International Journal of Engineering & Technology, 7 (4.36) (2018) 762-765

International Journal of Engineering & Technology

Website: www.sciencepubco.com/index.php/IJET

Research paper

A Method for Information Grabbing, Bypassing Security and Detecting Web Application Vulnerabilities

B.J. Santhosh Kumar[1*], B.R. Pushpa[2]

[1]Department of Computer Science, Amrita Vishwa Vidyapeetham, Amrita School of Arts and Sciences, Mysuru, Karnataka, India.
[2]Department of Computer Science, Amrita Vishwa Vidyapeetham, Amrita School of Arts and Sciences, Mysuru, Karnataka, India.
E-mail:preeths1@gmail.com
**Corresponding author E-mail:santhoshbj50@gmail.com, bj_*

Abstract

A single file on web contains text, images, audio, video and formatting instructions enclosed within a script. Website files are hosted on servers. The Servers "serve" those files to individual users upon request. Anonymous user with minimum user credentials can request on behalf of legitimate user to grab sensitive, confidential and personal information without legitimate users knowledge.[3] The proposed method makes use of URL as input for finding web vulnerabilities. Testing of proposed method is conducted to evaluate the performance based on the accuracy received. Performance is evaluated based on false negative and false positive results. Experiment is also conducted for web vulnerability assessment and penetration testing. The proposed method also checks for information grabbing from web using Google dork. Google dork helps to enter a network without permission and/or gain access to unauthorized information. Advanced search strings called Google dork queries used to locate sensitive information. This paper describes the method for web application vulnerabilities detection by using google dork, bypass first level security in any web and hack username and password in social networking site.

Keywords: *URL (Uniform Resource Locator), Google Dork, HTTP (Hyper Text transfer protocol, SQL (Structured Query Language), Open Web Application Security Project (OWASP), XSS (CRSOSS SITE SCRIPTING)*

1. Introduction

Web browser is a software application using which we can perform most of the internet-based activities. Many web applications provide extensions to browsers to enhance their functionality, while some of the extensions perform malicious activities to get access to the sensitive data without the user's knowledge [3]. This paper explains the web vulnerabilities detection using string comparison technique [1]. The application makes use of accepting URL as input and detects vulnerabilities in websites. The application bypasses the first level security in websites by providing login. Website files contains text, images, audio, video and formatting instructions enclosed within a web script. Most of the web script provide user details uploaded on to the servers. The Servers response user will requested page. The application will grab sensitive, confidential and personal information without legitimate users knowledge and finally detecting a method for hacking username and password in social networking site. The proposed method also checks for information grabbing from web using Google dork. Google dork helps to enter a network without user permission and/or gain access to unauthorized information. Advanced search strings called Google dork queries used to locate sensitive information. Experiment is conducted for web vulnerability assessment and penetration testing.

2. Proposed system

The two schemes that are associated with the application are:

(i) Web vulnerabilities detection
(ii) Bypass first level security in website
(iii) Web content vulnerability

The data is stored on to web servers in different formats. Data grabbing technique allows anyone to grab personal and sensitive information. Untrusted data is most often data that comes from the HTTP request, in the form of URL parameters, headers, or cookies. From a security perspective, data captured from web databases, web services, and other sources is frequently untrusted. That is, untrusted data is input that can be manipulated to contain a web attack payload. A vulnerability is detected and passive or active attack happens through untrusted data. The interconnected applications enforces downstream interpreter to decode. An untrusted data is concatenated with safe data and transformed, validated and encoded in different ways. An untrusted data is breakdown into parts and combined with actual application. This makes an injection problems very difficult to identify. These untrusted data inserted into a command or query or any other form of acceptance. The proposed method checks for information grabbing from web using Google dork. Google dork helps to enter a network without user permission and/or gain access to unauthorized information. Anonymous user with minimum user credentials can request on behalf of legitimate user to grab sensitive, confidential and personal information without legitimate users knowledge. Advanced search strings called queries used to locate sensitive information. The proposed method finds Web vulnerabilities detection by accepting URL as input.

3. Methodology

Parsers: - Injection attacks target web parsers which interprets source, attempting to trick them into interpreting data as commands. The key successful injection attacks lies in understanding how a particular interpreter's parser works and ultimately, the path to creating defences against injection. An undetected passive attack method can return many information about a user using google dork. It can return public information like username, email, personal information about user and few sensitive information. The leaked information can be used for any number of illegal activities including identity theft, industrial espionage, cyber terrorism and cyber stalking.

Google search operators are used for high powered google hack search terms. Inbuilt search operators can retrieve many user information from web using a google dork queries. Advanced search operators can narrow dowm search results.

i) Intitle:-searches for titles like user name, ID and return the results from web.
ii) Allintitle:- same as intitle but searches multiple terms in web and return the result.
iii) inurl:- The listed term is searched in the url.
iv) allinurl:- All listed term is searched in the url.
v) file type:- Searches for specific file types. It searches for acrobat file extension in websites.
vi) ext:- Similar to filetype. ext:pdf finds pdf extension files.
vii) intext:- Searches the content of the page. Somewhat like a plain google search. For example intext:"index of /".
viii) allintext:- Searches for all texts present in the url.

A. Case 1: intitle

intitle:workshop;https://www.somesite.edu;

This will display all workshop conducted in the website in order.

allintitle:Similar to intitle but searches for all the specified terms in the title.

B. Case 2:inurl

inurl:https://www.somesite.edu;email:

This will display all email IDs present in the website.

inurl:https://www.somesite.edu;contact:

This will display all contact details present in the website.

allinurl: Similar to intitle but searches for all terms in the URL.

C. Case 3: filetype

ebook:https://www.somesite.edu;filetype:pdf

This will display all ebook with pdf extension present in the website.

D. Case 4:intext

inurl:https://www.somesite.edu;index.html?id=;intext:"Address"

This will search entire website for keyword Address and display all Address present in website.

allintext: Similar to intext but searches for all text content of the page.

User can make own formula of Google Dorks and can also use combination to get better result.

Google dork search criteria in which a search engine returns results related to dork.

Time taken by the process may vary depending on the type of query but outcome is worth.

You can use following words instead of inurl :

Intitle,inurl,intext,define,site,phonebook,maps,book,Froogle,info,movie,weather,related,link.All these also help to find other things present in webpage. Data grabbing few examples.

inurl:https://www.somesite.com;intitle:"Name";info

Returns name with few information of the person.

inurl:https://www.somesite.edu;index.html?id=;intext:"Address"

Returns ID with address information of the person.

inurl:https://www.somesite.com;index.html?=intitle:"Name";intext:"Address"

Returns the name and address of the person.

inurl:https://www.somesite.edu;filetype:pdf;intitle:workshop;intext:"session on /"

Returns the workshop or session details saved in .pdf format

inurl:https://www.somesite.edu;intitle:seminar;

Returns the seminar details present in website.

inurl:https://www.somesite.edu;intitle:workshop;intext:"workshop on /"

Returns the workshop details present in website.

inurl:https://www.somesite.com;intext:"@gmail.com";intitle:Name

Returns the gmail ID of the person present in website.

inurl:https://www.somesite.com;intext:Name;

Returns the name of the person present in website.

inurl:https://www. somesite.com; intitle: organisation Name; intext:Employee Name;

Returns the name of the person present in website with public details.

inurl:https://www.somesite.com;intitle:"firstname" | "lastname";intext:Organisation;

The above e technique grab URL's or scan URL's parameters for vulnerabilities with 'scan'. Stolen credentials can be used to perform advanced passive or active attacks. The experiment done on popular social networking site to retrieve information. The result was accurate with low false negative and high false positive. The method makes use of search engines with built-in functions in URL. Google dorks scan for interesting files and disclosures of full path. Using list of URL's, scanner will scan for CSS [2], Remote File Inclusion, SQL Injection and vulnerabilities in Local File Inclusion [4].Difficult to detect and print XSS attacks.[5]. It is possible to perform mass brute force attacks for specific range of hosts. Whenever something helpful and interesting things found, like vulnerability or broken authentication credentials, data will be saved in text files.

- Info grabbing from social networking site
  - Automatic darking making use of 'dork.
  - Full path disclosure grabbing making use of 'fpds'
  - searching someone in databases using commmon and advanced 'search'
  - Scanning all top websites of specific nation using 'top'

inurl:https://www.socialnetwork.com;intitle:"username";intext:organisation;

https://www. socialnetwork.com/username/about

This will return entity unique ID and username with other details.

Fbid : 10000702684870
profile_id: 72053734
ip address : 173.252.120.68
Person name : Name
jobTitle : Designation
Locality : place
Organization name: Company

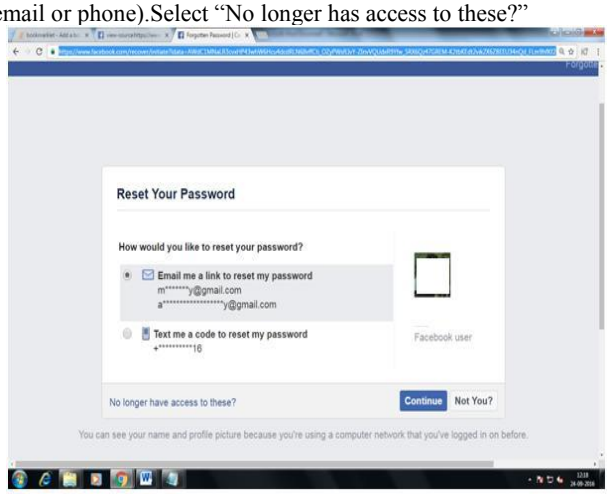
Work experience, education details, likes, favorites and close friends with photos from webpage source. Many information's are leaked from the page.

This can also be verified in http://findmyfbid.com site.

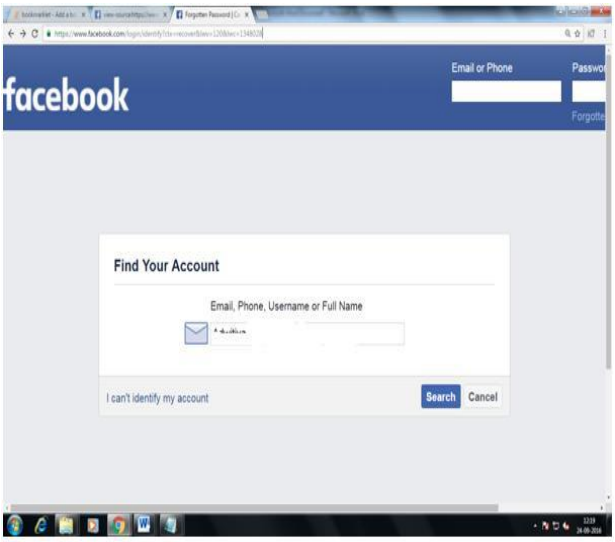
This site will accept a URL as input in search engine and will return unique correct facebook personal numeric ID.

3.1.Finding vulnerable website

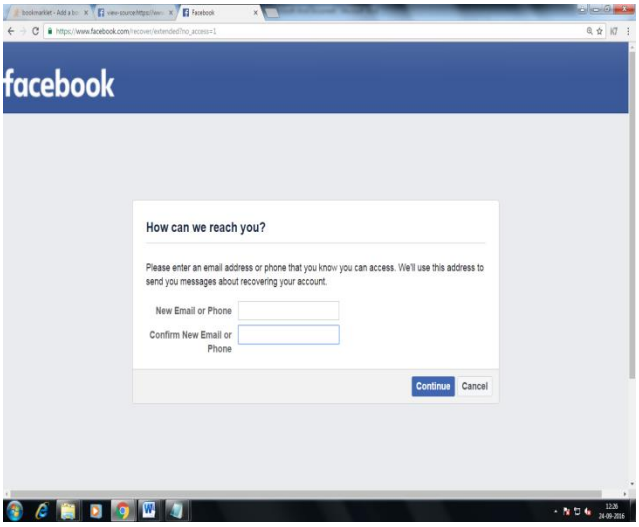
Hacking Facebook account
Most of user passwords are guessable. Passwords are short or based on user attributes/credentials.
Steps 1: select forgot password
User will be prompted with resetting password page. User can select any of the checkbox for receiving password reset link (email or phone).Select "No longer has access to these?"



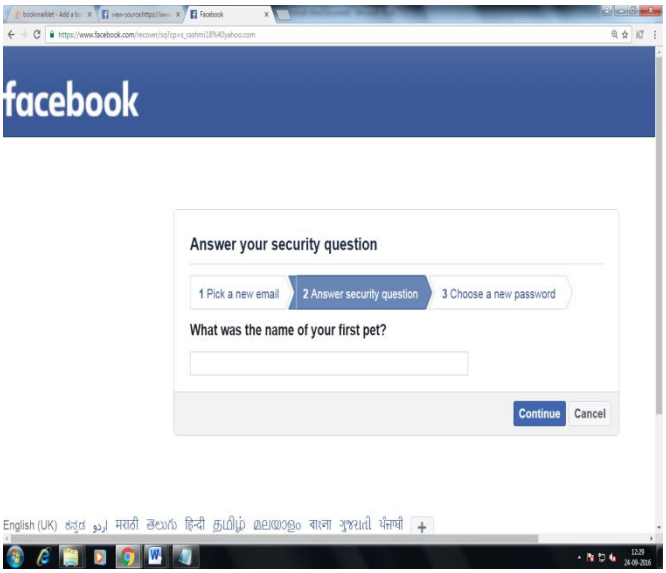
Step 2: Entering a username in textbox.



Step 3: This will accept new Email ID or phone. Entering new Email ID and phone number. IF user enters legitimate email or phone number then user will get a mail or message indication of resetting passwords.

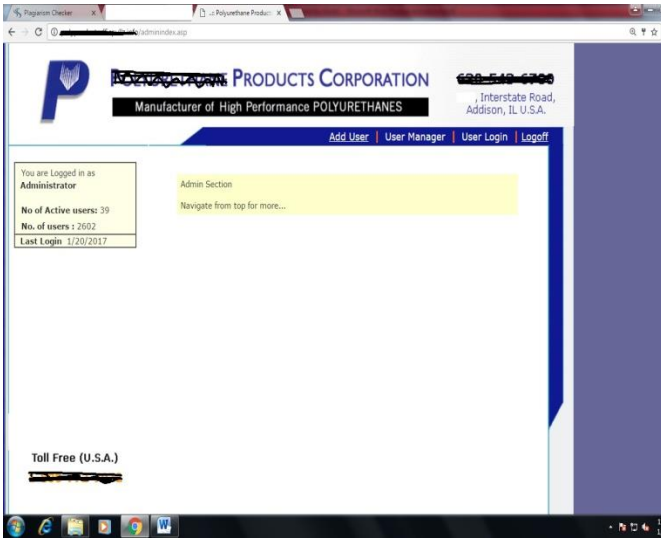


Step 4: This webpage will query few questions. The questions given by the legitimate user. After answering two or three questions the page will allow user to choose new password. This is the last step to change legitimate user password without any intimation.



3.2.Finding vulnerable website

The hacker can also find Vulnerable Websites which bypasses admin login. In search engine user can type "admin login.asp or admin login.aspx or admin login.php". This results in listing number of websites and user can type "admin" in admin login page and password can be "1'or'1'='1" This is helpful to check vulnerability in website logins.
Provide Username : Admin
Password : 1'or'1'='1



The above credentials will bypass the first level security and will get a logged in page. Anonymous user can view and navigate menu with contents and also can change the admin password based. The experiment done on basic and advanced regular expression to bypass login. Few examples

+'or'1'='1
?'or'1'='1
(a-z)(A-Z)'or'1'='1
(a-z)(A-Z)(0-9)'or'1'='1
{a-z}'or'{A-Z}'or'{0-9}'='1
^(?=.*[A-Za-z])'or'1'='1
^(?=.*[A-Za-z0-9])'or'1'='1

4. Security of web

Most of the web contents are vulnerable to anonymous user who has access to internet. There are different classes of web vulnerabilities namely SQL injection, Cross-site scripting, Authentication-Authorisation, Buffer errors, Path (directory) traversal, Code injection, Information leak/Disclosure, Cross-site request forgery. Web security requires immediate attention which involves protecting information by preventing, detecting and responding to attacks.Many of the web attacks are unreported.
Best practices to protect websites:-

1. Do not use the default login and password.
2. Do not allow special character in password to prevent SQL injection
3. Use alpha numeric password and have a two step verification
4. Change/Rename the URL of the admin login page.

5. Conclusion

I have considered a new requirement of finding vulnerable website. A method for web application vulnerabilities detection using advanced string matching technique and method to bypass first level security in website. Using Google dork for retrieving confidential and personal information. Advanced search strings called Google dork queries used to locate sensitive information. Advanced methodology with tools can be used to grab batch information from web.

Future enhancement

Advanced methodology with tools can be used to grab batch information from web. Using tools Hydra, Kali Linux to hack user accounts.

References

[1] Saleh AZM, Rozali NA, Buja AG, Jalil KA, Ali FHM & Rahman TFA, "A method for web application vulnerabilities detection by using boyer-moore string matching algorithm", *Procedia Computer Science*, Vol.72, (2015), pp.112-121.
[2] Jevitha KP & Vishnu BA, "Prediction of Cross-Site Scripting Attack Using Machine Learning Algorithms", *International Conference on Interdisciplinary Advances in Applied Computing*, (2014), pp.1-6.
[3] Arunagiri J, Rakhi S & Jevitha KP, "A Systematic Review of Security Measures for Web Browser Extension Vulnerabilities", *International Conference on Soft Computing Systems*, Vol. 2, (2016), pp.99–112.
[4] Gupta S & Gupta BB, "XSS-SAFE: a server-side approach to detect and mitigate cross-site scripting (XSS) attacks in JavaScript code", *Arabian Journal for Science and Engineering*, Vol.41, No.3, (2016), pp.897-920.
[5] Singh P, Thevar K, Shetty P & Shaikh B, "Detection of SQL Injection and XSS Vulnerability in Web Application", *International Journal of Engineering and Applied Sciences (IJEAS)*, Vol.2, No.3, (2015), pp.16-21.

Text books

[1] William S, *Cryptography and network security: principles and practice*, Prentice-Hall, Inc, (1999), pp.23-50.
[2] Kahate, A, *Cryptography and network security*. Tata McGraw-Hill Education, (2013).