Section 1/2

Instructor: Dr Shazi Shah Jabeen / Dr Devika Sharma

Group Members

S No	Name	ID No
1	Hussain Mithi	2020A4PS0049U
2	Rishin Shah	2020A9PS0007U
3	Marshal	2020A4PS0005U
4	Rohan Pandit	2020A9PS0199U
5	Harsh	2020A7PS0215U
6	Ahmed Thahir	2020A7PS0198U

Google LLC

The United States of America

A Report

On

The Problems of the Web and How to Fix them

Prepared For

Dr. John Mathews

Head

Web Experience

By

Research Team

Web Experience

Approved By

Dr. Josh Caleb

Head Researcher

Web Experience

11 December 2020

ABSTRACT

The report primarily focuses on improving the speed and security of the data that is transferred over the web. Speed and security are two of the most important components that ensure user satisfaction and user loyalty; two crucial aspects that determine the success of any website, whether it be social media platforms or e-commerce websites. The report also focuses on improving the “end-user experience”, and provides strategies and techniques to utilize current technologies. The report focuses on server placement, loading techniques, simple algorithms like Huffman Coding, compression of different kinds of data on the web, and encryption techniques to enhance cyber-security. The report also proposes recommendations to improve the user experience which new developers as well as regular users can adopt with ease.

ACKNOWLEDGEMENTS

We would like to thank Dr. Josh Caleb, Head Researcher, Web Experience for his guidance and constant supervision.

We would also like to thank our colleagues who supported us in collecting sufficient information to complete this report.

Table of Contents

LIST OF ILLUSTRATIONS

CHAPTER 1: INTRODUCTION

1.1 Authorisation

This report on 'The Problems of the Web and How to Fix them' has been authorized by Dr. Dan Williams, Head of Research, at Google LLC. on 11 December 2020.

1.2 Historical Background

Speed and security have always played a vital role when it comes to the transfer of data on the web. Speed of loading on the web is one of the biggest issues with the web currently, with a major fraction of users dissatisfied with their experience due to slow load times. Furthermore, web security is a field which has come to everyone's attention with the rapid increase of data in today's internet age. Cybercrimes have increased significantly and sensitive data such as ATM credentials, bank details etc, are always at risk of being stolen.

1.3 Objectives

- To ascertain the importance of fast and safe web experience
- To analyze the various technologies and algorithms to improve web experience
- To recommend appropriate measures to both developers as well as users to achieve great web experience

1.4 Scope

- The report covers an outline of CDN, lazy loading technique, basic data compression and security algorithms.
- It is enough to mention milestones since there are many trivial details which are not related to the subject.

1.5 Limitations

- The report only focuses on the above basic concepts, rather than related techniques as well as other aspects that factor in during their implementation.

- GZIP and Advanced algorithms such as Transform coding, Discrete Wavelet Transform, etc. have not been explained in detail, as their working is not relevant for this report.
- Lossless video compression has not been discussed due to limited application on the web.
- Cookies have not been discussed due to time constraints.
- The recommendations provided are limited to a basic level to ensure that even new developers can implement them, and users can effectively adopt them.

1.6 Methods and Sources of Data Collection

The main methods of collecting data were online videos, articles, periodicals, and *Understanding Compression: Data compression for Modern Developers* by Aleks Haecky and Colt McAnlis[7].

1.7 Report Review

Including the introduction, the report is divided into four chapters. Chapter-2 focuses on the importance of speed and cyber-security. Chapter-3 contains the conclusion which sums up the discussions and outlines the major issues faced. Chapter-4 focuses on recommendations and suggestions in order to solve the issue at hand.

CHAPTER 2: DISCUSSION

Speed and Security

Speed plays a vital role in ensuring a good user experience. As the amount of data in the current age has greatly increased, certain strategies must be adopted in order to reduce the time taken for transfer. Google's research showed that most users will not wait more than a few seconds before leaving a website - users were 32% more likely to leave the website when the load time increased by just 2 seconds. Furthermore, the number of visitors increased by 9% when Yahoo! reduced load times by just half a second. [1] These datasets show the importance of speed.

Another issue is security vulnerabilities which may allow user data to be read by a hacker, compromising the user's credentials. This can lead to identity theft, tracking, as well as exploitation of sensitive information.

Therefore, the time taken to transfer data should be greatly reduced along with the improvement of security protocols. In the background of the above discussion we have recommended strategies such as CDNs, lazy loading, data compression, and web encryption, to improve the speed of data transfer and to enhance cyber security.

2. 1. CDN

A CDN (Content Distribution Network) is a network of servers working together to provide fast internet experience to users. These servers are placed at different geographic locations, and deliver content to users in its proximity. The content stored in the main server, also called the 'origin server' is 'distributed' to these CDN servers located in different geographic locations. A CDN allows for the quick transfer of assets[1] needed for loading Internet content, such as images and videos. [2]

The main purpose of CDN is to cater to the users from different parts of the world when the 'Origin Servers' are present only in a few particular

[1] Any form of data requested by user

location(s). In a CDN system, the required content is provided from the nearest CDN server when a user requests for that particular content. Hence, a CDN system reduces delay experienced by the user, the load[2] experienced by the main server to provide the content, increases uptime [3]of the main server, and increases security as users are not directly communicating with the main server.

For example, if a server is located in the UAE, users from distant locations such as Australia will experience a delay, without a CDN system in place. However, if there are CDN servers present in each country, there will be minimal delay and the content will be provided to the users immediately. Moreover, the main server in the UAE will have a lower load as a majority of users will be availing their content from the nearest CDN server.

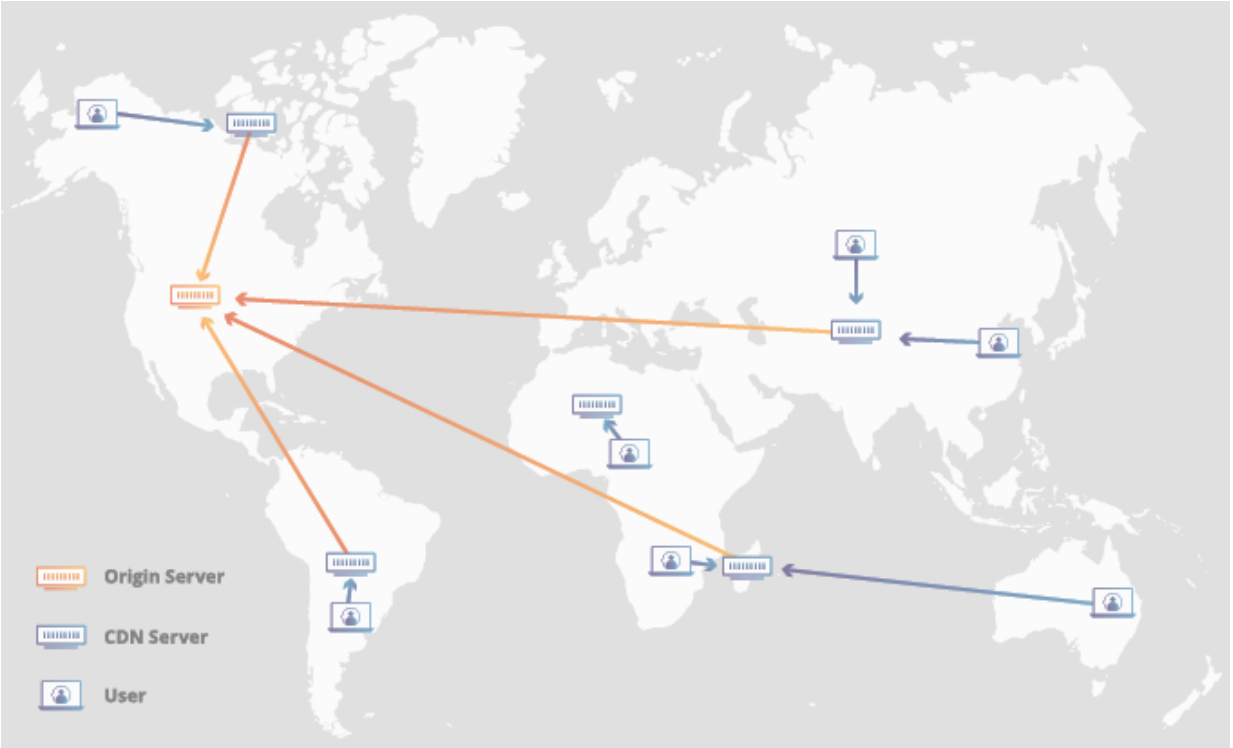

Figure-2.1 CDN network 'distributing' assets [2]

[2] Number of requests
[3] Time for which the server remains accessible by users

2. 2. Lazy Loading

Lazy loading is an optimisation technique to delay loading of content until the point at which it is needed. It can greatly improve web load times if used appropriately.[3] Ecommerce websites rely on lazy loading heavily for their loading, as there are a large number of images to be displayed for various products. This particular approach helps in improving the user's experience by ensuring user engagement wherein, the user will stay on the website and buy the required goods without leaving the website due to slow loading times.

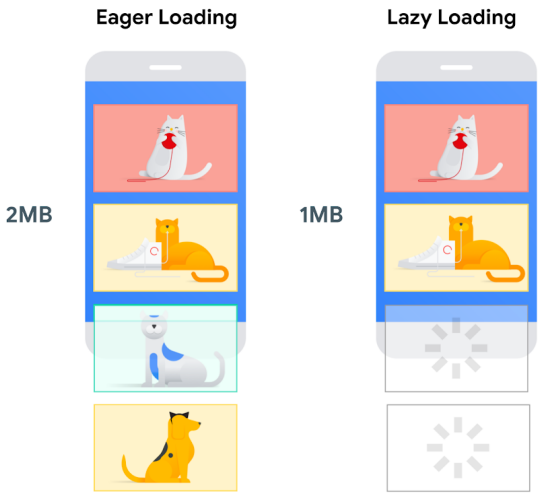

Figure-2.2 Lazy loading non-critical assets [3]

2. 3. Data Compression

Data Compression allows large amounts of data to be transferred over long distances instantly and efficiently by reducing the size, only with acceptable loss in quality. It works based on algorithms which are a set of processes that provide the framework for any data compression technique.

The steps involved in data compression are as follows:

1. Capture device: a device such as a camera, microphone etc. that captures data from the surrounding and helps in transfer of the data to the processor where it undergoes further processing and finally compression.

2. Compression: Various algorithms are used to reduce the file size. It makes storing the data easier within the memory of a device and increases the efficiency of the storage process.
3. Storage Device: The compressed data is then stored in the memory of electronic devices. This data can be accessed when required.
4. Decompression: The compressed data must be decompressed to be used. The data is made available for the user. This enables the user to extract meaningful information from the data displayed.

The most common forms of data on the web are text, audio, image, and video. Compression helps provide users with the required data quicker. Data compression methods can be broadly classified into two: Lossless and Lossy.

Parameter	Lossless Compression	Lossy Compression
Data Loss	No data is thrown away - data is rearranged in a way that reduces size	Data is thrown away to minify size
Savings	Usually minimal savings in file size.	Large savings in file size.
Quality	Identical to original file	Reduced
Compression/ Decompression Duration	High	Low
Restoration of original file	Possible	Not possible

Table-2.3 Contrast between Lossless and Lossy compression

2. 3. 1. Text Compression

Text compression aims at reducing the amount of data required to store text. A character of text usually occupies 8 bits. However, this method can usually be wasteful as unnecessary bits are stored. Only lossless compression techniques are adopted for text compression, ensuring no character is lost. The most common algorithms used are LZ77, Huffman Coding, and DEFLATE.

LZ77

Replaces repeated occurrences with references. A reference contains the distance from the first to the current occurrence of the character(s), and the length of the occurrence. The final compressed file contains characters and the references. In the following example, " FILE IS " is repeated twice, so the 2nd occurrence is replaced by <33, 9>. [5]

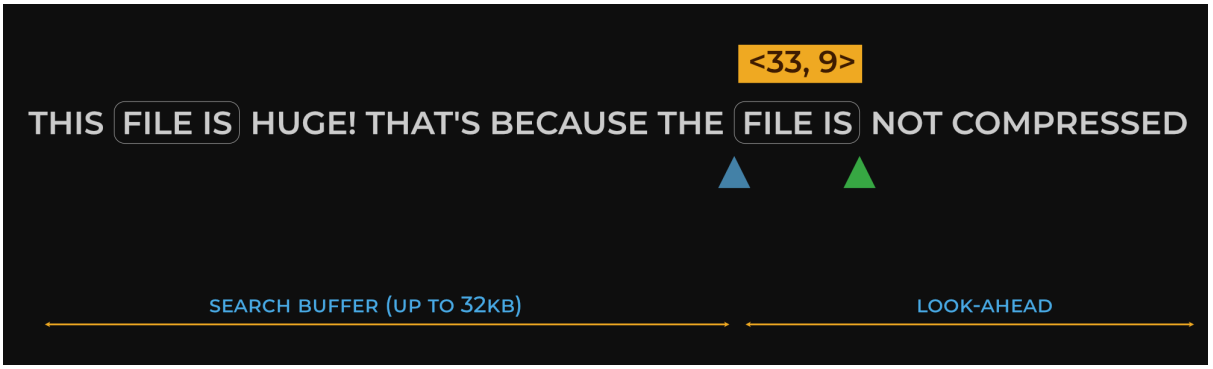

Figure-2.3.1.1 Visual Representation of LZ77 [5]

Huffman Coding

We can lower the number of bits[4] (and hence, the amount of data to be transferred) by intelligently assigning bits to characters. First, the frequency of each character is calculated, and then put in a list in order. Then take the 2 bottom branches - the 2 least used characters, write them down and write their frequency along with them. Then connect them together one level up with the sum of their frequencies. [6]

[4] The most basic unit of data on electronic devices

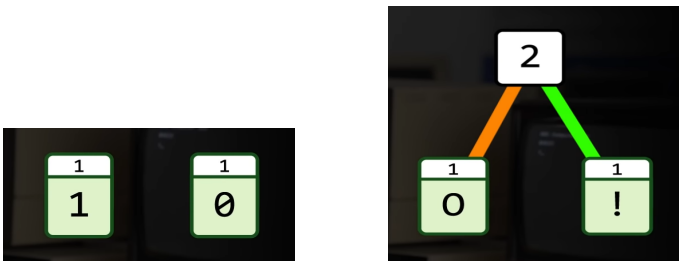

Figure-2.3.1.2 Adding frequencies of characters [6]

Now, add the new sum higher in the list - wherever it sits - and repeat. Eventually when we come across one of the sums, then add up the sum with the frequency of the character (just as if the sum was the frequency of a character). Finally, we'll end up with a Huffman tree and it tells how to convert the text into 1s and 0s. Each time you take the left hand side of the tree, write a zero; and 1 for the right hand side. The Huffman tree is then stored along with the compressed text, to provide translation between the new compressed representation and the uncompressed text.

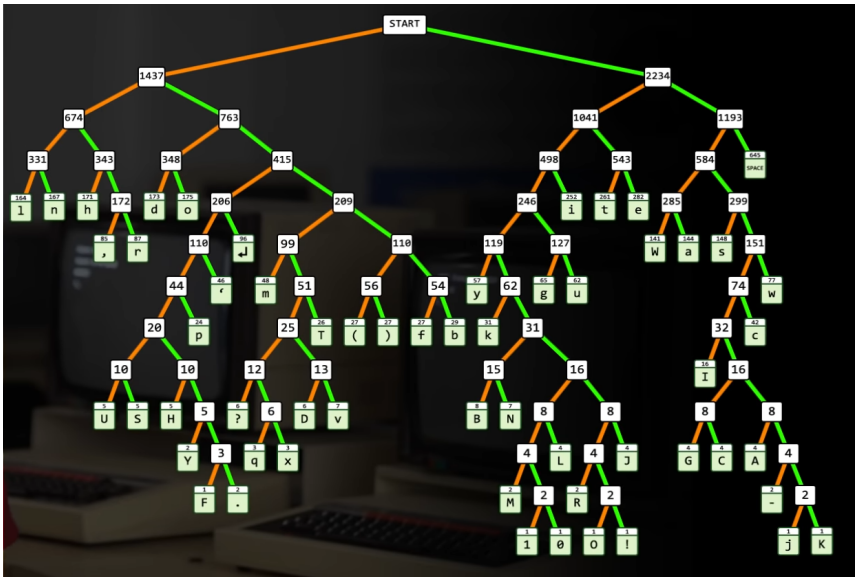

Figure-2.3.1.3 Example of a Huffman Tree [6]

During decompression, take the left branch everytime you see a 0, and the right branch everytime you see a 1. When you eventually end up on a character instead of the sum, that's the required character. For example, W is: 01010111 according to ASCII, but using the Huffman Tree algorithm, W is only 11000, which is a reduction by 3 bits.

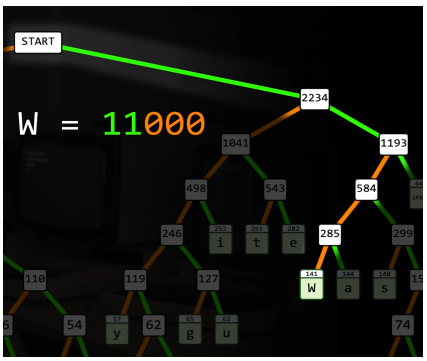

Figure-6 Example of Decompression [6]

DEFLATE

Deflate algorithm involves the use of Huffman Coding algorithm to compress the output from LZ77 algorithm. Data is compressed in blocks, and each block is compressed separately, making it very effective. [5]

Figure-2.3 Flowchart of DEFLATE compression.

2. 3. 2. Audio Compression

Audio compression aims at reducing bitrate[5] used to transmit and store audio. It is carried out with the help of algorithms based on perception of what humans hear. Audio compression generally involves two types of compression: Lossless and Lossy.

Lossless compression

In lossless compression, audio data is compressed in such a way that none of it is eliminated in order to reduce the memory required to store the audio data. It is simply based on tagging redundant audio data with specific bits to signify its repetitions. The original audio data compressed using this method can be retrieved at any instant. The FLAC format uses this type of compression and helps to preserve the entire data and provides the highest quality of audio when decompressed.

[5] The amount of data in bits

Lossy compression

Lossy audio compression involves elimination of certain audio data, which humans cannot perceive (perceptual coding - comes from a field of study called psychoacoustics[6]). It provides excellent file size but the original audio data is lost and can never be recovered. Distortion[7] and clipping[8] are the most common artifacts in lossy compressed audio. The following formats are commonly used on the web:

- MP-3 (MPEG-3) is a file format that is used to store lossy compressed audio. It can result in 75%-90% reduction in the file size, with negligible loss in quality.
- AAC (Advanced Audio Coding) is the successor to mp3, and uses more advanced algorithms to compress audio data even more effectively.

2. 3. 3. Image

Image compression uses algorithms that take into account the visual properties of images to provide results better than regular data compression methods. Images are also resized appropriately to reduce the data stored. [7][8] The following are the most common image forms:

Vector images

The SVG format produces beautiful vector images by rendering lines/shapes from code. It supports transparency, has minimal file size and is ideal for the web, where load speed is crucial. SVG compression works by converting lines into simpler shapes, arranging similar shapes together, and minimising the number of colors.

Lossy Compressed Images

The commonly-used JPEG format is ideal for images with complex shapes and many colors. Firstly, it looks at 8x8 blocks of an image and expels data which do not contribute heavily to the appearance of the image, thereby

[6] Scientific study of how humans perceive audio
[7] Reduced quality due to data damage
[8] Data being out of range

reducing file size with minimal loss of apparent visual quality. Secondly, it uses Chroma Subsampling[9] which throws away unnecessary color data. Thirdly, data in high detail areas is thrown away. However, aggressive compression may lead to blurriness, blocky artifacts and color bleeding which may look unpleasant to the viewer. Moreover, it does not support transparency, which SVG and PNG do.

Lossless Compressed Images

The PNG format adopts lossless algorithms (such as DEFLATE) and is more suitable for images with fewer colors and simpler shapes. It also supports transparency, which jpeg does not. It uses the data from the surrounding pixels and only stores the difference, in order to reduce file size. In other words the colors are put into a table and referenced by number. If the image is limited to 256 (or fewer) colors, the compression is especially effective. However, it struggles with photographic data due to high complexity.

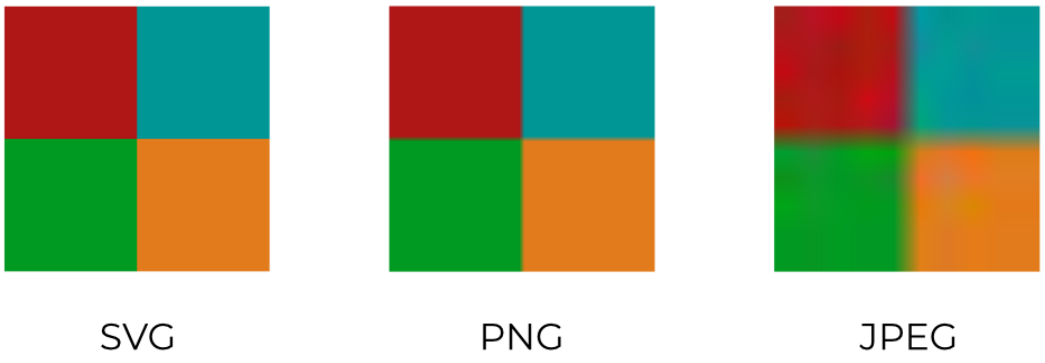

Figure-2.3.3.1 Visual qualities of formats for a simple illustration

Despite all three image formats having roughly the same file size for the above simple illustration (~1.2 kB), SVG is the sharpest, followed by PNG, followed by JPEG.

[9] The color data is of lower quality as compared to the luminance data; humans are less sensitive to color data.

654.01 kB 41.20 kB

Figure-2.3.3.2 Visual qualities of formats for a complex photograph

However, for complex photographic images (both 800px wide and 539px high), PNG is much larger than JPEG despite appearing to be nearly identical; SVG is only for vector shapes and is not appropriate to be used here.

2. 3. 4. Video

Video is the largest of all the four types of data generally seen on the internet. Hence, video data requires greater compression as compared to audio data, image data and text data to be smoothly displayed to the users on the internet. An effective method to compress video data is to resize the video. The effect of resizing videos is highly beneficial as the difference in file size adds up over the multiple frames that are present in the video, thereby greatly reducing the memory required to store video data in an electronic device. Streaming services such as Youtube and Netflix adopt dynamic resizing methods that resize the video based on several factors including -but not limited to- internet speed and screen size. [9] There are 2 forms of compression which reduce the bitrate of videos - Spatial and Temporal.

Spatial Compression

Spatial (intra-frame) compression performs compression on each frame of the video. This compression on every frame is identical to jpeg compression.

Temporal Compression

Temporal (inter-frame) compression performs compression between frames and reduces complexity wherever possible. Frames are divided into blocks, and blocks of consecutive frames (reference frames) are analysed and only the difference between frames is stored using block motion estimation. Firstly, for identical blocks, the block is just repeated instead of storing each one in the video file. Secondly, for non-identical blocks, motion vectors are calculated for moving blocks. These motion vectors are used to move the reference frame in order to match the next frame. These two steps are called motion block estimation and motion compensation. Thirdly, the frame difference between the actual and motion frames is stored in residual frames, as residual frames are highly compressible as they have much lesser data as compared to individual frames of the video. During playback, the video is played by compensating for the motion of the blocks using the motion vectors and adding the residual frames to the reference frames.

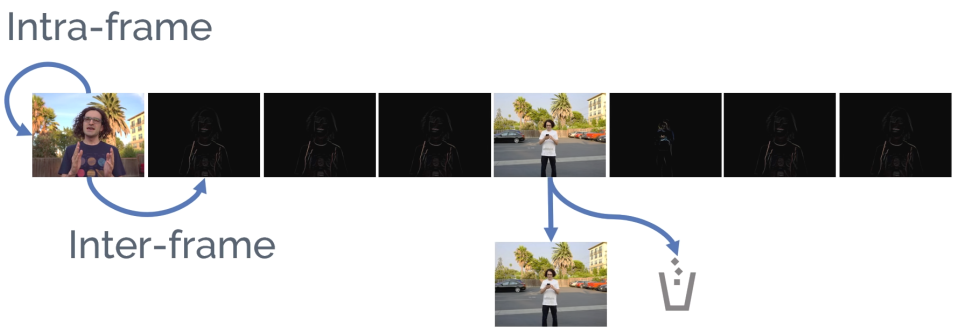

Figure-2.3.4 Inter-frame and intra-frame compression

2. 4. Web Encryption

Web Encryption is a strategy to protect information transferred on the web. It involves taking plain text and rearranging it into a protected format called ciphertext. This helps to protect data stored on devices or transferred on the web. [10]

There are browser extensions such as ‘HTTPS Everywhere’, which claim to “protect you against many forms of surveillance and account hijacking”. It does so by “automatically switching websites from using insecure ‘http’ protocol to secure ‘https’ protocol”, while accessing data on the web. [11]

There are 2 types of encryption systems:

- Symmetric encryption uses a single password to encrypt and decrypt data.
- Asymmetric encryption uses two keys for encryption and decryption: a public key which is shared among users encrypts the data, and a private key which is not shared decrypts the data.

A public key allows encryption of any information on the server, but it cannot decrypt this information. Hence, a private key is used only by the receiver, and only those with the private key can decrypt the information.

The main forms of encryption are E2E, SSL and TLS encryption.

2. 4. 1. End-to-End Encryption

End-to-end is a type of asymmetric encryption used by WhatsApp to ensure that messages sent can only be seen by the sender and the intended recipient and by no one else, not even WhatsApp. [12]

When a user first opens WhatsApp, a public and a private key are generated. The message is encrypted by the public key on the device which sends the message. The private key remains with the user while the public key gets sent to the recipient via the WhatsApp server, which only acts as a medium and receives the encrypted message. The receiver is the only one who can decrypt the message using their private key.

2. 4. 2. SSL and TLS Encryption

SSL(Secure Sockets Layer) is a format for data transmission developed to provide users with a secure connection on the internet. TLS(Transport Layer Security) was later introduced as a successor to SSL and provided enhanced security by encrypting every type of web traffic. [13]

Most web browsers and websites use these protocols for providing the user with a safe mode of entering personal details like credit card numbers. It is easy to identify whether a website supports SSL by checking whether the URL starts with an 'HTTPS' or not. If it does, it specifies that the site is SSL-encrypted, which uses symmetric encryption.

TLS, on the other hand, uses asymmetric encryption only at the start of the transmission session to encrypt the entire transfer of data between the user and the server. This is done after the user and the server agree to use a single session key that both parties will use for encrypting.

The basic method for both protocols is a 3-step handshake method

1. The user requests a secure connection from the server. This is led by algorithms which allow the user to find a suitable connection.
2. The server provides the client with its digital certificate which proves its authenticity. This also includes providing the user with the server's public key.
3. The user and the server will then establish an encrypted session, using the public key.

CHAPTER 3: CONCLUSION

The main ways to improve speed on the web are implementing a CDN system, Lazy Loading, and Data compression, while Data Encryption enhances security.

3. 1. CDN

The implementation of a CDN system improves the speed of data transfer, by reducing the time required for data to load, and by providing instant access to requested data. Furthermore, it facilitates an expansive network that is wide enough to provide fast internet connection to users in regions distant from the main server.

3. 2. Lazy Loading

Lazy Loading is a technique to initially load only critical assets, and delay loading all the others. It is extremely beneficial for e-commerce websites, ensuring user engagement and loyalty.

3. 3. Data Compression

- Text compression mainly involves algorithms such as LZ77, Huffman Coding, and DEFLATE which is a combination of the other two.
- Audio compression reduces file size based on how humans perceive audio. MP3 and AAC provide the best reductions in size.
- Image compression uses visual properties to reduce file size through SVG, PNG, and JPEG formats. The following table shows what factors increase file size of these formats

Lossy	Lossless	Vector
Sharp edges	Frequent, unpredictable, large changes in color	Number of shapes
Small details in undetailed areas	Large number of colors	Complexity of Shapes

Table-3.3 Factors increasing image file size

- Video compression uses a combination of inter-frame and intra-frame compression methods, which reduce file size while maintaining quality.

3. 4. Web Encryption

Web Encryption is a strategy to protect information transferred on the web using a private and/or a public key. End-to-End, SSL and TLS encryption methods help protect credentials from hackers who may exploit security vulnerabilities.

CHAPTER 4: RECOMMENDATIONS

4. 1. For Developers

1. Invest in a CDN system.
2. Lazy loading of non-critical assets must be enabled.
3. Audio must be compressed using mp3 or aac formats.
4. While choosing an image format, SVGs must be used for vectors, PNGs for images requiring transparency, and JPEGs for complex photographs (refer to Figure 11 in appendix for flowchart).
5. Videos must be compressed lossily in mp4 format, at the lowest acceptable bitrate.

4. 2. For Users

1. Try to browse websites with https protocol in the link, whenever possible.
2. Install browser extensions such as HTTPS everywhere to switch websites to use https protocol, if available.
3. Use messaging services such as Whatsapp which provide End-to-End encryption, instead of services such as Instagram which profit off user data by tracking them.

REFERENCES

[1] Maura Monaghan, 2020 *Load Time Statistics* [Online] https://www.websitebuilderexpert.com/building-websites/website-load-time-statistics/?cv=1

[2] IBM Cloud, 2019 *What is a Content Delivery Network* [Online] https://www.youtube.com/watch?v=Bsq5cKkS33I

[3] Houssein Djirdeh, Addy Osmani, 2019 *Browser-level image lazy-loading for the web* [Online] https://web.dev/browser-level-image-lazy-loading/

[4] Encoding audio and video [Online] https://www.bbc.co.uk/bitesize/guides/z7vc7ty/revision/5

[5] Raul Fraile, 2014, How GZIP compression works [Online] https://www.youtube.com/watch?v=wLx5OGxOYUc

[6] Tom Scott, 2017 *How Computers Compress Text* [Online] https://www.youtube.com/watch?v=JsTptu56GM8

[7] Aleks Haecky and Colt McAnlis, 2016, *Understanding Compression - Data Compression for Modern Developers*

[8] Jake Archibald, 2020, *Image compression deep-dive* [Online] https://youtu.be/F1kYBnY6mwg

[9] Leo Isikdogan, 2018, *How Video Compression Works*, [Online] https://youtu.be/QoZ8pccsYo4

[10] Alison Grace Johansen, 2020, *What is encryption and how does it protect your data?* [Online] https://us.norton.com/internetsecurity-privacy-what-is-encryption.html

[11] HTTPS Everywhere, 2020, *Encrypt the Web! Automatically use HTTPS security on many sites.* [Online] https://chrome.google.com/webstore/detail/https-everywhere/gcbommkclmclpchllfjekcdonpmejbdp?hl=en

[12] BusinessToday.in, 2019, *How does WhatsApp end-to-end encryption work* [Online] https://www.businesstoday.in/buzztop/buzztop-feature/how-does-whatsapp-end-to-end-encryption-work/story/307998.html

[13] Digicert, 2020, *What is SSL, TLS and HTTPS?* [Online] https://www.websecurity.digicert.com/security-topics/what-is-ssl-tls-https

[14] Home DJ Studio, 2020, *Ultimate Guide To Audio Bitrate & Audio Formats* https://homedjstudio.com/audio-bitrates-formats/

[15] Youtube Help, 2020, *Recommended upload encoding settings - YouTube Help* [Online] https://support.google.com/youtube/answer/1722171?hl=en

APPENDICES

Audio

The following are the audio bitrates generally used, with 128Kbps(kilobits per second) being the most common:

Bitrates (Kbps)	Quality
< 96	speech or low-quality streaming
128 or 160	acceptable quality
192	medium quality bitrate
256	high-quality
320	highest level supported by the MP3 standard, rarely used online

Table 4 Commonly-used audio bitrates [14]

Images

"Images look good at quality 75, but are half as large as at quality 95. Below 75, there are larger apparent visual changes and reduced savings in file size. The human eye is pretty sensitive to a number of things when viewing images, including edges and gradients. While “quality” is kinda fuzzy, it includes color accuracy, sharpness, contrast, and distortion. Quantization and blocking are some of the most common forms of image compression that result in visual issues." [7]

"Blocking process of jpeg is geared toward photographic images. If you were to compress an image with a much flatter color palette, such as a hand-drawn cartoon, the lossless compressor in PNG would perform better because it can collapse runs of similar colors into single compressed tokens" [7]

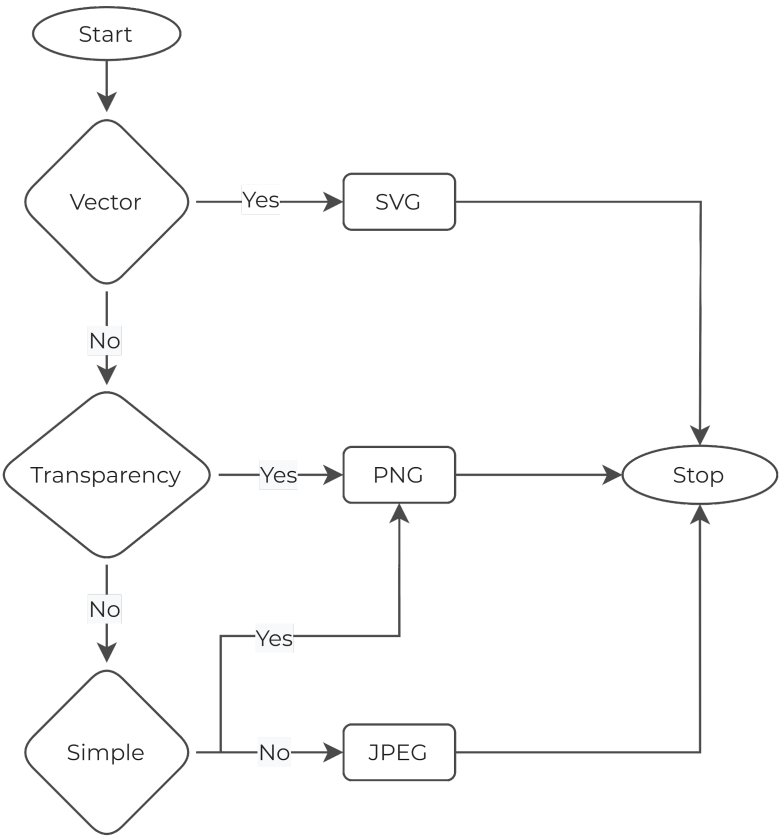

Figure 3 Flowchart for choosing an image format [8]

Video

The following table shows the bitrate used by Youtube videos, for various Dimensions and Bitrates.

Video Dimension	**Video Bitrate (Mbps)** **Frame Rates - 24, 25, 30**
2160p (4k)	35 - 45
1440p (2k)	16
1080p	8
720p	5
480p	2.5
360p	1

Table 5 Recommended video bitrates [15]

We can see that lower bitrates are given to videos with smaller dimensions, i.e. every video file is only given the minimum amount of data it requires to be viewable with minimal artifacts. The implications of this is huge, as unnecessary data is not stored, reducing file size, and hence ensuring smoother playback for users.