfer Protocol (SMTP) to intercept or modify email communication.
5. FTP Vulnerabilities: Attackers can exploit vulnerabilities in File Transfer Protocol (FTP) to steal data or execute arbitrary code on the server.
6. Injection Flaws: This type of vulnerability can occur when an application fails to properly validate or sanitize user input. Attackers can then inject malicious code into the application, which can lead to unauthorized access to sensitive data or the ability to execute arbitrary code.
7. Cross-site Scripting: This vulnerability occurs when an attacker is able to inject malicious code into a website, which is then executed by other users who visit the site. This can allow the attacker to steal sensitive information, such as login credentials or personal data.
8. Broken Authentication and Session Management: This vulnerability can occur when an application fails to properly authenticate users or manage user sessions. Attackers can then steal user credentials, hijack sessions, and gain unauthorized access to sensitive data.
9. Security Misconfiguration: This vulnerability occurs when an application is configured in an insecure manner, such as using default passwords, leaving unnecessary ports open, or failing to apply security updates. Attackers can then exploit these vulnerabilities to gain access to sensitive data or execute arbitrary code.
10. Improper Error Handling: This vulnerability can occur when an application fails to properly handle errors or exceptions. Attackers can then exploit these errors to gain unauthorized access to sensitive data or execute arbitrary code.