An Efficient and Usable Client-Side Cross Platform Compatible Phishing Prevention Application

FIRST REVIEW

Guide
Dr. Angelin Gladston
Associate Professor
Department of CSE

Submitted by
N. Dhanush   2016103021
G. Santhosh   2016103057
S. Ben Stewart   2016103513

OUTLINE

1. INTRODUCTION
2. OVERALL OBJECTIVE
3. LITERATURE SURVEY
4. PROPOSED SYSTEM
5. HIGH LEVEL BLOCK DIAGRAM
6. MODULE LIST
7. IMPLEMENTATION
8. EVALUATION METRICS
9. REFERENCES

INTRODUCTION

- Phishing
- Lists of such sites
- Time constraints
- Computational resources
- Vulnerabilities
- Cross platform

OVERALL OBJECTIVE

- Create a phishing list
- Cross Platform application
- Web browser add-on
- Provide temporal resilience
- Remove false positives from list

LITERATURE SURVEY

- Previous work by Samuel Marchal, Giovanni Armano, Tommi Grondahl, Kalle Saari, Nidhi Singh, and N. Asokan
- IEEE Trans. Comput., vol. 66, no. 10, pp. 1717-1733, Oct. 2017
- Implemented a client-side phishing prevention application.
- Had background tasks communicate with a browser add-on.
- Not platform independent.

AUTOMATIC PHISHING CLASSIFICATION

- Colin Whittaker, Brian Ryner and Marria Nazif for Google
- Proc. Netw. Distrib. Syst. Security Symp., 2010
- Features used
    1. The URL of the page
    2. The HTML page contents
    3. The host server details
- Needs blacklist updating.

CANTINA

- Guang Xiang, Jason Hong, Carolyn P. Rose and Lorrie Cranor
- ACM Trans. Inf. Syst. Secur., 2011
- Page similarity
- SHA 1 algorithm
- Easy to break
- Performance gains

AUTO UPDATED WHITELIST

- Ankit Kumar Jain and B. B. Gupta
- EURASIP J. Inf. Secur., vol. 2016, no. 1, Dec. 2016
- Whitelist
  a. the domain name
  b. the IP address
- Reverts to old system if not in whitelist

# FUZZY ROUGH SET FEATURE SELECTION TO ENHANCE PHISHING ATTACK DETECTION

- Mahdieh Zabihimayvan and Derek Doran
- IEEE International Conference on Fuzzy Systems, June 2019
- Fuzzy Rough Set (FRS) theory
- Feature selection algorithm
- Random Forest classification
- No third party features

| Paper	Journal/Conf. , Year	Contributions	Limitations
Large-Scale Automatic Classification of Phishing Pages	Proc. Netw. Distrib. Syst. Security Symp., 2010	Machine learning model can be used with reliable accuracy.	Needs blacklist for updating.
CANTINA: A feature-rich machine learning framework for detecting phishing Web sites	ACM Trans. Inf. Syst. Secur., 2011	SHA1 based similarity check for similar looking sites.	SHA1 could be manipulated.
A novel approach to protect against phishing attacks at client side using auto-updated white-list	EURASIP J. Inf. Secur., vol. 2016, no. 1, Dec. 2016	Auto-updated whitelist for faster detection of sites on average.	Not temporally resilient.
Fuzzy Rough Set Feature Selection to Enhance Phishing Attack Detection	IEEE International Conference on Fuzzy Systems, June 2019	Feature selection.	Not a user oriented application.

PROPOSED SYSTEM

- Platform independent
- Browser add-on
- Reduce false warnings
- Context independent detection
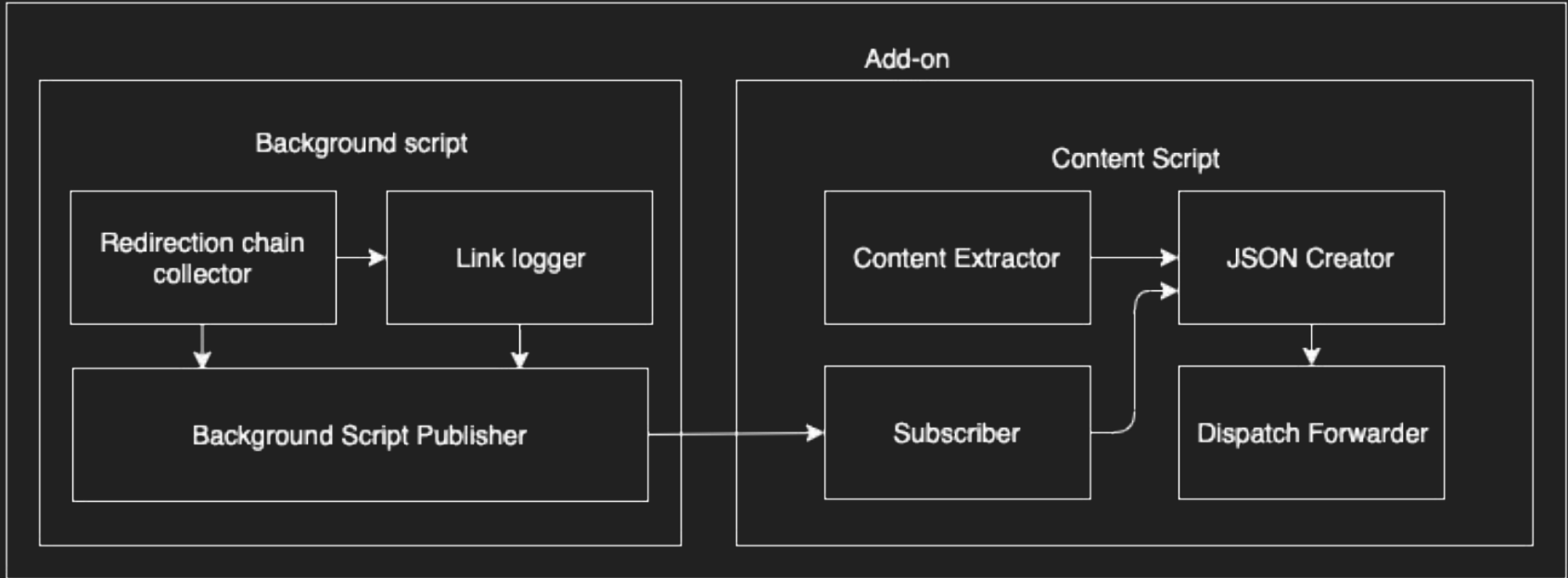- Static observations

SYSTEM ARCHITECTURE
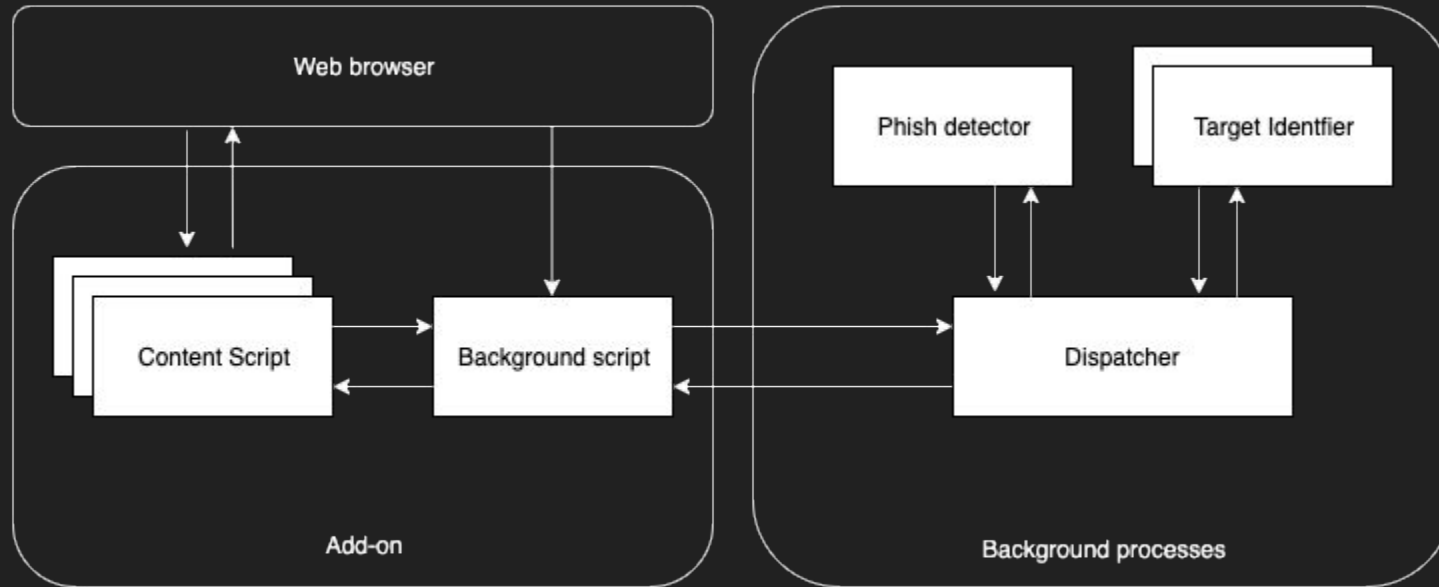
HIGH LEVEL
BLOCK DIAGRAM
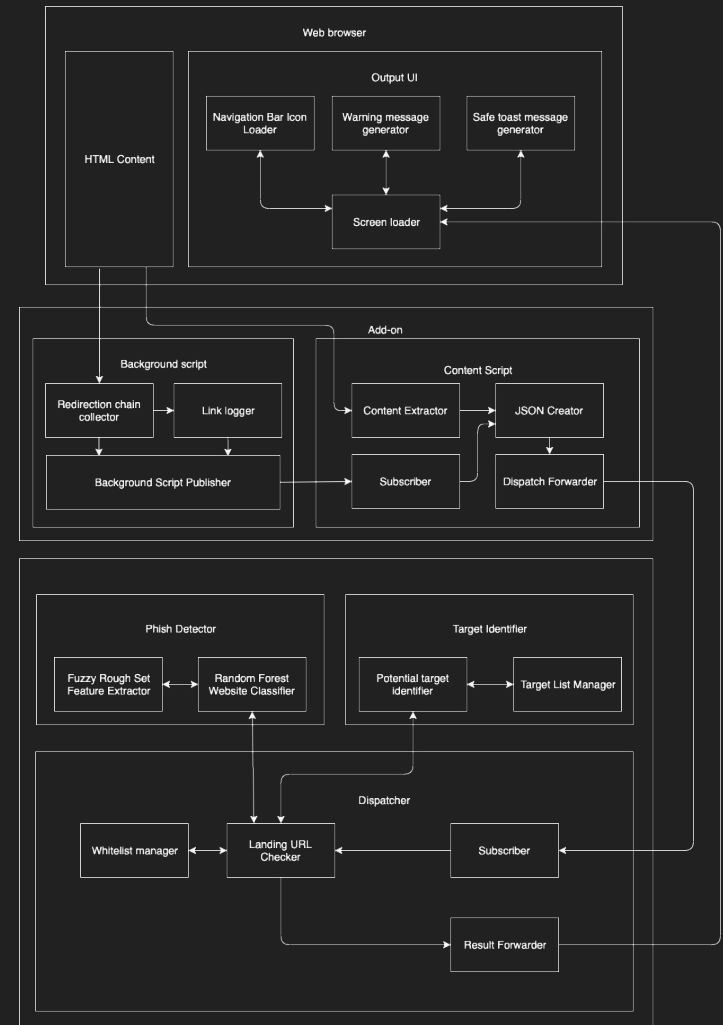
MODULE LIST

- Add on
  a. Background script
  b. Content script
- Background process
  a. Dispatcher
  b. Phish Detector
  c. Target Identifier

- Web Browser
  a. HTML content
  b. Output UI

ADD-ON

BACKGROUND SCRIPT

Begin

    For each page load redirect

        Add listener to that event

        Get the list of redirects from listener

        If page is fully loaded

            Send the list of redirects to content script

    Done

End

CONTENT SCRIPT

Begin

    For each page load redirect

        If page is fully loaded

            Get the URL from the tab

            Get the HTML content from innerHTML tag

            Get redirection list from background script

        Send them to the background process

    Done

End

BACKGROUND PROCESS
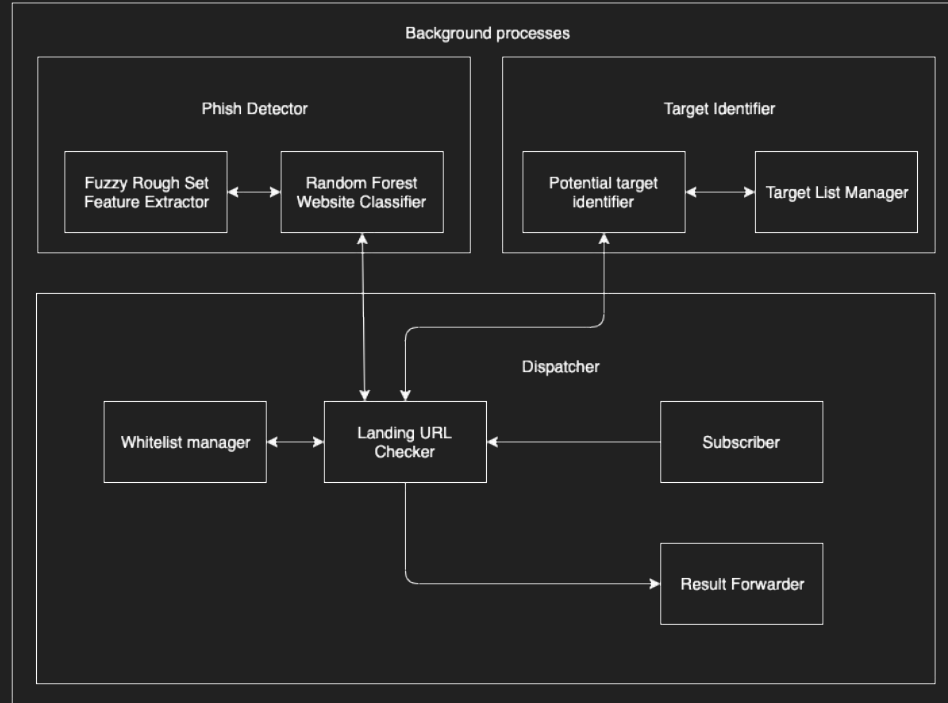
DISPATCHER

Begin
If page address is in whitelist
    Send the GREEN signal
Else
    Send content to phish detector
    Get results from phish detector
    If phish is FALSE
        Send the GREEN signal

Else
    Send the RED signal
    Send content to target identifier
    If target is found
        Publish target
    Else
        No target matched
End

PHISH DETECTOR

Begin

For each page URL

Get the fuzzy set feature values for the URL

Load the saved random forest model

Publish the result

Done

End

RANDOM FOREST MODEL

Begin

    For each record in dataset

        Get the fuzzy set feature values

        Create an arff file to save results

    Done

    Train the dataset with at least 7 splits as random forest

    Save the model as pkl file

End

TARGET IDENTIFIER

Begin

   Get the hash value for page content

   Compare with values in hash list

   If match

      Display target

   Else

      No target found

End

WEB BROWSER
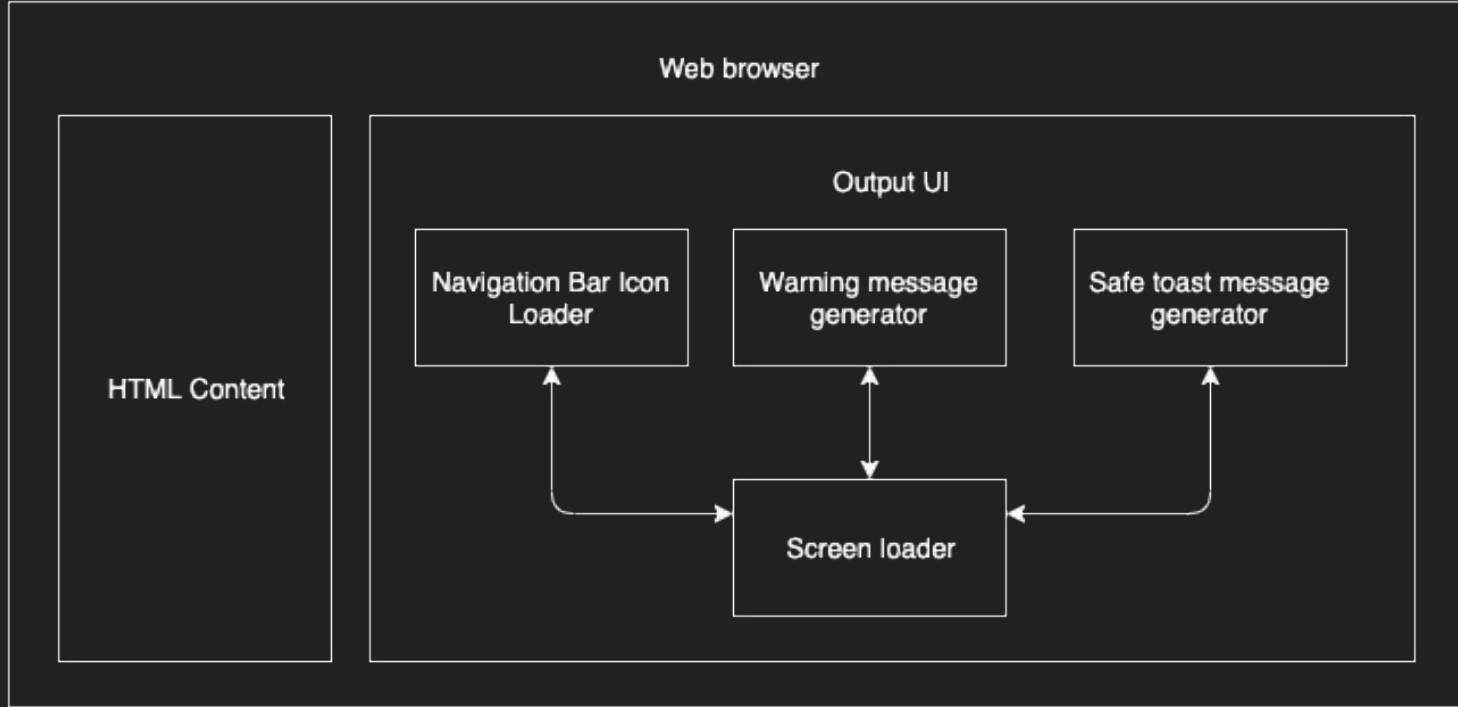
OUTPUT UI

Begin

    If site is phish

        Change icon to red

        Display warning message

        If site has target

            Display target link

        Else

            Display no target

    Else

        Change icon to green

        Display safe to proceed message

End

IMPLEMENTATION

- CONTENT SCRIPT
- BACKGROUND SCRIPT

CONTENT SCRIPT
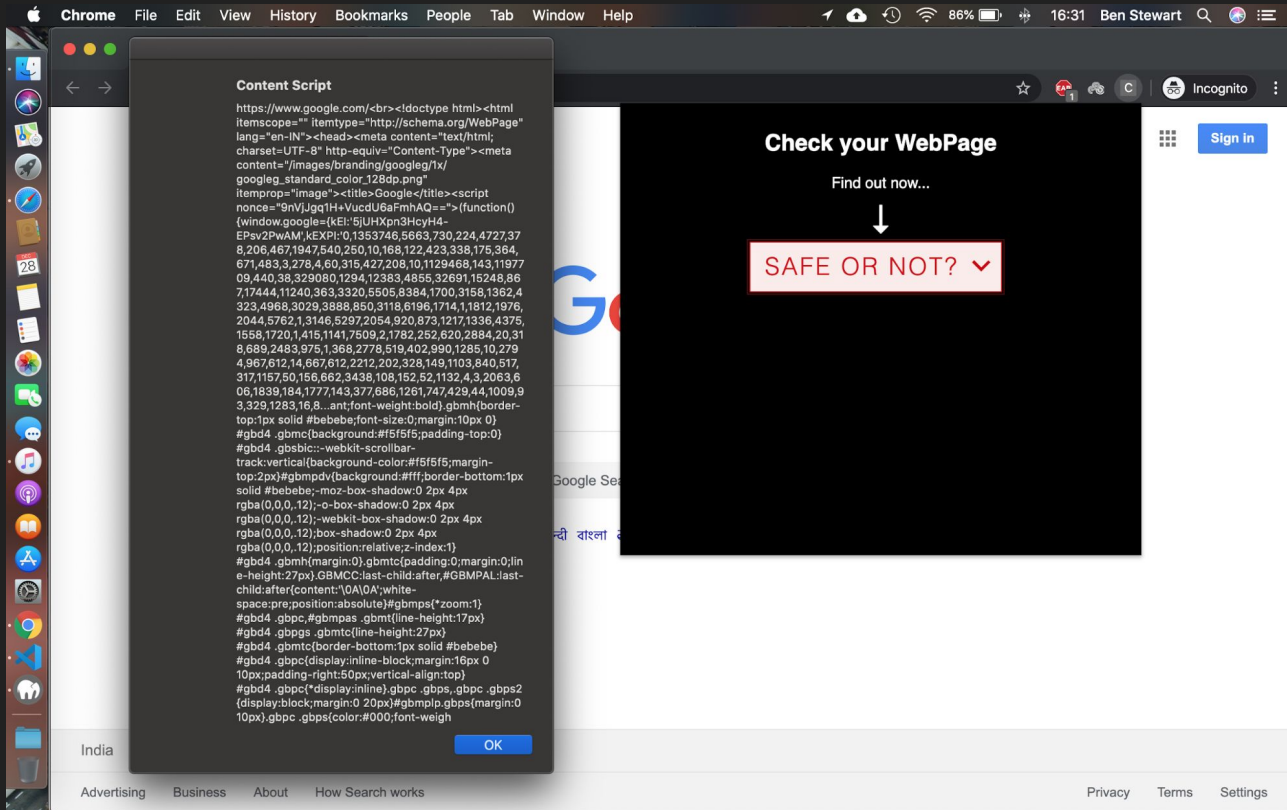
CONTENT SCRIPT

```
//Retrieve URL JS
tablink = tab.url;


//Retrieve Page content PHP
$site=$_POST['url'];
$html = file_get_contents($site);
```

BACKGROUND SCRIPT

BACKGROUND SCRIPT

```
//URL path item
url: pathItem.url,
status: pathItem.status_line,
redirect_type: pathItem.redirect_type,
redirect_url: pathItem.redirect_url,
meta_timer: pathItem.meta_timer
```

EVALUATION METRICS

1. Phish detection accuracy

Accuracy = (TP+TN)/(TP+TN+FP+FN)

2. Target detection ratio

Detection Ratio = (TP)/(TP+TN+FP+FN)

3. Memory usage profiling

Current total memory usage = Total Memory - (Free + Buffers + Cached)

4. Addon rendering time

Rendering Time = End time - Start time

5. Temporal resilience accuracy

Accuracy = (TP+TN)/(TP+TN+FP+FN)

REFERENCES

1. Mahdieh Zabihimayvan and Derek Doran, "Fuzzy Rough Set Feature Selection to Enhance Phishing Attack Detection", IEEE International Conference on Fuzzy Systems, June 2019.
2. S. Marchal, G. Armano, T. Gröndahl, K. Saari, N. Singh, N. Asokan, "Off-the-hook: An efficient and usable client-side phishing prevention application", IEEE Trans. Comput., vol. 66, no. 10, pp. 1717-1733, Oct. 2017.
3. A. K. Jain, B. B. Gupta, "A novel approach to protect against phishing attacks at client side using auto-updated white-list", EURASIP J. Inf. Secur., vol. 2016, no. 1, Dec. 2016.
4. G. Xiang, J. Hong, C. P. Rosé, L. Cranor, "CANTINA: A feature-rich machine learning framework for detecting phishing Web sites", ACM Trans. Inf. Syst. Secur., vol. 14, no. 2, 2011.
5. Implementation for the Usage of Google Safe Browsing APIs (v4), 2019, [online] Available: https://github.com/google/safebrowsing.
6. C. Whittaker, B. Ryner, and M. Nazif, "Large-scale automatic classification of phishing pages," in Proc. Netw. Distrib. Syst. Security Symp., 2010, pp. 1–14.