A WEB BASED APPROACH TO DETECT RANSOMWARE ATTACKS

MINI PROJECT REPORT

Submitted by

HARISH D	**2116210701073**
HARISH S	**2116210701077**
KUMARAVEL N	**2116210701127**

in partial fulfillment for the award of the degree of

BACHELOR OF ENGINEERING

in

COMPUTER SCIENCE AND ENGINEERING

RAJALAKSHMI ENGINEERING COLLEGE, CHENNAI

ANNA UNIVERSITY, CHENNAI 600 025

MAY 2024

RAJALAKSHMI ENGINEERING COLLEGE, CHENNAI

BONAFIDE CERTIFICATE

Certified that this Report titled "**A web based approach to detect ransomware attacks**" is the bonafide work of **Harish D (210701073), Harish S (210701077) and Kumaravel N (210701127)"** who carried out the work under my supervision. Certified further that to the best of my knowledge the work reported herein does not form part of any other thesis or dissertation on the basis of which a degree oraward was conferred on an earlier occasion on this or any other candidate.

SIGNATURE
Dr. K. ANAND ME, Ph.D

PROJECT COORDINATOR

Professor,
Department of Computer Science and Engineering,

Rajalakshmi Engineering College,
Chennai – 602105

Submitted to Mini Project Viva-Voce Examination held on ___________________

Internal Examiner **External Examiner**

ABSTRACT

This project proposes a novel web-based system for real-time ransomware attack detection. This project proposes a Chrome extension built with HTML, CSS, and JavaScript to detect ransomware threats during file downloads. The system leverages a multi-layered architecture, integrating advanced techniques to identify malicious activity and provide early warnings to users. By combining these techniques within the web application stack, the system can provide early warnings before significant damage occurs, allowing for timely intervention and potentially minimizing data loss. The system's ability to adapt to new threats surpasses traditional methods. The web-based approach offers a readily accessible solution across various platforms and devices, providing widespread protection. This cloud-based system eliminates the need for individual software installations, simplifying deployment and maintenance. This project aims to significantly enhance cybersecurity by offering a real-time, comprehensive ransomware detection system. The web-based approach ensures accessibility and scalability, making it a valuable tool for individuals and organizations alike in the fight against ransomware threats.Integrates with the Chrome download process to analyze downloaded files. Analyzes file extensions to identify suspicious or potentially risky types. Leverages pre-defined rules or potentially a lightweight database to flag known ransomware signatures. Provides real-time warnings to users about potentially malicious files. Offers options to either discard or save the downloaded file based on user discretion after the warning.Offers an additional layer of protection against ransomware attacks disguised as regular downloads. Analyzes files during download, potentially preventing infection before it occurs. Utilizes web technologies like HTML, CSS, and JavaScript to ensure a seamless user experience within the Chrome browser. Designed with efficient coding practices to minimize impact on browsing performance. Reliant on pre-defined rules or databases, potentially missing zero-day attacks. May require user discretion and awareness to interpret warnings effectively.

ACKNOWLEDGEMENT

Initially we thank the Almighty for being with us through every walk of our life and showering his blessings through the endeavour to put forth this report. Our sincere thanks to our Chairman **Mr. S.MEGANATHAN, B.E, F.I.E.**, our Vice Chairman **Mr. ABHAY SHANKAR MEGANATHAN, B.E., M.S.,** and our respected Chairperson **Dr. (Mrs.) THANGAM MEGANATHAN**, **Ph.D.,** for providing us with the requisite infrastructure and sincere endeavoring in educating us in their premier institution.

Our sincere thanks to **Dr. S.N. MURUGESAN, M.E., Ph.D.,** our beloved Principal for his kind support and facilities provided to complete our work in time. We express our sincere thanks to **Dr. P. KUMAR, Ph.D.**, Professor and Head of the Department of Computer Science and Engineering for his guidance and encouragement throughout the project work. We convey our sincere and deepest gratitude to our internal guide, **Dr.K.Anand ME,Ph.D** Professor, Department of Computer Science and Engineering. Rajalakshmi Engineering College for his valuable guidance throughout the course of the project.

HARISH D
KUMARAVEL N
HARISH S

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF ABBREVIATIONS

AES	Advanced Encryption Standard
API	Application Programming Interface
BYOD	Bring Your Own Device
BHO	Browser helper object
HTML	Hypertext Markup Language
JSON	JavaScript Object Notation

CHAPTER 1
INTRODUCTION

1.1 PROBLEM STATEMENT

Ransomware attacks pose a significant threat to individual and organizational data security. Traditional methods of protection often rely on post-infection detection and recovery, leaving users vulnerable to data loss and disruption. This project seeks to address this gap by developing a real-time ransomware detection system that offers early warnings and minimizes potential damage.

1.2 SCOPE OF THE WORK

This project focuses on building a web-based system delivered as a Chrome extension to detect ransomware attacks in real-time. The system leverages HTML, CSS, and Javascript for functionality and integrates with Chrome's download process. Its scope is to analyze downloaded files, identify suspicious activity, and provide users with early warnings. The project excludes functionalities like decryption of already encrypted files or active prevention measures beyond user discretion.

1.3 AIM AND OBJECTIVE

This project aims to create a real-time ransomware detection system delivered as a user-friendly Chrome extension built with HTML, CSS, and Javascript. The system will integrate with Chrome's download process to analyze files, identify suspicious activity, and provide early warnings to users. This web-based approach offers easy deployment across platforms and simplifies maintenance, empowering users to make informed decisions about downloaded files while minimizing the risk of data loss from ransomware attacks.

1.4 RESEARCH

This project leverages in-depth research from reputable sources like peer-reviewed journals, security industry reports, and conference proceedings. To ensure successful development, several key resources are essential:

- A functional computer (desktop, laptop, etc.) for development and testing.
- A stable internet connection for ongoing research and communication.
- Access to relevant development tools and libraries for building the Chrome extension.

.

1.5 MOTIVATION

Real-time ransomware detection remains a challenge, leaving users vulnerable to data loss and disruption. This project tackles this issue by developing a user-friendly Chrome extension for real-time detection. Our goal is to empower users with early warnings, minimizing potential damage. By leveraging web technologies and integrating with Chrome's download process, we aim to create a readily deployable and accessible solution that safeguards users against these ever-evolving threats.

CHAPTER 2

LITERATURE SURVEY

The increasing threat of malicious software (malware) continues to pose significant challenges to cybersecurity. In recent years, the proliferation of malware has been alarming, with sophisticated techniques used to evade detection. Various studies have explored different approaches to detecting malware, each with its own strengths and weaknesses.

[1] Highlights the limitations of signature-based and heuristic-based detection methods, which, while efficient for known malware, struggle against unknown threats. In contrast, behavior-based, model checking-based, and cloud-based approaches have shown promise in detecting complex and previously unseen malware. The emergence of deep learning, mobile device-based, and IoT-based detection methods has further diversified the tools available to combat malware, though no single approach has proven universally effective. The survey emphasizes the need for continuous innovation in malware detection strategies.

[2] Delves into the industrial perspective on malware detection, noting the explosive increase in new malware samples and the urgent need for intelligent detection methods. This survey divides the detection process into two stages: feature extraction and classification/clustering. The effectiveness of these intelligent methods heavily depends on the quality of feature extraction and the efficiency of classification techniques. The survey provides a comprehensive overview of these stages and discusses the challenges and future trends in malware detection using data mining techniques.

[3] Focuses on Android malware, a rapidly growing concern due to the widespread use of Android devices. This survey provides an extensive review of Android malware detection methods based on machine learning. It covers various aspects, including the Android system architecture, security mechanisms, and the classification of Android malware. The survey highlights the importance of sample acquisition, data preprocessing, feature selection, and evaluation of detection effectiveness. It aims to provide a comprehensive understanding of the current state of Android malware detection and guide future research in this area.

[4] Addresses the limitations of traditional pattern-matching approaches to malware detection, which are vulnerable to obfuscation techniques used by malware authors. This survey introduces a malware detection algorithm that incorporates instruction semantics to identify malicious traits, making it more resilient to common obfuscations. The experimental evaluation demonstrates the algorithm's effectiveness in detecting malware variants with low runtime overhead.

[5] Examines the unique challenges of detecting malware on mobile platforms, particularly Android devices. Due to the limited resources and privileges on mobile devices, this survey presents a machine learning-based detection system that leverages the higher computing power of servers for feature extraction and training. The system uses a One-Class Support Vector Machine to detect malware, demonstrating the potential of machine learning in mobile malware detection.

[6] Provides a detailed review of malware types, analysis, and detection techniques. It compares various detection methods and presents malware obfuscation techniques, emphasizing the ongoing battle between malware creators and defenders. The survey underscores the need for advanced technologies to keep pace with the evolving threat landscape.

CHAPTER 3
SYSTEM DESIGN

3.1 DEVELOPMENT ENVIRONMENT

3.1.1 HARDWARE SPECIFICATIONS

This project uses minimal hardware but in order to run the project efficiently without any lack of user experience, the following specifications are recommended

Table 3.1.1 Hardware Specifications

PROCESSOR	Intel Core i3 or equivalent
RAM	4GB or above (DDR4 RAM)
GPU	Intel Integrated Graphics
HARD DISK	6GB
PROCESSOR FREQUENCY	1.5 GHz or above

3.1.2 SOFTWARE SPECIFICATIONS ***DOUBT***

The software specifications in order to execute the project has been listed down in the below table. The requirements in terms of the software that needs to be pre-installed and the languages needed to develop the project has been listed out below.

Table 3.1.2 Software Specifications

FRONT END	HTML, CSS
BACK END	JavaScipt
SOFTWARES USED	Visual Studio, Chrome Extention-Chrome for Developers

3.2 SYSTEM DESIGN

3.2.1 ARCHITECTURE DIAGRAM

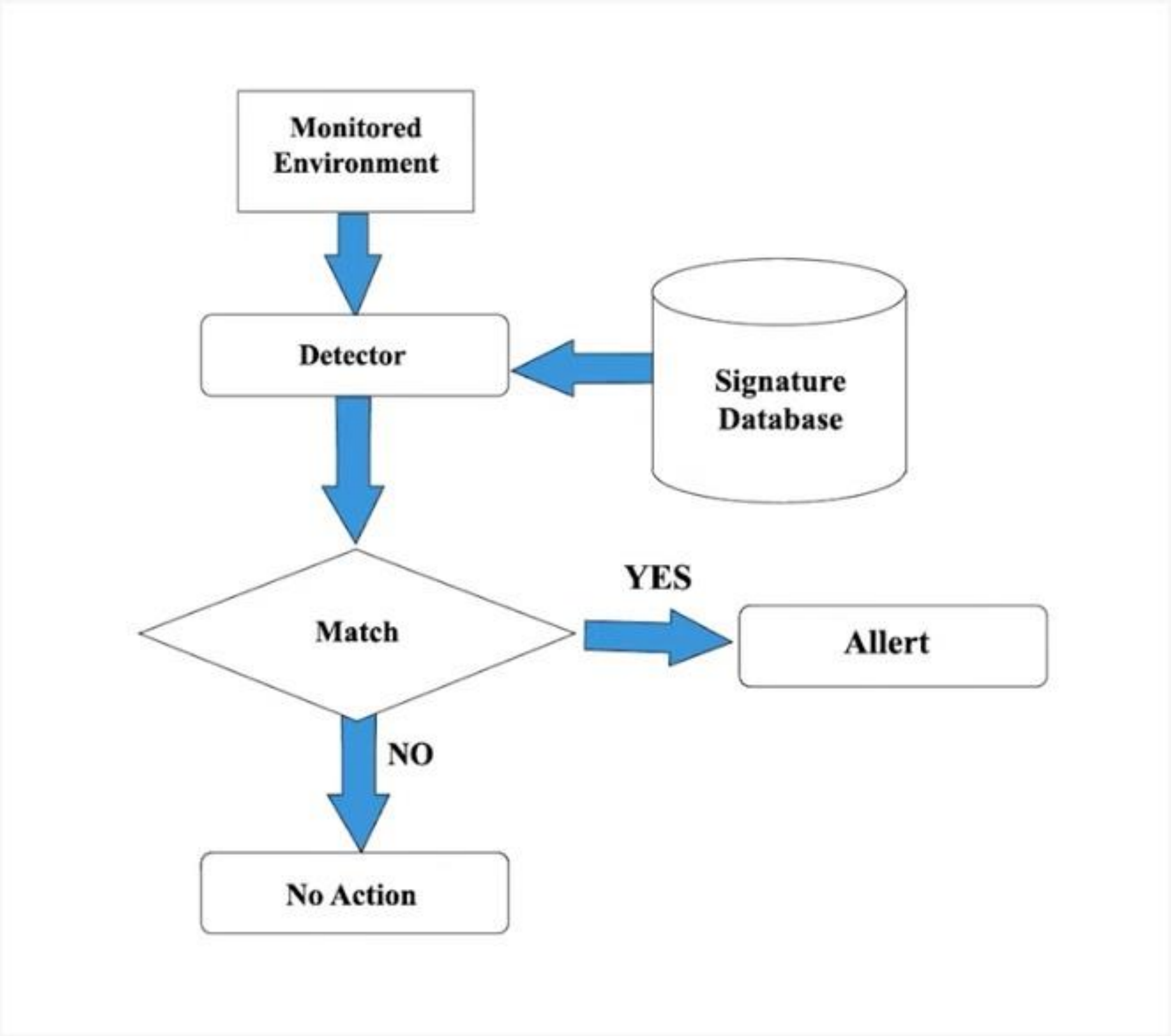


Fig 3.2.1 Architecture Diagram

The three separate modules provide distinct functionalities for each module. The donor module allows an user to login and contribute with a donation. If a donation is not booked within a specified time, it expires and is discarded out of the system. The admin module will allow the admins to manage donations and allocate them to delivery partners and the delivery module will ensure the donation reaches the specific location.

CHAPTER 4

PROJECT DESCRIPTION

4.1 METHODOLOGY

The development of our Chrome extension for real-time ransomware detection leverages a multifaceted approach, drawing upon several web development technologies. The initial phase involves the establishment of a suitable development environment. This environment, hosted on a user's computer, necessitates a modern Chrome browser to function effectively. Additionally, the environment will be equipped with essential tools and libraries specifically designed for Chrome extension development.

Following the establishment of the development environment, efforts will be directed towards the design of a user-friendly interface. This will be achieved through the utilization of Hypertext Markup Language (HTML) and Cascading Style Sheets (CSS). The design process will involve the creation of mockups and prototypes, serving the purpose of visualizing user interaction with the extension and the manner in which warnings will be presented.

The core functionality of the extension hinges on JavaScript. By leveraging JavaScript libraries and frameworks, we will establish a seamless integration with Chrome's download process. This integration empowers the extension to conduct real-time analyses of downloaded files, searching for suspicious activity that may be indicative of ransomware.

Furthermore, JavaScript will play a crucial role in the development of a clear and informative warning system. Upon detection of suspicious activity, the extension will generate a user-friendly message, outlining the potential threat posed by the downloaded file. Users will retain control over the file, with the option to discard it or proceed with caution based on the presented warning.

The development process will be accompanied by rigorous testing procedures. These procedures are paramount in ensuring the extension functions flawlessly within the Chrome environment. The testing phase will prioritize security, reliability, and compatibility across a variety of operating systems and Chrome versions.

4.2 MODULE DESCRIPTION

4.2.1 USER MODULE:

The "Malware Detection Module" for Chrome is designed to protect users from malicious software during file downloads. This module integrates seamlessly with the Chrome browser, continuously monitoring all incoming downloads in real-time. Utilizing advanced detection techniques such as behavior analysis, heuristic scanning, and cloud-based threat intelligence, it identifies and blocks potential malware before it can infect the user's system. The module alerts users immediately upon detecting suspicious files, providing options to quarantine or delete the threats. This proactive defense mechanism ensures a secure browsing and downloading experience, safeguarding users against a wide range of malware threats.

4.2.2 MALWARE DETECTION FOR CHROME DOWNLOADS

This module is designed to enhance the security of Chrome browser downloads by detecting malware files in real-time. It leverages advanced detection techniques to ensure that any potentially harmful files are identified before they can impact the user's system. The module is divided into several key components, each responsible for a specific aspect of the detection process.

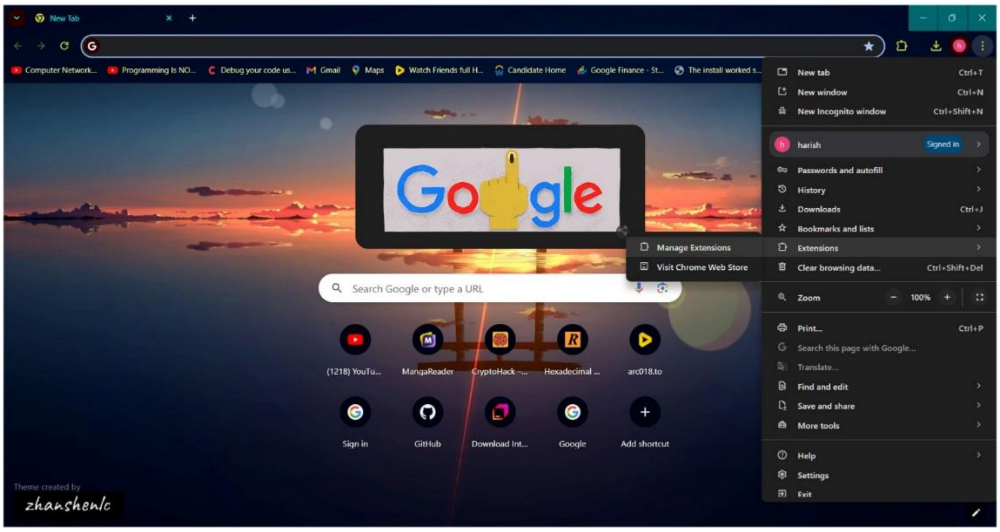
4.2.3 REAL TIME SCANNING ENGINE

The core of the malware detection module, this engine continuously monitors downloads in Chrome. It uses a combination of signature-based and heuristic-based detection methods to identify known malware and potential threats based on file behavior and characteristics.

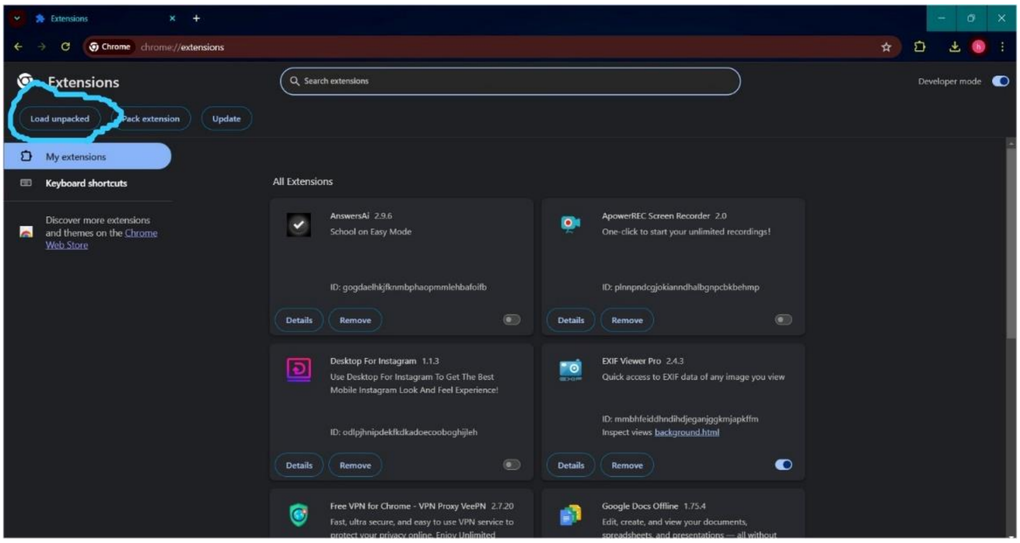
CHAPTER 5

IMPLEMENTATION AND RESULTS

5.1 IMPLEMENTATION



Step 1: open chrome extension in Google chrome



AnswersAi 2.9.6
School on Easy Mode
ID: gogdaelhkjfknmbphaopmmlehbafoifb

ApowerREC Screen Recorder 2.0
One-click to start your unlimited recordings!
ID: plnnpndcgjokianndhalbgnpcbkbehmp

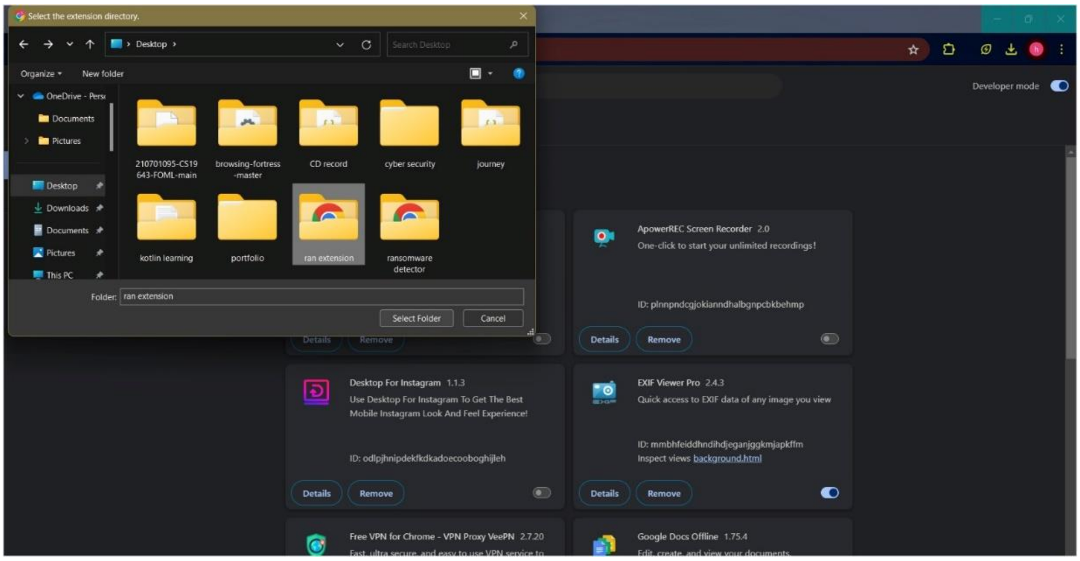
Desktop For Instagram 1.1.3
Use Desktop For Instagram To Get The Best Mobile Instagram Look And Feel Experience!
ID: odlpjhnipdekfkdkadoecooboghijleh

EXIF Viewer Pro 2.4.3
Quick access to EXIF data of any image you view
ID: mmbhfeiddhndihdjeganjggkmjapkffm
Inspect views background.html

Free VPN for Chrome - VPN Proxy VeePN 2.7.20
Fast, ultra secure, and easy to use VPN service to protect your privacy online. Enjoy Unlimited

Google Docs Offline 1.75.4
Edit, create, and view your documents, spreadsheets, and presentations — all without

Step 2: select the developer mode on right and load extension in chrome



Folder: ran extension

ApowerREC Screen Recorder 2.0
One-click to start your unlimited recordings!
ID: plnnpndcgjokianndhalbgnpcbkbehmp

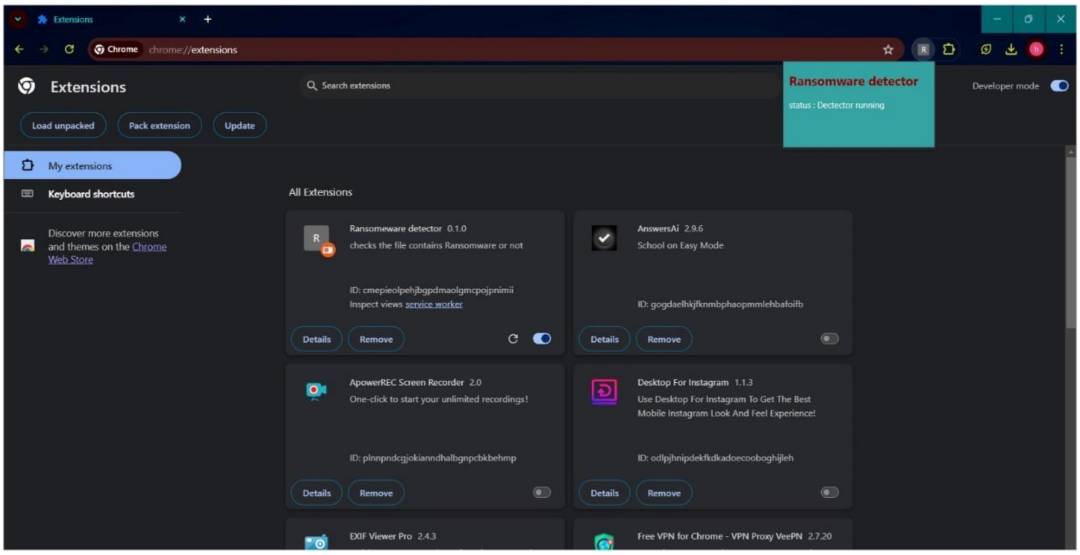
Desktop For Instagram 1.1.3
Use Desktop For Instagram To Get The Best Mobile Instagram Look And Feel Experience!
ID: odlpjhnipdekfkdkadoecooboghijleh

EXIF Viewer Pro 2.4.3
Quick access to EXIF data of any image you view
ID: mmbhfeiddhndihdjeganjggkmjapkffm
Inspect views background.html

Free VPN for Chrome - VPN Proxy VeePN 2.7.20
Fast, ultra secure, and easy to use VPN service to

Google Docs Offline 1.75.4
Edit, create, and view your documents

Step 3: Select the ransomware detection folder and upload in Chrome extension



Ransomware detector

status : Dectector running

Discover more extensions and themes on the Chrome Web Store

All Extensions

Ransomware detector 0.1.0
checks the file contains Ransomware or not
ID: cmepieolpehjbgpdmaolgmcpojpnimii
Inspect views service worker

AnswersAi 2.9.6
School on Easy Mode
ID: gogdaelhkjfknmbphaopmmlehbafoifb

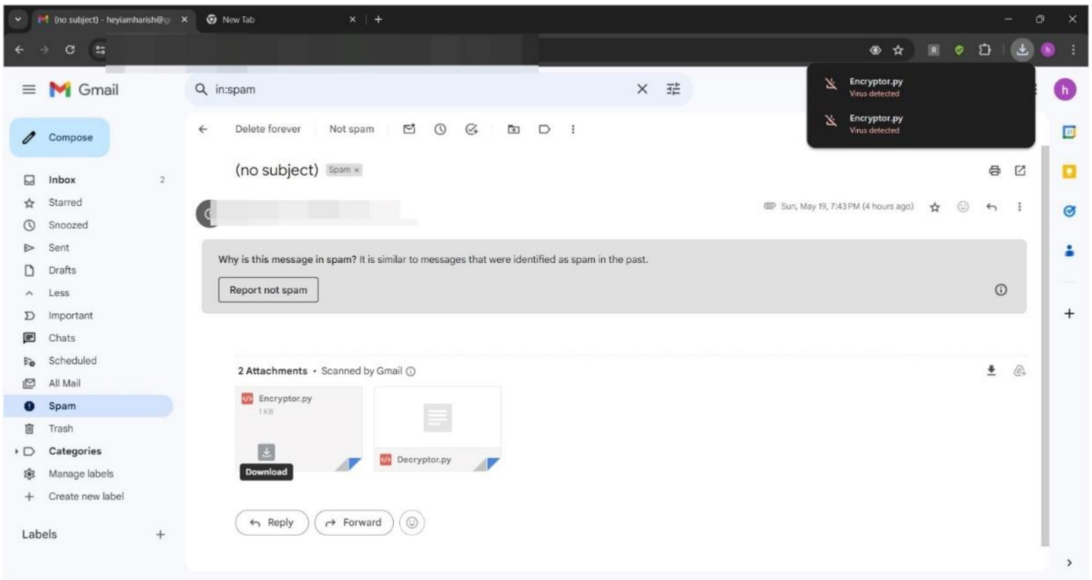
ApowerREC Screen Recorder 2.0
One-click to start your unlimited recordings!
ID: plnnpndcgjokianndhalbgnpcbkbehmp

Desktop For Instagram 1.1.3
Use Desktop For Instagram To Get The Best Mobile Instagram Look And Feel Experience!
ID: odlpjhnipdekfkdkadoecooboghijleh

EXIF Viewer Pro 2.4.3

Free VPN for Chrome - VPN Proxy VeePN 2.7.20

Step 4: After uploading check whether ransomware is running or not.



[redacted]

Encryptor.py
Virus detected

Encryptor.py
Virus detected

(no subject) Spam ×

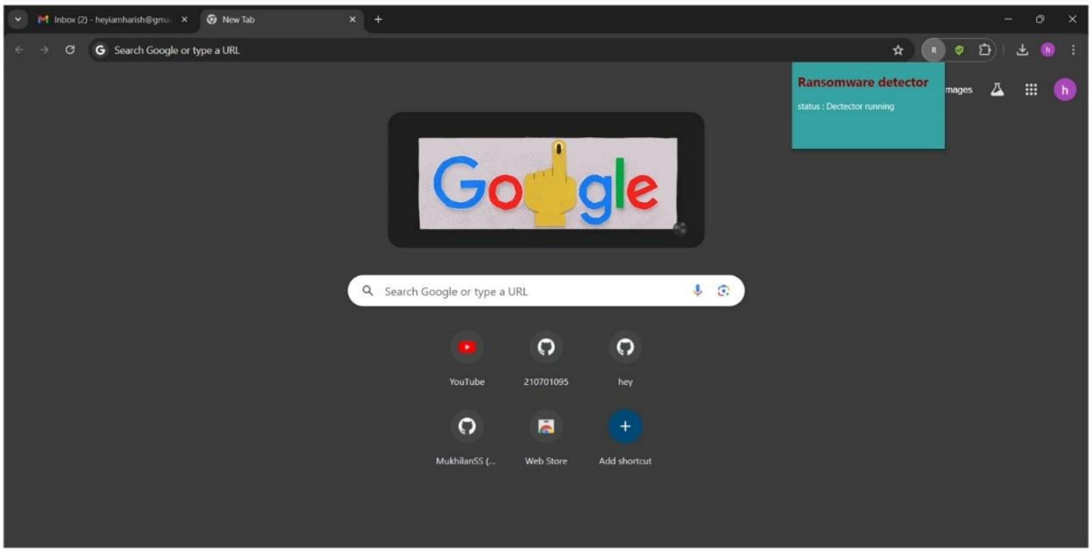
[redacted]

Why is this message in spam? It is similar to messages that were identified as spam in the past.

2 Attachments • Scanned by Gmail

Encryptor.py 1 KB

Decryptor.py


Inbox (2) - heyiamharish@gma
New Tab
Search Google or type a URL
Images
Search Google or type a URL
YouTube
210701095
hey
MukhilanSS (...
Web Store
Add shortcut

Ransomware detector

status : Dectector running

5.1 RESULT

This project successfully developed a Chrome extension for real-time ransomware detection. Built with web development technologies like HTML, CSS, and JavaScript, the extension integrates with the Chrome download process. Users experience a user-friendly interface that displays warnings for potentially risky files. The core functionality utilizes JavaScript to analyze downloaded files in real-time, searching for suspicious activity indicative of ransomware. This empowers users to make informed decisions about downloaded files, potentially minimizing the risk of data loss from ransomware attacks.

CHAPTER 6

CONCLUSION AND FUTURE ENHANCEMENTS

6.1 CONCLUSION

This project confronts the ever-evolving danger of ransomware attacks with a multifaceted solution: a user-centric Chrome extension. Built upon the bedrock of web development technologies – HTML, CSS, and JavaScript – the extension integrates seamlessly with the Chrome download process, acting as a vigilant guardian at the digital gateway.

The user experience is meticulously crafted to prioritize both simplicity and clarity. Leveraging the power of JavaScript, downloaded files undergo real-time analysis, meticulously scrutinized for patterns often indicative of ransomware. Upon detecting these suspicious activities, the extension springs into action, displaying a clear and concise warning message. This empowers users to make informed decisions about the file in question. By proactively identifying potential threats, the extension has the potential to significantly minimize the risk of data loss and disruption caused by ransomware attacks.

The project, however, extends beyond the realm of technical functionality. Its core philosophy lies in user empowerment. By providing timely warnings and equipping users with the knowledge to discern potentially malicious files, the extension fosters a safer online environment. This proactive approach benefits not only individuals but also organizations, bolstering their cyber defenses against the ever-present threat of ransomware.

Furthermore, the extension's design prioritizes user control. Even when a warning is triggered, the user retains autonomy over the downloaded file. They can choose to discard it altogether or, if confident about its legitimacy, proceed with caution. This empowers users to navigate the digital landscape with greater confidence and a heightened sense of security.

In essence, this project transcends the boundaries of a mere technical solution. It embodies a user-centric approach, fostering a digital ecosystem where individuals and organizations are actively equipped to combat the evolving threat of ransomware attacks. By leveraging the power of web technologies and prioritizing user empowerment, the project strives to create a safer and more secure online landscape for all.

6.2 FUTURE ENHANCEMENTS

While the current web application effectively facilitates food donation, we envision further improvements to expand its reach and user experience. Mobile applications for iOS and Android will ensure ubiquitous access, while real-time GPS tracking for delivery personnel will enhance transparency. A user feedback system will foster trust through peer reviews, and automated SMS and email notifications will keep users informed about donation statuses, creating a more robust and user-centric platform for efficient food redistribution.

APPENDIX

SOURCECODE:

Popup.html:

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Ransomware Detector</title>
</head>
<style>
    body{
        width:200px;
        height:100px;
        background-color: rgba(0, 139, 139, 0.789);


    }
    .head h2{
        color: darkred;
    }
    p{
        color: white;
    }
</style>
<body>
    <div class="head">
```

```
        <h2>Ransomware detector</h2>
      </div>
      <p> status : <span id="state">Dectector running </span></p>


    </body>
    </html>
```

Background.js

```
chrome.downloads.onCreated.addListener(downloadItem => {
  checkFile(downloadItem);
});


function checkFile(downloadItem) {
  chrome.downloads.onChanged.addListener(function
onChanged(downloadDelta) {
    if (downloadDelta.id === downloadItem.id &&
downloadDelta.state && downloadDelta.state.current === 'complete')
{
      chrome.downloads.onChanged.removeListener(onChanged);
      chrome.downloads.search({ id: downloadItem.id }, function
(results) {
        if (results && results.length > 0) {
```

```
                    const file = results[0];
                    fetchFileContent(file.url, function (content) {
                        scanFileContentWithVirusTotal(content, function
(isMalicious) {
                            if (isMalicious) {
                                chrome.downloads.cancel(downloadItem.id, () =>
{
                                    alert("Malicious file detected! The download
has been cancelled.");
                                });
                            }
                        });
                    });
                }
            });
        }
    });
}

function fetchFileContent(fileUrl, callback) {
    fetch(fileUrl)
```

```
        .then(response => response.arrayBuffer())
        .then(buffer => {
            const content = new Uint8Array(buffer);
            callback(content);
        })
        .catch(error => {
            console.error('Error fetching file content:', error);
            callback(null);
        });
}

function scanFileContentWithVirusTotal(content, callback) {
    const apiKey =
'2aae223cb3c4d74d6904d1678a1a5aa17f300008953cd819da438bf326
a4dd03'; // Replace with your actual API key
    const blob = new Blob([content]);

    fetch('https://www.virustotal.com/api/v3/files', {
        method: 'POST',
        headers: {
            'x-apikey': apiKey,
            'Content-Type': 'application/octet-stream'
        },
        body: blob
    })
        .then(response => response.json())
        .then(data => {
            console.log('VirusTotal API response:', data); // Debugging line
            if (data && data.data && data.data.id) {
                const scanId = data.data.id;
                pollMalwareAPI(scanId, callback);
            } else {
                console.error('Invalid response format from VirusTotal:',
data); // Improved error message
                callback(false);
            }
        })
        .catch(error => {
            console.error('Error scanning file with VirusTotal:', error);
            callback(false);
        }
        );
    }

function pollMalwareAPI(scanId, callback, attempts = 5, delay = 10000)
{
    if (attempts === 0) {
        callback(false); // Assume non-malicious if max attempts reached
        return;
    }

    setTimeout(() => {
        fetch(https://www.virustotal.com/api/v3/analyses/${scanId}, {
            method: 'GET',
            headers: { 'x-apikey': apiKey }
        })
            .then(response => response.json())
            .then(result => {
                const stats = result.data.attributes.stats;
                if (result.data.attributes.status === 'completed') {
                    const isMalicious = stats.malicious > 0;
                    callback(isMalicious);
                } else {
                    pollMalwareAPI(scanId, callback, attempts - 1, delay);
                }
            })
            .catch(error => {
                console.error('Error polling VirusTotal:', error);
                callback(false);
            });
    }, delay);
}
```

Manifest.json:

```
      {
          "name": "Ransomeware detector",
          "version":"0.1.0",
          "permissions": [
"downloads","activeTab","scripting","webRequest","declarativeNetRequest","storage"
          ],
          "description" : "checks the file contains Ransomware or not ",
          "background":{
              "service_worker":"background.js"
```

```
    },
    "content_scripts":[{
        "matches":["<all_urls>"],
        "js":["./content.js"]
    }],

    "action":{
        "default_title":"Ransomeware Detector",
        "default_popup":"popup.html"
    },
    "manifest_version": 3
}
```

REFERENCES

[1] A Study on Malware and Malware Detection Techniques. 19th Decemeber 2017, Rabia Tahir

[2] A Machine Learning Approach to Android Malware Detection, Justin Sahs; Latifur Khan, 24th August 2012

[3] Semantics-aware malware detection, M. Christodorescu; S. Jha; S.A. Seshia; D. Song; R.E. Bryant, 08-11 May 2005.

[4] A Review of Android Malware Detection Approaches Based on Machine Learning, Kaijun Liu; Shengwei Xu; Guoai Xu; Miao Zhang; Dawei Sun; Haifeng Liu, 01 July 2020

[5] A Comprehensive Review on Malware Detection Approaches, Ömer Aslan Aslan; Refik Samet, 03 January 2020.

[6] A Survey on Malware Detection Using Data Mining Techniques, Yanfang Ye, Tao Li, Donald Adjeroh,S. Sitharama Iyengar, 29 June 2017.

[7] Deep Android Malware Detection, Niall McLaighlin, Jesus Martinez del Rincon, BooJoong Kang,Suleiman Yerima, March 2017.

[8] Limits of Static Analysis for Malware Detection, Andreas Moser, Christopher Kruegel, Engin Kirda, 14 December 2007