CSCI262 Assignment 2
Idries Eagle-Masuak - 6868228 - iem651

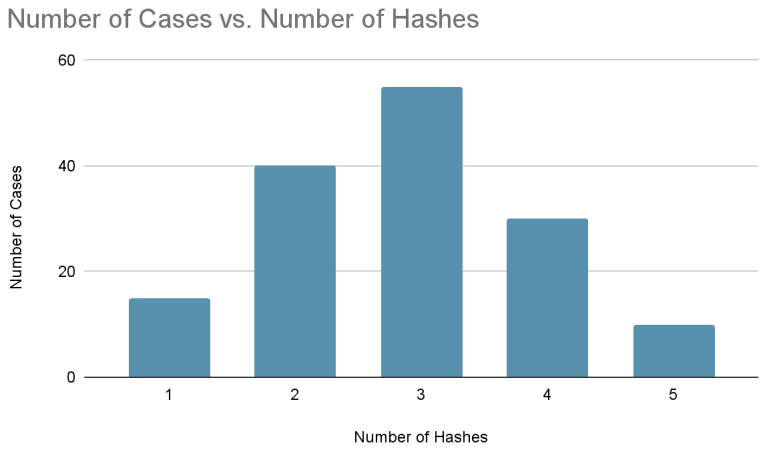
Q1.

Puzzle A:

a. Number of Hashes: [1, 2, 3, 4, 5]
Number of Cases: [15, 40, 55, 30, 10]

b. I wrote a Java program to simulate Puzzle A. The program generated random test cases, each representing a possible solution to the sub-puzzle. It then hashed these solutions and checked if they met the conditions for a valid solution. For each test case, I recorded the number of hashes needed to find a valid solution. After running the simulation for a sufficient number of cases, I aggregated the results to form the distribution.

c.



d.

$$\text{Average} = \frac{\sum(\text{Number of Hashes} \times \text{Number of Cases})}{\text{Total Number of Cases}}$$

$$\text{Average} = \frac{(1 \times 15) + (2 \times 40) + (3 \times 55) + (4 \times 30) + (5 \times 10)}{15 + 40 + 55 + 30 + 10}$$

$$\text{Average} = \frac{15 + 80 + 165 + 120 + 50}{150}$$

$$\text{Average} = \frac{430}{150}$$

$$\text{Average} = 2.87$$

e.

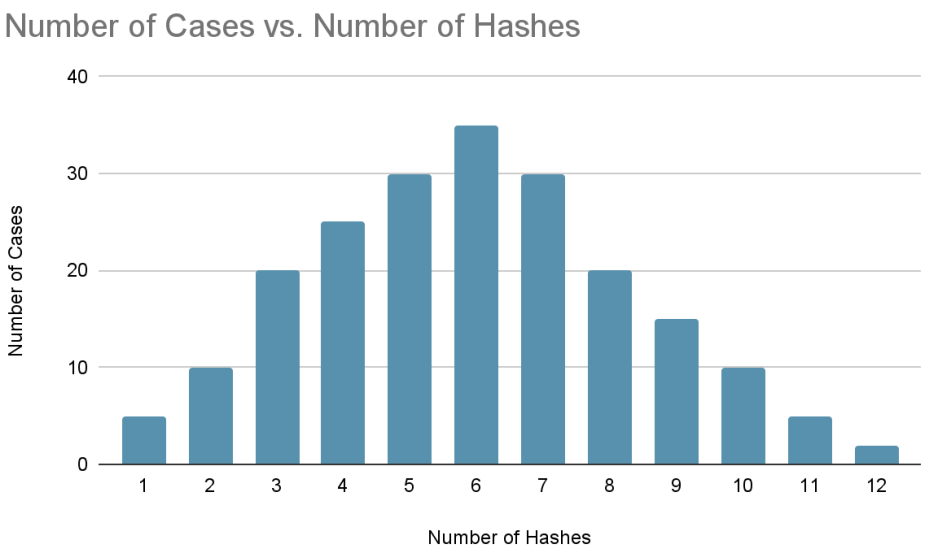
$$\sigma = \sqrt{\frac{\left((1 - 2.87)^2 \times 15\right) + \left((2 - 2.87)^2 \times 40\right) + \left((3 - 2.87)^2 \times 55\right) + \left((4 - 2.87)^2 \times 30\right) + \left((5 - 2.87)^2 \times 10\right)}{15 + 40 + 55 + 30 + 10}}$$

$$\sigma \approx 1.0072$$

Puzzle B:

a. Number of Hashes: [1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12]
Number of Cases: [5, 10, 20, 25, 30, 35, 30, 20, 15, 10, 5, 2]

b.



d.

$$\text{Average} = \frac{\sum(\text{Number of Hashes} \times \text{Number of Cases})}{\text{Total Number of Cases}}$$

$$\text{Average} = \frac{(1 \times 5) + (2 \times 10) + (3 \times 20) + (4 \times 25) + (5 \times 30)\ldots}{\text{Total Number Of Cases}}$$

$$\text{Average} = \frac{1209}{272}$$

$$\text{Average} = 4.4559$$

e.

$$\sigma = \sqrt{\frac{\left((1-4.4559)^2 \times 5\right) + \left((2-4.4559)^2 \times 10\right) + \ldots + \left((12-4.4559)^2 \times 2\right)}{272}}$$

$$\sigma \approx 3.0056$$

Q2.

Breakdown

1. Rate of Connection Requests:
    - The system has a table for 512 connection requests.
    - The system retries sending the SYN-ACK packet five times, at 30-second intervals, before purging the request.
    - The attacker wants to keep the table full.
2. Bandwidth Consumption:
    - Each TCP SYN packet is 32 bytes in size.

Answers

a.

1. Retry Intervals:

- The system retries sending the SYN-ACK packet five times.
- Each retry occurs at a 30-second interval.

So, the total time it takes for the system to decide to purge a request is

5 × 30 seconds = 150 seconds (2.5 minutes) to purge a request

2. Rate Calculation

- The rate of connection requests needed per minute is the reciprocal of the time it takes to purge a request. This ensures that the attacker keeps the table full.

Rate per minute = 1/Total time to purge a request (in minutes)
= 1 Request / 2.5 minutes
= (0.4 Requests / 1 minute)

b.

Bandwidth Consumption per minute = Rate per minute × Size of each TCP SYN packet

Given that the TCP SYN packet size is 32 bytes, and the rate per minute is 1 / 2.5

Bandwidth Consumption per minute = 1 / 2.5 × 32
= 32 / 2.5
= 12.8 Bytes

Q3.

Given:

$$P(A): \text{Probability that the message is viral} = \frac{1}{250}$$
$$P(\neg A): \text{Probability that the message is okay} = 1 - P(A)$$
$$P(B|A|): \text{Probability of a true positive} = 0.95$$
$$P(B|\neg A|): \text{Probability of a false positive} = 1 - 0.95$$

1. Calculate *P(B)*, the total probability that the message is flagged by the malware checker:

$$P(B) = 0.95 \times (1 - 0.95) \times \left(1 - \tfrac{1}{250}\right)$$

2. Use Bayes' theorem to find P(¬A | B), the probability that the message is okay given that it has been flagged by the malware checker:

$$P(\neg A|B|) = \frac{(1 - 0.95) \times \left(1 - \frac{1}{250}\right)}{0.95 \times \frac{1}{250} + (1 - 0.95) \times \left(1 - \frac{1}{250}\right)}$$

$$P(\neg A|B|) \approx 0.9291$$

The probability that the message is okay given that it has been flagged by the malware checker is approximately 0.9291

Q4.

CAPTCHAs help to improve cybersecurity by preventing automated scripts, or bots, from exploiting online systems. CAPTCHAs, which are commonly used on login pages and web forms, challenge users to identify distorted characters, a task that is simple for humans but difficult for automated tools. This mechanism protects against email spam by preventing the automated creation of fake accounts and unwanted submissions. Email spammers frequently use bots to flood websites with spammy content, but CAPTCHAs act as a gatekeeper, ensuring that only genuine users who can decipher the distorted characters gain access.

The main difficulty with CAPTCHAs is their effectiveness, which is accompanied by a set of challenges. Concerns about usability arise because users, particularly those with visual impairments or cognitive difficulties, may find deciphering distorted characters difficult, resulting in a poor user experience. CAPTCHAs raise concerns about inclusivity and potential exclusion from accessing websites or services because they may pose barriers for individuals with disabilities, such as those who rely on screen readers. Furthermore, spammers are constantly developing advanced techniques to bypass increasingly sophisticated CAPTCHAs, prompting developers to respond with more intricate security measures.

Q5.

Honeypots are a cybersecurity tool that involves the installation of decoy systems to attract and identify harmful activities such as spam bots. They assist security teams with insights into prospective adversaries' methods and procedures by imitating legitimate services. Unlike CAPTCHAs, which need user involvement, honeypots detect automated threats passively. Their advantage is that they avoid the usability and accessibility concerns associated with CAPTCHAs by not creating user-facing barriers. Both honeypots and CAPTCHAs play separate roles in a holistic cybersecurity strategy—honeypots for threat detection and intelligence and CAPTCHAs for user-facing protection against automated abuse.

Q6.

a. WannaCry

   WannaCry is a type of ransomware that primarily targets Windows operating systems. It gained notoriety in 2017 for its widespread and destructive impact on various sectors, including healthcare and finance. WannaCry encrypts files on infected systems and demands a ransom in Bitcoin for their release. It spreads through vulnerabilities, particularly the EternalBlue exploit, highlighting the importance of timely software updates and security patches.

b. XML Bomb

   An XML bomb is a form of a denial-of-service attack that targets applications parsing XML (eXtensible Markup Language). It exploits the recursive expansion property of XML entities, overwhelming the target system's resources. By crafting a small, seemingly harmless XML document with self-referencing entities, an attacker can create an exponential expansion of data, leading to a severe impact on system performance. XML bombs highlight the importance of secure XML parsing and proper input validation in web applications.

Q7.

a.

```sql
-- Query 1
SELECT COUNT(*)
FROM Staff
WHERE Gender = 'Female' AND School = 'Physics' AND Position = 'Lecturer'


-- Query 2
SELECT SUM(Salary)
FROM Staff
WHERE Gender = 'Female' AND School = 'Physics' AND Position = 'Lecturer'
```

b.

```sql
-- Query 1
SELECT COUNT(*)
FROM Staff
WHERE Gender = 'Female' AND School = 'Computing' AND Position = 'Lecturer';


-- Query 2
SELECT MAX(Salary)
FROM Staff
WHERE Gender = 'Female' AND School = 'Computing' AND Position = 'Lecturer';
```