Secure Search Engine for Mobile Users for Countering the Attacks over Internet

Mohan Gowda G S, Janardhan Singh

Abstract*— Mobile user browsing internet is vulnerable to internet attacks. Security is the very important issue for any mobile user. Eventhough there are many security solutions to overcome internet attacks, these solutions may rely on human factors to achieve a good result against phishing websites, SSLStrip-based, Man-In-The-Middle attack and Spam detection. This paper presents a secure web referral service, which is called Secure Search Engine (SSE) for mobile devices. This method uses mobile cloud-based virtual computing and provides each user a Virtual Machine (VM) as a personal security proxy where all Web traffics are redirected through it. Inside the VM SSE uses web crawling technology with a set of checking services to validate IP addresses and certificate chains. Phishing Filter is used to check given URLs in a minimum execution time. This approach uses seperate private caches to protect user privacy and improve performance.

Index Terms*— Security, SSL Strip, Man in the middle, Mobile cloud.

I. INTRODUCTION

Using mobile devices, the web-based communications have become easy targets for attackers to compromise end-to-end communications, e.g., using Man-in-the Middle(MITM) [1] attacks and deploy malicious phishing web sites to delude Internet users to expose their private information. Cryptography enhanced Internet protocols have been widely used to protect Internet users from being attacked. Strong Cryptographic algorithms cannot be used in mobile devices because of the limited processing power, memory, and battery constraints. In this paper the focus is on security concerns security problems incurred due to human errors: e.g., SSLStrip MITM attack and web based phishing attack, which require users to be involved to make decisions on accepting or rejecting a potentially breached web site. In SSLStrip attack, attackers explore the vulnerability that a user may request a secure web site by initiating an insecure HTTP request. Phishing is the act of attempting to acquire sensitive information such as usernames, passwords, and credit card details sometimes, by masquerading as a trustworthy entity in an electronic communication. This work presents a Secure Search Engine (SSE), for mobile devices. The system uses mobile cloud based virtual computing and provides each user a VM as a personal security proxy where all Web traffics are redirected through it. Each VM is isolated from other VMs to protect the user's privacy. Within the VM, the SSE checks the Web traffics for potential MITM or phishing attacks. SSE is designed as an automatic web referral service involving minimal interventions from humans for security decisions.

Manuscript Received on July 2014.

Mr. Mohan Gowda G S, M.Tech student of Cambridge Institute of Technology, Bangalore, India

Mr. Janardhan Singh, Asso. Prof. in Cambridge Institute of Technology, Bangalore, India.

II. EXISTING SYSTEM

There are many existing systems that tries to prevent the Man in the middle attack and phishing attack.

ForceHTTPS [2]: this approach forces the browser to open a secure connection to the destination. If the destination does not support an SSL connection, then the user needs to manually set a policy in ForceHTTPs.

Disadvantage : This approach does not prevent MITM attacks since attackers can intercept the HTTPS request and return a no-HTTPSsupport‖ message and force the web browser to initiate HTTP sessions.

Phishing filter CANTINA [3]: this approach make use of robust hyperlinks and *t*erm- *f*requency - *i*nverse *d*ocument frequency where *tf – idf* is a technique to give less importance to the common occurring words. When a URL is fed into CANTINA, it first calculates the *tf − idf* scores for the page, then it calculates the lexical signature using the top-5 *tf−idf*, and finally it uses Google Search engine to check if the web site is in the top *N* results

Disadvantage : CANTINA depends on the Google's Crawler. When a new web site is up, it can stay online for approximately 53 hours that cannot be crawled by Google's search engine immediately.

Tahoma[4]: uses a browser OS running on top of a client-side Xen-managed virtual machine to serve as a local proxy to scan web applications.

Flashproxy[5]: targeted the performance and security of Flash object browsing on mobile devices.

Disadvantage : These approaches share virtual machines among users, the user's privacy may be an issue and inncur overheads when switching from one virtual machine to another.

III. PROPOSED SYSTEM

The architecture of proposed system is as shown below

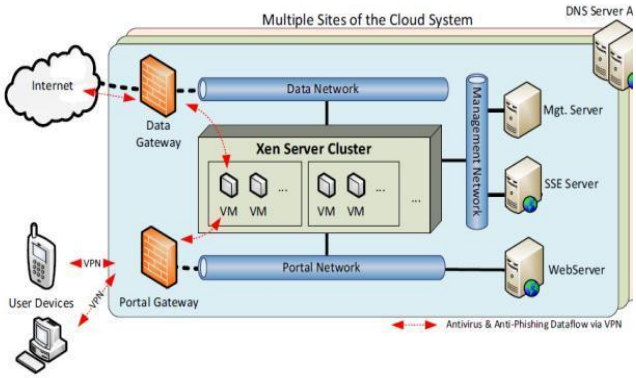

Fig. 1. System Components of the SSE Mobile Cloud System.

- **Xen Server Cluster:** The Xen server cluster is a set of Xen servers that provisions VM resource pools [6].
- **Web Server:** The web server hosts a website to provide a management portal for users and system administrators.
- **Mobile User VM:** Each mobile cloud user has a dedicated VM that incorporates several components to provide features such as http proxy, caching and logging.
- **Portal Gateway & Portal Network**: The Portal Gateway is the access point for mobile users to access internal VMs and web services.
- **SSE & Management Server:** SSE server provide the SSE service and the Management Server is in charge of the system resource allocation.
- **Data Network & Data Gateway:** The networks that are used by VMs and SSE fetch web data from the Internet and send to the user mobile devices.

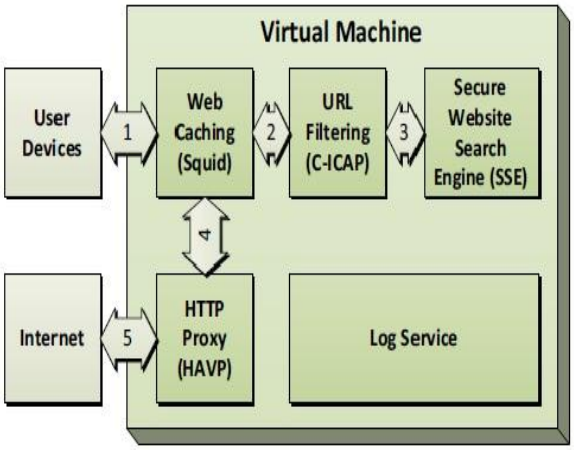

Fig. 2. Components of the user VM.

A. Secure Search Engine

1) **Secure Search Engine Architecture***:* The SSE is a cloud service that can be used by each mobile user VM. To provide web proxy and caching functions, a mobile user VM incorporates multiple components, as shown in Fig. 3.

The SSE is implemented as a layered approach as shown in figure 3.

Steps involved between VM and SSE service

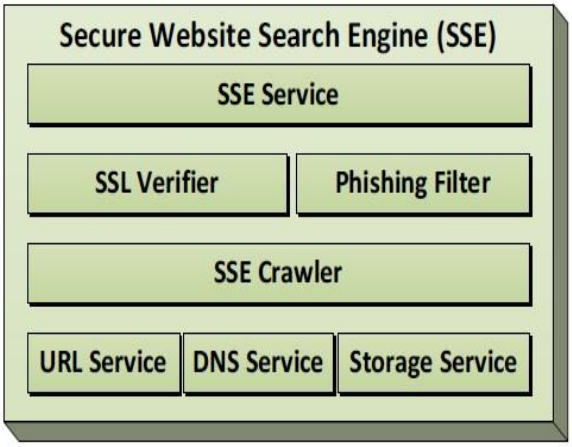

Fig. 3. SSE Service Models.

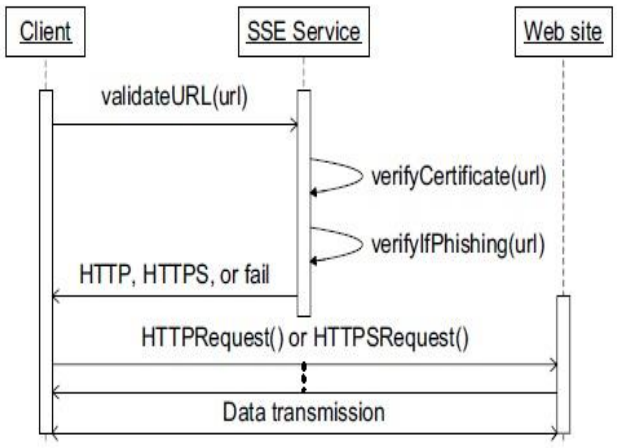

Fig. 4. Steps involved between a VM and the SSE service.

2)Procedures of using SSE Service: When the SSE receives a web request, it inspects whether the requested URL links to a phishing site or not. a)SSL verifier validates if the web server provides valid certificates and supports HTTPS for the requested domain. b)Phishing filter identifies identifies phishing properties of the inspected web page.

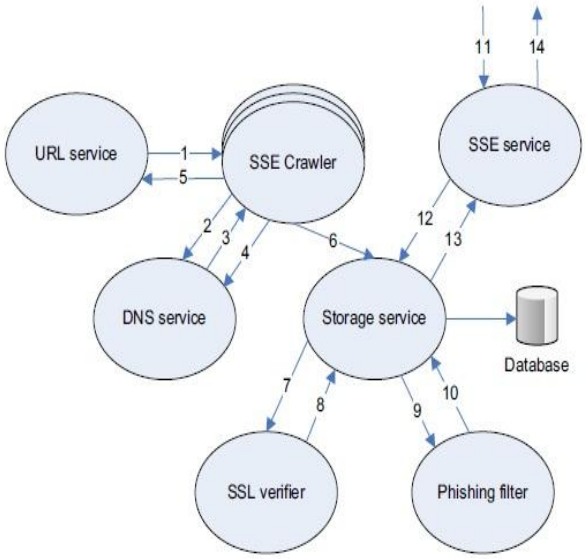

Fig. 5. Dataflow within the Secure Search Engine.

Fig. 5. presents the data flows within the SSE server.

Step: (1) Crawler gets the unprocessed URLs from the URL service.

Step: (2 & 3) Crawler requests and gets the rewritten URL.

Step: (4 & 5) After crawling the new URL, Crawler updates the DNS cache and URL service with the extracted URL.

Step: (6) The collected results are persisted using the storage service.

Step: (7 & 8) The SSL verifier collects, validates, and then stores the certificates.

Step: (9 & 10) The Phishing Filter runs the algorithm to find out whether a web site is a phishing site or not.

Step: (11) The user VM requests the SSE to verify a web site.

Step: (12 & 13) The SSE service uses the storage service to get the corresponding data to evaluate.

Step: (14) The SSE service answers the query.

B) Countering SSLStrip MITM Attacks:

To counter SSLStrip MITM attack, the SSE automates the security inspection procedure and the user will be notified only if potential MITM attack are detected. This minimizes the human factors in the process.

Algorithm 1 : SSL Verifier

```
urls = extractAllHttpsUrls();
for each URL in urls do
  certChain = getCertificate();
  if certChain is valid then
    for each certificate in the certChain
     do params = extractCertParameters();
       store(params, certificate);
    end for
    else
    mark invalid certificate against the URL;
  end if
end for
```

The SSL verifier gets the unprocessed URLs from the storage use *extractAllHttpsUrls*(). For each URL, the SSL verifier checks for an SSL connection to the server. If an SSL connection request is accepted by the server, then it inspects the certificate chain and validates each certificate in the chain towards the site certificate. If the web site rejects the HTTPS request, or the web site does not have a valid certificate, and thus no support for HTTPS.

C) Countering Web-based Phishing

Phishing Filter checks with the SSE database and see if the IP address has hosted any phishing site in the past. The filter checks if the site has valid certificate and its Google page rank. These components are choosen for following reasons.

(i)Phishing sites usually do not provide valid certificates since the process for obtaining a certificate is not desired for phishing attackers.

(ii)Google page rank is calculated using many parameters to evaluate the popularity of a webpage.

Algorithm 2: Phishing Filter

```
for web page i do
  Bayes classifier will return with a probability of a
  web site being a phishing site
  P(i)=getProbabilityForPhishing(URL);
  Cert(i) = hasValidCertificate(URL);
  GPR(i) = getGooglePageRank(URL);
  Compute IP(i) = 1-1/logt;
  Compute the Confidence value for page i;
  if Confidence(i) <= confidenceThresholdr return
  phishing site found;
  else
        return trustable site;
  end if
end for
```

Probabilty of web page i, being a phishing site can be calculated as follows

Let *t* represent the number of days that an IP address has not hosted the phishing site.

$$IP(i) = \begin{cases} 1 - \frac{1}{logt}, & if\ t > 1; \\ 0, & otherwise \end{cases}$$

Increase in value of *IP(i)* would increase the value of *confidence.*

D) Spam Detection:

Spam web pages intend to achieve higher-than-deserved ranking. Human experts could easily identify spam web pages, but the manual evaluating process of a large number of pages is still time consuming and cost consuming. To assist manual evaluation, we propose an algorithm to assign spam values to web pages and semi-automatically select potential spam web pages. We first manually select a small set of spam pages as seeds. Then, based on the link structure of the web, the initial R-SpamRank[7] values assigned to the seed pages propagate through links and distribute among the whole web page set. After sorting the pages according to their R-SpamRank values, the pages with high values are selected. Experiments and analyses show that the algorithm is highly successful in identifying spam pages.

$$RSA\ (A) = (1 - \lambda)I(A) + \lambda \sum_{i=1}^{n} \frac{RSR\ (T_i)}{C(T_i)}$$

$$I\ (A) = \begin{cases} 1, & if\ A\ in\ the\ blacklist \\ 0, & otherwise \end{cases}$$

where *RSR(A)* is the R-SpamRank value of page A, λ is a damping factor, I(A) is the initial value for page A, n is the number of forward links of page A, and *Ti* is the ith forward link page of page A, C(*Ti*) is the number of in links of Page *Ti*; *RSR(Ti)* is the R-SpamRank value of page *Ti*.

IV. PERFORMANCE EVALUATION OF PHISHING FILTER

Two parameters are used to evaluate the Phishing Filter - false positives and false negatives. To show the effectiveness of SSE Phishing Filter, the paper compares it with built-in phishing filters of Firefox 3.5, Chrome 2.0, and IE 8.0.

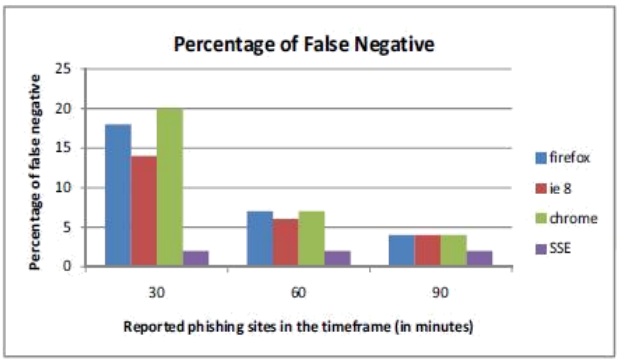

Fig. 4 Percentage of the False Negative.

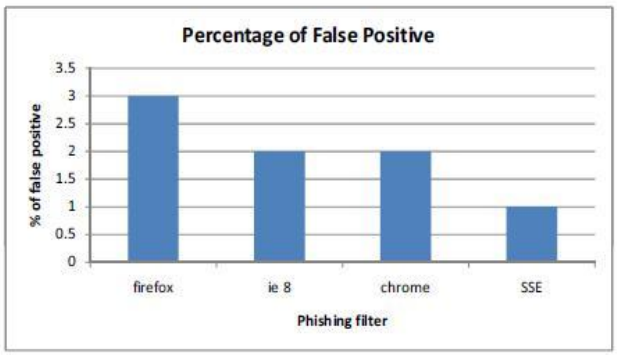

Fig. 5 Percentage of the False Positive.

It is proved from the results that SSE provides better phishing results than the existing approaches that the browsers use.

IV. CONCLUSION AND FUTURE WORK

Mobile cloud based Secure Web referral service provides an effective search engine to counter Web – based Man in the middle, phishing attack and spam detection. SSE is designed to be non-intrusive in nature and consumes no resource on the mobile device and involves minimum human interventions to of humans for security decisions. SSE phishing filter produces low false possitives and false negatives. In the future SSE can be expanded to counter any other types of attacks.

REFERENCES

1. M. Moxie, Sslstrip software, ―http://www. thoghtcrime.org/software/sslstrip. 2009
2. C. Jackson and A. Barth, ―ForceHTTPS: Protecting high-security web sites from network attacks, 2008.
3. Y. Zhang, J. I. Hong, and L. F. Cranor, ―Cantina: a content-based approach to detecting phishing web sites.
4. R. Cox, J. Hansen, S. Gribble, and H. Levy, "A safety-oriented platform for web applications," in *Security and Privacy, 2006 IEEE Symposium on*, may 2006, pp. 15 pp.
5. A. Moshchuk, S. D. Gribble, and H. M. Levy, "Flashproxy: transparently enabling rich web content via remote execution".
6. Xen, ―Xen Virtualization Open Source Project.[Online]. Available: http://xen.org
7. R-SpamRank: A Spam Detection Algorithm Based on Link Analysis

AUTHORS PROFILE

Mr. Mohan Gowda G S is an M.Tech student of CAMBRIDGE INSTITUTE OF TECHNOLOGY, Bangalore, India Presently he is pursuing his M.Tech in Computer Network Engineering from this college and he received his B.Tech degree from Dr. Sri Shivakumara Swamy College of Engineering affiliated to VTU University, Bellary in the year 2012. His area of interest includes Cloud computing, Computer networks and current trends and techniques in Computer Science.

Mr. Janardhan Singh is working as Associate Professor in Cambridge Institute of Technology, Bangalore. He received his Masters degree from Visvesvaraya university. His area of interest includes Cloud computing, Wireless Sensor networks.