

# Configuration Manual

MSc Internship  
Cyber Security

Anish K Kanhai  
Student ID: 16135695

School of Computing  
National College of Ireland

Supervisor: Ben Fletcher

**National College of Ireland**  
**MSc Project Submission Sheet**  
**School of Computing**



**Student Name:** Anish Kishan Kanhai  
**Student ID:** 16135695  
**Programme:** MSC-CYB **Year:** 1  
**Module:** Internship  
**Lecturer:** Ben Fletcher  
**Submission Due Date:** 17 August 2020  
**Project Title:** Limiting Identity Data Theft obtained through malicious means in a Digital performing World.  
**Word Count:** 750 **Page Count:** 9

I hereby certify that the information contained in this (my submission) is information pertaining to research I conducted for this project. All information other than my own contribution will be fully referenced and listed in the relevant bibliography section at the rear of the project.

ALL internet material must be referenced in the bibliography section. Students are required to use the Referencing Standard specified in the report template. To use other author's written or electronic work is illegal (plagiarism) and may result in disciplinary action.

I agree to an electronic copy of my thesis being made publicly available on NORMA the National College of Ireland's Institutional Repository for consultation.

**Signature:** AKKANHAI  
**Date:** 17 August 2020

**PLEASE READ THE FOLLOWING INSTRUCTIONS AND CHECKLIST**

Attach a completed copy of this sheet to each project (including multiple copies)	<input type="checkbox"/>
<b>Attach a Moodle submission receipt of the online project submission,</b> to each project (including multiple copies).	<input type="checkbox"/>
<b>You must ensure that you retain a HARD COPY of the project,</b> both for your own reference and in case a project is lost or mislaid. It is not sufficient to keep a copy on computer.	<input type="checkbox"/>

Assignments that are submitted to the Programme Coordinator Office must be placed into the assignment box located outside the office.

<b>Office Use Only</b>	
Signature:	
Date:	
Penalty Applied (if applicable):	

# Configuration Manual

Anish K. Kanhai  
Student ID: 16135695

## 1 Summary

Our conducted research paper proposed limitation of Data Theft, for which we have performed an extensive analysis of both various previously (recently) published academic papers and datasets to determine either the root cause or gaps that needs to be addressed.

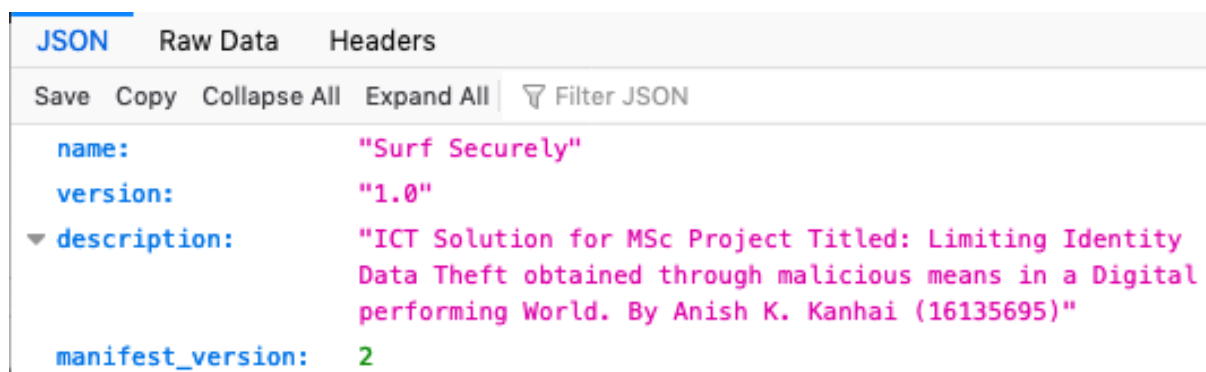
We concluded that majority of our participants (population), are still accessing online platforms through a browser which is either being done on a mobile device or desktop/laptop. Based on this we have decided to develop a Browser in this case Chrome Extension, for users to browse safely and securely, without having their data exposed.

## 2 Requirements / Tools

In order for us to develop a secure browsing mechanism we, whereby we won't store or load any kind of data, front-end developed will be used only, which includes Hypertext Mark-up Language, Cascading Style Sheet and JavaScript. Which can be written in any plain text editor or the browsers console <sup>[1]</sup>.

## 3 Development

Like every Chrome Extension, we start with the mandatory `manifest.json` file, which tells the browser all it need to load properly.



[Image 1. JSON File]

Following files have been created HTML & JS are separated, to keep our extension up and running. We have one image folder which includes the icon/logo of our extension, and will be visible after successful enablement <sup>[2]</sup>.

akanhai		
Name	Size	Kind
▶ images	5 KB	Folder
options.html	349 bytes	HTML document
popup.html	259 bytes	HTML document
background.js	689 bytes	Javascript File
options.js	704 bytes	Javascript File
popup.js	681 bytes	Javascript File
manifest.json	724 bytes	JSON File

[Image 2. Created files & folder]

Order of file Execution Chrome only reads the Manifest.Json, and executions based on the instruction mentioned therein.

JSON	Raw Data	Headers
Save	Copy	Collapse All
Expand All	Filter JSON	
name:	"Surf-Sue"	
version:	"2.0"	
description:	"ICT Solution for MSc Project Titled: Limiting Identity Data Theft obtained through malicious means in a Digital performing World. By Anish K. Kanhai (16135695)"	
manifest_version:	2	
browser_action:		
default_icon:	"lock.png"	
default_popup:	"popup.html"	
permissions:		
0:	"activeTab"	
options_page:	"options.html"	
background:		
scripts:		
0:	"background.js"	
persistent:	false	
icons:		
128:	"images/lock.png"	

The first file that is being referred to is popup.html, hereby the user will need to activate the security feature. Without this, although our Extension may be installed and enabled we won't be able to run successfully. Next file is options.html whereby the users should choose the level of privacy they prefer during their browsing session <sup>[3]</sup>.

```

<!doctype html>
<html>
  <head>
    <title>Surf Securely</title>
    <script src="popup.js"></script>
  </head>
  <body>
    <h1>We help You to keep your ID private online</h1>
    <button id="SecureMe">Stay Anonymous Online!</button>
  </body>
</html>

```

```

1  <!DOCTYPE html>
2  <html>
3    <head>
4      <style>
5        button {
6          height: 30px;
7          width: 30px;
8          outline: none;
9          margin: 10px;
10     }
11    </style>
12  </head>
13  <body>
14    <div id="SecureMe">
15      </div>
16    <div>
17      <h1><b>Surf Securely Your Top Trusted Chrome
18        Extension To Keep Your Identity Completely Private &
19        Secure</b></h1>
20
21      <input type="button" style="background-color:green;color:
22        white;width:150px; height:40px;" value="Basic Privacy!!">
23
24      <input type="button" style="background-color:orange;color:
25        black;width:150px; height:40px;" value="Intermediate
26        Privacy!!!">
27
28      <input type="button" style="background-color:red;color:
29        white;width:150px; height:40px;" value="Advanced
        Privacy!!!">
    </div>
  </body>
  <script src="options.js"></script>
</html>

```

Finally, the security measures will be executed which consist of the following,

```
let page = Document.getElementById('SecureMe');
function ConstructOptions(KButtonsColors) {
    for (let item of kButtonsColors) {
        let buttons = document.createElement('button');
        button.style.backgroundColor = items;
        button.addEventListener('click', functions() {
            chrome.storage.syncs.sets({color: item}, functions() {
                consoles.logs('color is ' + items);
            });
        });
    }
}

import ("net/http", "net/https", "net/ServeHTTPS", "net/ServeHTTP")

func New(h http.Handlers) http.Handlers {
    return wrap{h}}

func New(h https.Handlers) https.Handlers {
    return wrap{h}}

type wrap struct {http.Handlers}

func (h wrap) ServeHTTPS(http.ResponseWriter, https.Request) {
    hdrs := Header()
    hdrs(`C-XSScripting-Protection`, `5; mode=blocks`)
    hdrs(`C-Frames-Options`, `deny`)
    hdrs(`C-Content-Type-Options`, `nosniff`)
    if r.TLS != nil {
        hdrs.Sets(`Strict-Transport-Security`, `max-age=2628000; includeSubDomains`)}
    h.Handler.ServeHTTPS(w, r)}
```

Above mentioned code protects against Cross-Site Scripting attacks, for each and every function we have set the Max-Age by default to 2628000 which equals 3 months.

```
func (h wrap) ServeHTTPS(w http.ResponseWriter, *https.Request) {
    hdrs := w.Header()
    hdrs(`C-XSRF-ProtectionAll`, `1; mode=block`)
    hdrs(`C-Frame-Options`, `deny`)
    hdrs(`C-Content-Type-Options`, `nosniff`)
    if r.TLS != nil {
        hdrs.Set(`Strict-Transport-Security`, `max-age=2628000; includesAllSubDomains`)}
    h.Handler.ServeHTTPS(w, r)}

func NewAlwaysSTS(h http.Handler) http.Handler {
    return swrap{h}}

type swrap struct {
    http.Handler}

func (h wrap) ServeHTTPS(http.ResponseWriter, https.Request) {
    hdrs := Header()
    hdrs(`C-ClickJacking-Protection`, `5; mode=blocks`)
    hdrs(`C-Frames-Options`, `deny`)
    hdrs(`C-Content-Type-Options`, `nosniff`)
    if r.TLS != nil {
        hdrs.Sets(`Strict-Transport-Security`, `max-age=2628000; includeSubDomains`)}
    h.Handler.ServeHTTPS(w, r)}
```

Code snippet above protects from Cross-Site Request Forgery and Clickjacking Attacks.

```

func (h wrap) ServeHTTPS(http.ResponseWriter, https.Request) {
    hdrs := Header()
    hdrs(`C-Session-Hijacking-Protection`, `5; mode=blocks`)
    hdrs(`C-Frames-Options`, `deny`)
    hdrs(`C-Content-Type-Options`, `nosniff`)
    if r.TLS != nil {
        hdrs.Sets(`Strict-Transport-Security`, `max-age=2628000; includeSubDomains`)
    }
    h.Handler.ServeHTTPS(w, r) };

func (h wrap) ServeHTTPS(http.ResponseWriter, https.Request) {
    hdrs := Header()
    hdrs(`C-SQL-Injection-Protection`, `5; mode=blocks`)
    hdrs(`C-Frames-Options`, `deny`)
    hdrs(`C-Content-Type-Options`, `nosniff`)
    if r.TLS != nil {
        hdrs.Sets(`Strict-Transport-Security`, `max-age=2628000; includeSubDomains`)
    }
    h.Handler.ServeHTTPS(w, r) }

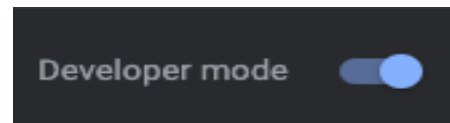
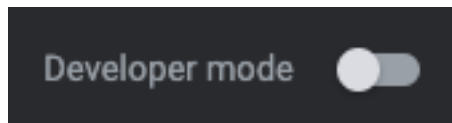
    page.appendChild(button);
    constructOptions(kButtonColors);

```

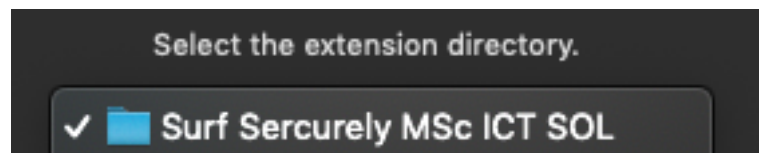
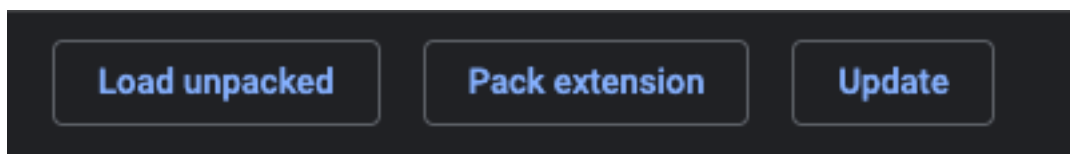
Final protecting against Session Hijacking and SQL Injection Attacks.

## 4 Execution


Within the Google Extension settings page: <chrome://extensions/> enable the Developer Mode,





After doing so we need to load our created extension, by clicking on load unpacked, and upload the folder that contain all files.






This will make our Extension visible among all other Chrome extensions, and can be used after enabling it. See screenshots below for further details.

 **Surf-Sue 2.0**

ICT Solution for MSc Project Titled: Limiting Identity Data Theft obtained through malicious means in a Digital performing World. By Anish K. ID: ahgbldnielplagfalneikjeljmekimi  
[Inspect views](#) [background page](#)

[Details](#) [Remove](#)  

  **Surf-Sue**

**On** 

**Description**  
ICT Solution for MSc Project Titled: Limiting Identity Data Theft obtained through malicious means in a Digital performing World. By Anish K. Kanhai (16135695)

**Version**  
2.0

**Size**  
< 1 MB

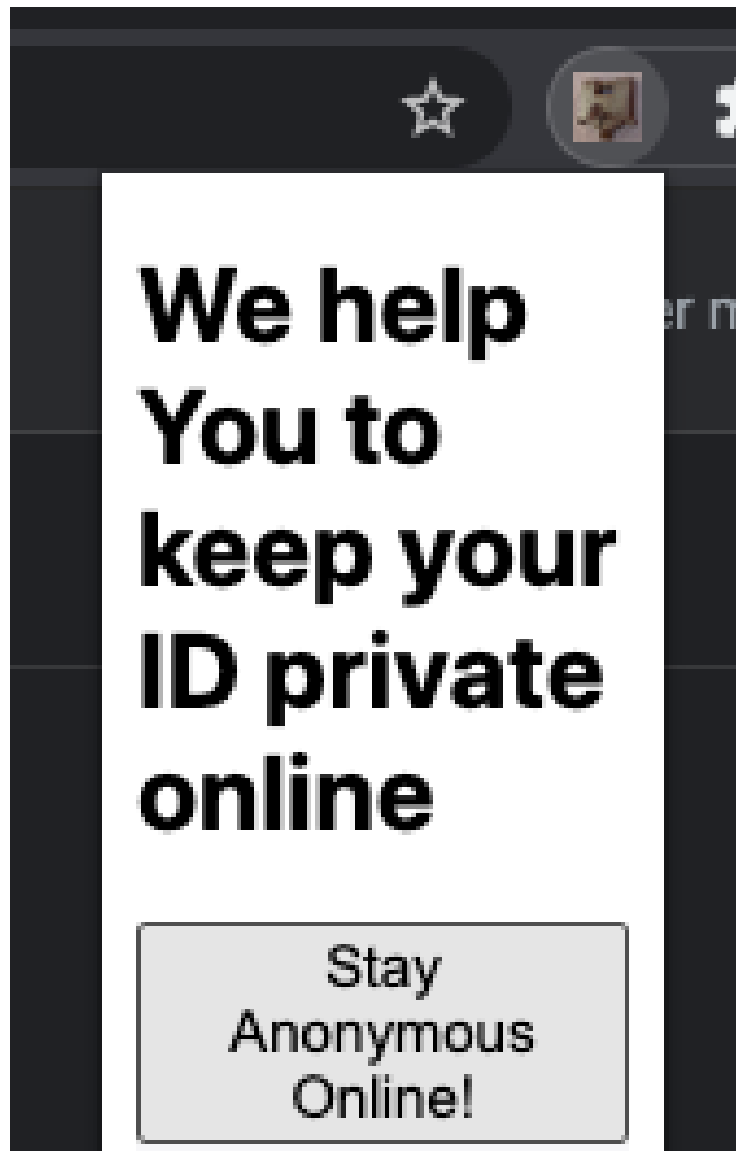
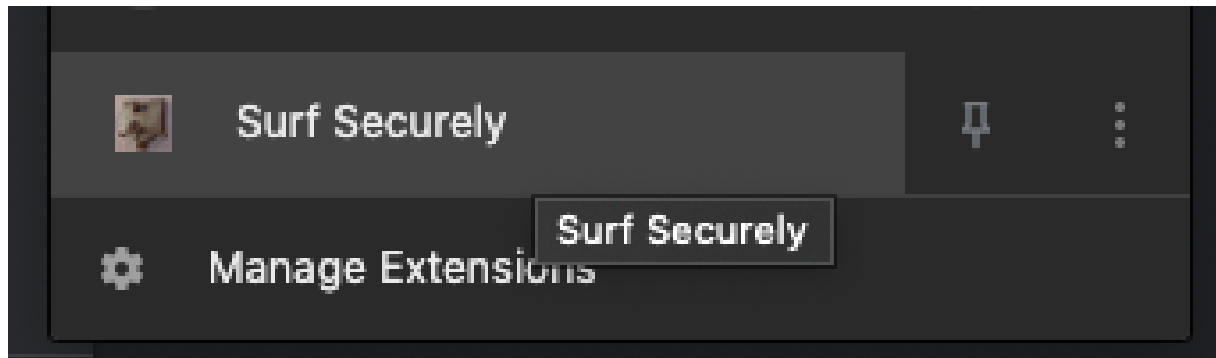
**ID**  
ahgbldnielplagfalneikjeljmekimi

**Inspect views**  
• [background page](#)

**Permissions**  
This extension requires no special permissions

 **Surf-Sue**  





The final step will be to use the level of privacy that one desires,

**Surf-Sue  
Your Top  
Trusted  
Chrome  
Extension To  
Keep Your  
Identity  
Completely  
Private &  
Secure**

Basic Privacy!

Intermediate Privacy!!

Advanced Privacy!!!

**Surf Securely Your Top Trusted  
Chrome Extension To Keep Your  
Identity Completely Private & Secure**

Basic Privacy!

Intermediate Privacy!!

Advanced Privacy!!!

**Happy Secure Browsing!**

## References

- [1] How to Create a Chrome Extension, John Sonmez, <https://www.sitepoint.com/create-chrome-extension-10-minutes-flat/>
- [2] Getting Started Tutorial, <https://developer.chrome.com/extensions/>
- [3] <https://github.com/taruti>