ISSN: 2229-6948(ONLINE)
DOI: 10.21917/ijct.2016.0203

EVALUATING EFFECTIVENESS OF MOBILE BROWSER SECURITY WARNINGS

Ronak Shah[1] and Kailas Patil[2]
[1,2]Department of Computer Engineering, Vishwakarma Institute of Information Technology, India
E-mail: [1]ronnshah92@gmail.com, [2]kailas.patil@viit.ac.in

Abstract

This work precisely evaluates whether browser security warnings are as ineffective as proposed by popular sentiments and past writings. This research used different kinds of Android mobile browsers as well as desktop browsers to evaluate security warnings. Security experts and developers should give emphasis on making a user aware of security warnings and should not neglect aim of communicating this to users. Security experts and system architects should emphasis the goal of communicating security information to end users. In most of the browsers, security warnings are not emphasized, and browsers simply do not show warnings, or there are a number of ways to hide those warnings of malicious sites. This work precisely finds that how inconsistent browsers really are in prompting security warnings. In particular, majority of the modern mobile web browsers are vulnerable to these security threats. We find inconsistency in SSL warnings among web browsers. Based on this work, we make recommendations for warning designers and researchers.

Keywords:

Mobile Security, Mobile Web Browsers, Malicious Sites, SSL Warnings

1. INTRODUCTION

With the most up to date era of cell phones, web utilization on cell phones at long last hits the masses. Till now, security and protection consciousness of mobile Internet use has attracted little consideration in research and industry. In any case, with its raise, the quantity of clients that utilize those gadgets for security delicate assignments like Internet Banking raises too. Like this, security and protection mechanisms for cell phones ought to be considered in future work.

Web security and its mindfulness is a regularly talked about theme nowadays. The differing qualities and the capability of current web browser applications have very expanded in the most recent years. With this, the method for how security of such web pages is appraised and the way it is introduced to the clients has changed too. New security symbols, location bar colorization and the visualization of Extended Approval SSL declarations have been presented. Despite the fact that concentrates so far appear that even this is insufficient, no endeavours of any sort have been put into mobile browsing experience. Rather than at any rate utilizing the bits of knowledge that have been picked up in this way, browser producers for cellular telephones begin with outdated UI components (e.g. the padlock symbol). With much research done on adequacy of such warnings on standard browsers, endeavours ought to be spent to ensure clients of cell telephones too. Since more clients utilize their cell phones to search the web and read their messages, they are getting powerless when utilizing this option method for perusing. Embracing the security ideas of today's browsers is not by any means the only approach to raise security mindfulness on cell phones. Due to the distinctive equipment of those little gadgets, different ideas of raising security mindfulness get to be conceivable fusing different actuators. The important and utmost goal of this paper is to investigate whether modern mobile browser security warnings protect users in practice.

According to previous study, more than 50% users click through SSL warnings and simply ignore security measures [1]. There are many reasons why user ignores security warnings, SSL warnings and other security related warnings. Lot of work has done on desktop browsers but still there is no effective work has been done in case of mobile browsers.

Unfortunately, most of the mobile browsers did not show any security warnings while assessing through site which has a weak encryption key, a site with an invalid certificate, a site with malicious things, phishing and malware sites. If the user ignores such warnings, it can have very adverse effect on the user itself. Security attacks, Man-the-middle attack, BCP attacks, password stealing, any of the given things can happen if the user ignores security warnings.

This study finds that most of the mobile browsers almost 90% of them are inconsistent in showing security warnings. And this is major concern for today's mobile users as well as to developers.

1.1 CONTRIBUTIONS

This paper makes following contributions:

- To our knowledge, this paper gives a large-scale field study of android mobile browser security warnings.
- Examined various security warnings
- Also examined how different mobile browsers respond to all these security warnings
- This paper provides suggestions for a browser warning designers and make recommendations for future studies.

2. BACKGROUND

While browsing the web if HTTPS protocol is used, it simply establishes secured connection between user and targeted web site, in such scenario if user ignores any security warning or SSL certificate warnings, it might affect user adversely and attacker can intercept communication in the SSL channel [2]. While browsing the web, browser should notify particular user about threat. Web browsers prompt security warnings to users when a threat might be occurring. If the browser identifies certain attack or error in certificate validation, it simply prompt error page that the user should notice and should not bypass.

2.1 CERTIFICATE

Public key certificate authentications are broadly used to give keying material and pass on a site's personality data to the client. The W3C characterizes four sorts of certificates. We give our understanding to the meanings of certificate sorts in the W3C archive where they are questionable. For extra data with respect to the business routine of issuing and overseeing SSL declarations, please allude to the necessities characterized by the CA/Browser forum.

2.2 CERTIFICATE ERROR

An SSL certificate error or security certificate error shows an issue with HTTPS encryption. You'll just see this mistake when associating with a site utilizing HTTPS. At the point when utilizing HTTPS encryption, sites present certificate to recognize that they are honest to goodness. For instance, Yahoo.com has a security certificate issued by a trusted authority declaration power. The authentication power (certificate authority) checks that Yahoo is the genuine proprietor of Yahoo.com and is qualified for the certificate. When you associate with Yahoo.com utilizing HTTPS, Yahoo displays this certificate. Browser checks that the certificate was issued by a known true blue authentication power to check you're associating with the genuine Yahoo.com, not another server putting on a show to be Yahoo.com. When you see an authentication mistake, this demonstrates you're not as a matter of course interfacing with the genuine, true blue site. For instance, on the off chance that you attempt to get to your bank's site on an open Wi-Fi system and see this blunder, it's conceivable that the system is bargained and somebody is endeavouring to mimic your bank's site. Nonetheless, it's likewise conceivable that a site neglected to appropriately recharge or design its authentication. In any case, you ought not to proceed when you see this error message.

- Validated certificate
- Augmented assurance certificate
- Self-signed certificate and untrusted root certificate

2.3 W3C RECOMMENDATIONS

The World Wide Web Consortium (W3C) has characterized client interface rules for the presentation and correspondence of web security connection data to end-clients of both desktop and mobile browsers. Taking after are the rules characterized by W3C (World Wide Web Consortium)

2.3.1 Identity Signal Availability:

Primary or secondary interfaces must have security indicators that shows and verifies identity of a website [4].

2.3.2 Certificates - Required Content:

Notwithstanding the personality flag, the web browsers must make the accompanying security connection data accessible through information sources (certificates): the site's domain name and the motivation behind why the showed data is trusted (or not) [4].

2.3.3 TLS Indicators:

- Importance of indicators: Any UI pointer, (for example, the padlock) must not flag the nearness of a certificate unless all parts of the site page are stacked from servers exhibiting in any event approved declarations over emphatically TLS-ensured connections[4].
- Content and Indicator Proximity: Content must not be shown in a way that confounds facilitated substance and program chrome indicators, by permitting that substance to copy chrome markers in a position near them.
- Availability: Primary and secondary interface should contain TLS indicators at all times [7].

2.3.4 Robustness - Visibility of Indicators:

Web content must not obscure the security user interface [5].

2.3.5 Error Messages:

- Interruption: Both cautioning/alert and threat messages must interfere with the client's present assignment; such that the client needs to recognize the message [6].
- Proceeding options: Cautioning/alert messages must give the client unmistakable alternatives for how to continue (i.e., these messages must not prompt a circumstance in which the main choice introduced to the client is to release the notice and proceed) [4].
- Inhibit interaction: The communications for threat messages must be displayed in a way that makes it unthinkable for the client to go to or connect with the destination site that made the risk circumstance happen, without first unequivocally interfacing with the peril message [8].

2.4 PHISHING AND MALWARE WARNING

Your browser will likewise show phishing (or “web forgery”) [3] and malware notices. Whether you utilize Firefox, Chrome, or Internet Explorer, your browser frequently downloads a rundown/list of perilous sites. When you endeavour to interface with a site on this rundown/list, you'll see a mistake message. Sites are set on these rundowns since they contain malware or on the grounds that they endeavour to mimic a genuine site to take your passwords, credit card numbers, or other delicate data. Sometimes, a site may briefly be added to this rundown since it was traded off. At the point when the site is altered, it ought to be expelled from this rundown. When you see this message, you ought not to proceed.

3. RELATED WORK

Patil et al. [11] performed a large scale measurement of content security policy (CSP) usage in real-world web application and proposed a solution [12] to allow security savvy users to protect themselves from cross-site scripting attacks. Other researchers proposed various solutions to defend against script injection attacks, ranging from privilege separation [9], filtering [10, 14], to security policy enforcement mechanism [13]. However, these solutions and study were performed for desktop browsers. There is a lack of mechanisms for mobile browsers.

4. OBSERVATIONS

This paper assesses 30 mobile browsers against the W3C prescribed practices for security indicators. For each of the guidelines depicted above in paper, we make and run an arrangement of tests to check consistence on all the competitor browsers and record our observations. All the investigations were performed on web browsers on genuine cell phones (mobile phones), and are reproduced in the particular emulators to create a large portion of the figures all through the paper. The browser adaptations utilized as a part of our assessment are the most recent as of April 22, 2016.

Most of the mobile browsers do not show security warnings only Aesir browser, Apus browser, Best browser, Browser, Cool browser, Dolphin Zero, Javelin, Maxthone, Mini browser, Opera mini, Puffin, Web browser all these browser do not meet the guidelines provided by W3C consortium, these browsers simply neglect security warnings and do not show any indicator or anything regarding security. Some browsers give user false indication (i.e. false sense of security). Even if webpage does not have valid certificate they show it valid. Browsers like Boat browser, CM browser, Dolphin, DU browser, Next browser, UC browser all these browsers show same padlock icon for both secure as well as insecure webpages.

Also browsers like Google Chrome and FireFox shows security warnings but these browsers also implement only subset of W3C guidelines and do not show sufficient information about malicious webpage. Moreover in chrome we can access such sites but Mozilla prohibits user to access site. Due to insufficient information a sophisticated user can also go wrong way and might click through warning. Mozilla does not show certificate and its content, it only shows whether webpage is secure or not and gives name of security authority who validated certificate of that particular site.

It shows up entirely clear that the absence of accessible screen land has drastically impacted the utilization of security indicators on cell phones. While traditional desktop browsers have an extensively wealthier space spending plan to suit current (and even exploratory) indicators, mobile browsers battle to obviously indicate content, not to mention signs of the cause and security of the association with that substance. Present study recommends that browser vendors have separately settled on various choices to best adjust these contending requests, autonomously selecting to actualize diverse subsets of the indicators being used on desktop browsers. Tragically, our concentrate likewise uncovers that the decisions made by every browser evacuates signals accessible to clients to identify, and conceivably stay away from, particular assaults regardless of the fact that lone by master clients.

Table.1. Results of experiments on candidate mobile browsers to test compliance with the first two W3C guidelines given above in paper

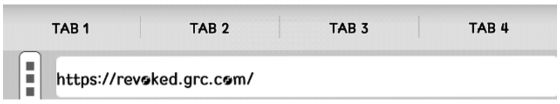
Name of mobile browsers	Identity signal availability		Certificates: required content
	Owner information available?	Certificate issuer's information available?	
Aesir browser	NO	NO	NO
Apus browser	NO	NO	NO
Best browser	NO	NO	NO
Boat browser	NO	NO	NO
Browser	NO	NO	NO
Browser	NO	NO	NO
CM browser	NO	NO	NO
Cool browser	NO	NO	NO
Dolphin	NO	NO	NO
Dolphin Zero	NO	NO	NO
DU browser	NO	NO	NO
FireFox	YES	YES	YES
FlashFox	YES	YES	YES
Javeline	NO	NO	NO
Maxthon	NO	NO	NO
Mini browser	NO	NO	NO
Next browser	NO	NO	NO
Opera Mini	YES	YES	YES
Opera	NO	NO	NO
Puffin	NO	NO	NO
UC browser	NO	NO	NO
Web browser	NO	NO	NO
Yolo	NO	NO	YES

Table.2. Results of experiments on candidate mobile browsers to test warning effectiveness

Name of mobile browsers	Malware warnings	Phishing warnings	Certificate error
Aesir browser	NO	NO	NO
Apus browser	NO	NO	NO
Best browser	NO	NO	NO
Boat browser	NO	NO	NO
Browser	NO	NO	NO
Browser	NO	NO	NO
CM browser	NO	NO	NO
Cool browser	NO	NO	NO
Dolphin	NO	NO	NO
Dolphin Zero	NO	NO	NO
DU browser	NO	NO	NO
FireFox	YES	YES	YES
FlashFox	YES	YES	YES
Javeline	NO	NO	NO
Maxthon	NO	NO	NO
Mini browser	NO	NO	NO
Next browser	NO	NO	NO
Opera Mini	YES	YES	YES
Opera	NO	NO	NO
Puffin	NO	NO	NO
UC browser	NO	NO	NO
Web browser	NO	NO	NO
Yolo	YES	YES	YES

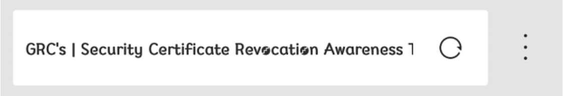
5. RESULTS

Under this research twenty three different mobile browsers has been tested against security warnings, some of them indicated the warning, some of them simply neglected it, some of them presented warning in partial way, and some of them presented warning in approximately accurate way. The percentage of mobile browsers that notify users about security warnings in comprehensive way is very less. During research we found only four such browsers namely FireFox, Chrome, Yolo and FlashFox. Remaining all browsers are unsafe to use and do not show any security warning in case of malicious webpage. Some of them use indicators to notify a user about security threat but that is also not enough to provide sufficient information to a user to allow him or her to make safe decision. Some browsers simply give false sense of security, browsers like boat browser, CM browser, dolphin, DU browser, next browser show false information.



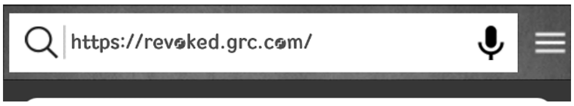
https://revoked.grc.com/

Fig.1. Aesir browser



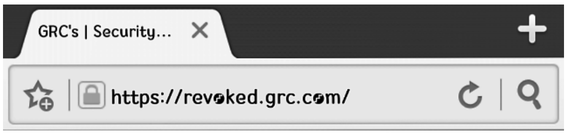
GRC's | Security Certificate Revocation Awareness T

Fig.2. Apus browser



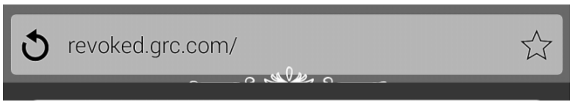
https://revoked.grc.com/

Fig.3. Best browser



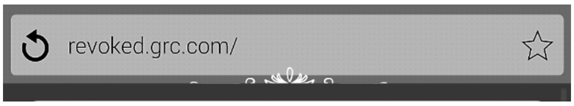
https://revoked.grc.com/

Fig.4. Boat browser



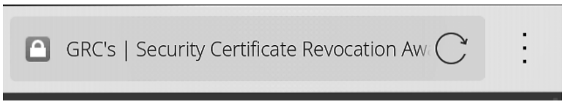
revoked.grc.com/

Fig.5. Browser 1



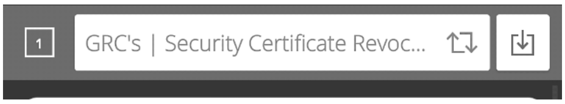
revoked.grc.com/

Fig.6. Browser 2



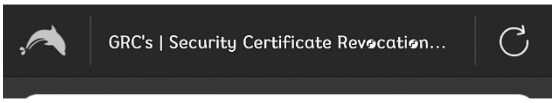
GRC's | Security Certificate Revocation Aw

Fig.7. CM browser



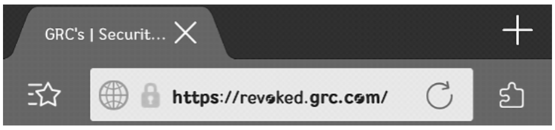
GRC's | Security Certificate Revoc...

Fig.8. Cool browser



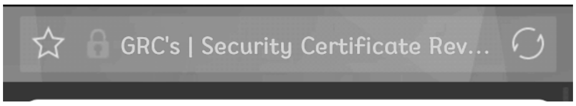
GRC's | Security Certificate Revocation...

Fig.9. Dolphin zero



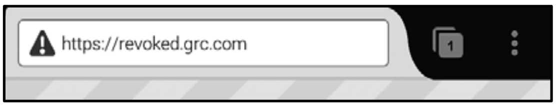
https://revoked.grc.com/

Fig.10. Dolphin



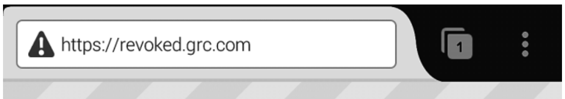
GRC's | Security Certificate Rev...

Fig.11. DU browser



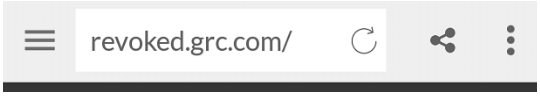
https://revoked.grc.com

Fig.12. FireFox



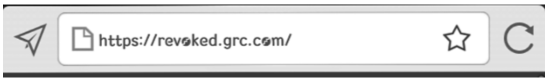
https://revoked.grc.com

Fig.13. Flashfox



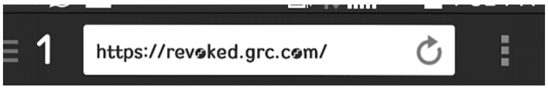
revoked.grc.com/

Fig.14. Javeline



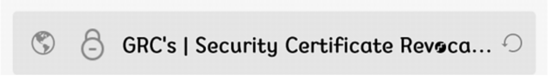
https://revoked.grc.com/

Fig.15. Maxthon



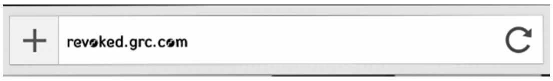
https://revoked.grc.com/

Fig.16. Mini browser



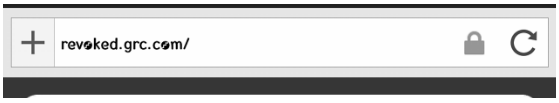
GRC's | Security Certificate Revoca...

Fig.17. Next browser



revoked.grc.com

Fig.18. Opera Mini



revoked.grc.com/

Fig.19. Opera

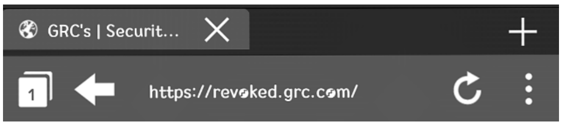


Fig.20. Puffin

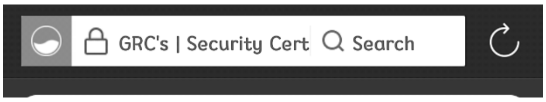


Fig.21. UC browser

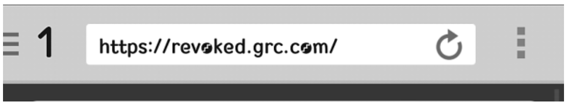


Fig.22. Web browser

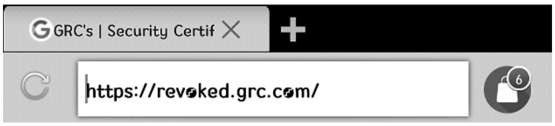
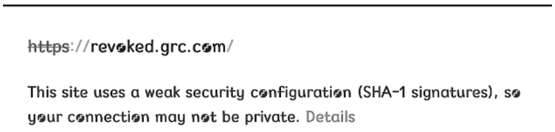


Fig.23. Web browser



~~https~~://revoked.grc.com/

This site uses a weak security configuration (SHA-1 signatures), so your connection may not be private. Details

Fig.24. Yolo

6. RECOMMENDATIONS

There are lot of constraints in mobile browsers like small screen size, limited space, limited resources and many more. Keeping all these things in mind we should design warning layout and content. We cannot copy everything from desktop browsers due to all these constraints. But we should at least implement security measures that will keep a user away from security threats. Following are some recommendations; this research paper gives, in direction to achieve more safer and reliable mobile browser.

- ***Describe the warning in more simple and complete way:*** Warning should be described in simpler and more understandable language, so that average user can also read it and can interpret it. Moreover it should be described comprehensively and should present important content of which can make user aware of threat. Enough information should be provided to the end user to allow him or her to make a proper and safe decision.
- ***Describe the consequences:*** Give few important consequences if a user ignores security warning. And also describe a way to avoid it.
- ***Be concise and accurate****:* Warnings should be concise and accurate; it should not be too long or ambiguous, if warning is ambiguous and too long, user simply ignores it and discard it and so the purpose is ignored.
- ***Avoid using difficult technical terms:*** Use of technical terms should be minimized and simple terms should be used to describe warning. Warning should always be designed from a user's viewpoint and not from vendor's viewpoint.
- ***Use of symbols:*** Warning should contain symbols resembling security threats. Use of symbols should be encouraged in order to make warning more precise and readable even by novice user. Symbols make warning easily readable and attractive too.
- ***Use of indicators:*** Proper set of security indicators should be implemented. Today different vendors implement different set of security indicators and most of them implement these security measures as per their convenience. So vendors should implement at least a subset of these indicators which can potentially increase the security of a user. In case of mobile browsers we cannot implement all the indicators which we do use in desktop browser but we can at least implement subset of these indicators to ensure security to an end user.
- ***Use of mobile hardware:*** Using the rest of the device's hardware for additional details could make security warnings more effective. Mobile browsers have small screens and all the information cannot be presented to user due to small screen so in such cases we can convey more information to a user if we use device hardware elements like LED, Sound alerts, and vibration. To build such systems, we suggest incrementally modify a running web browser to include capability to alter users with new indicators, hardware and dialogs.

7. CONCLUSION

Having performed exhaustive estimations of security pointers in the most generally utilized portable programs (more than 90% of the piece of the pie); we now examine what we see to be the suggestions. We have isolated this from our estimations to permit free translations by others. It shows up entirely clear that the absence of accessible screen land has drastically affected the utilization of security pointers on cell phones. While conventional programs have an extensively wealthier space spending plan to oblige current (and even exploratory) markers, versatile programs battle to obviously demonstrate content, not to mention signs of the source and security of the association with that substance. Our estimations propose that program sellers have separately settled on various choices to best adjust these contending requests, freely selecting to actualize diverse subsets of the pointers being used on desktop programs. Lamentably, our concentrate additionally uncovers that the decisions made by every program evacuates signals accessible to clients to recognize, and perhaps stay away from, particular assaults-regardless of the possibility that exclusive by master clients. It is not our decision that executing the very same security pointers on portable stages is outlandish from a designing viewpoint; but instead, the land furthest reaches of cell telephones makes vague the method for doing as such in a way that does not just overpower the substance on little versatile screens. Adding the https pointer to the location bar on the essential interface would make by far most of the URLs even less intelligible.

Modern browsers should follow guidelines of W3C consortium. Also developers should put sufficient security indicators to notify user about security breach. Analysis of different web browsers has shown that some browsers show security warnings and some does not, some browsers show security warnings but it is just for the sake of showing something and these warnings simply prompt insufficient information about security error, some browsers indicates false security information. In short every browser implements subset of security indicators according to their convenience; they simply neglect security of user. Modern mobile browsers enable a range of sensitive operations over SSL/TLS connections. Although mobile web browsers aim to implement equivalent functionality in traditional desktop web browsers, their smaller screen size has resulted in significant changes to the presentation and availability of SSL indicators. This research work first presents the large scale, cross sectional measurement of this class of applications and compares the security indicators used in the overwhelming majority of mobile browsers to traditional desktop counterparts.

Modern mobile web browsers are partially implementing recommended indicators from the desktop web browsers. Hence, it eliminates the opportunity for security savvy users to avoid attacks such indicators might signal. Our large-scale evaluation results lead us to the conclusion that current security indicators force our community to either accept a false sense of security or to argue for the complete implementation of indicators.

ACKNOWLEDGMENTS

This work is in part supported by an AWS in Education Research Grant award, USA.

REFERENCES

[1] Devdatta Akhawe and Adrienne Porter Felt, "Alice in Warningland: A Large Scale Field Study of Browser Security Warning Effectiveness", *Proceedings of 22nd Usenix Security Symposium*, pp. 257-272, 2013.

[2] Devdatta Akhawe, Bernhard Amann, Matthias Vallentin and Robin Sommer, "Here's My Cert, So Trust Me, Maybe? Understanding TLS Errors on the Web", *Proceedings of International World Wide Web Conference*, pp. 2-11, 2013.

[3] Yaoqi Jia, Yue Chen, Xinshu Dong, Prateek Saxena, Jian Mao and Zhenkai Liang, "Man-in-the-Browser-Cache: Persisting HTTPS Attacks via Browser Cache Poisoning", *Computers and Security*, Vol. 55, pp. 62-80, 2015.

[4] Chaitrali Amrutkar and Patrick Traynor and Paul C. Van Oorschot, "An Empirical Evaluation of Security Indicators in Mobile Web Browsers", *IEEE Transactions on Mobile Computing*, Vol. 14, No. 5, pp. 889-903, 2013.

[5] Lujo Bauer, Cristian Bravo-Lillo, Lorrie Cranor and Elli Fragkaki, "Warning Design Guidelines", Technical Report, Carnegie Mellon University, pp. 1-28, 2013.

[6] Max-Emanuel Maurer, "Bringing Effective Security Warnings to Mobile Browsing", *Proceedings of 2nd International Workshop on Security and Privacy in Spontaneous Interaction and Mobile Phone User*, pp. 1-2, 2013.

[7] W3C: Web Security Context: User Interface Guidelines, Available at: http: //www.w3.org/TR/wsc-ui/, August 2010.

[8] Apple Human Interface Guidelines, Available at: http://developer.apple.com/Mac/library/documentation/ UserExperience/Conceptual/AppleHIGuidelines. Accessed on 2013.

[9] Kailas Patil, Xinshu Dong, Xiaolei Li, Zhenkai Liang and Xuxian Jiang, "Towards Fine-Grained Access Control in JavaScript Contexts", *Proceedings of 31st International Conference on Distributed Computing Systems*, pp. 720-729, 2011.

[10] X. Dong, K. Patil, J. Mao, and Z. Liang. "A Comprehensive Client-Side Behavior Model for Diagnosing Attacks in Ajax Applications", *Proceedings of 18th International Conference on Engineering of Complex Computer Systems*, pp. 177-187, 2013.

[11] Kailas Patil and Braun Frederik, "A Measurement Study of the Content Security Policy on Real-World Applications", *International Journal of Network Security*, Vol. 18, No. 2, pp. 383-392, 2016.

[12] Kailas Patil, T. Vyas, F. Braun, M. Goodwin, and Z. Liang, "Poster: UserCSP-User Specified Content Security Policies", *Proceedings of Symposium on Usable Privacy and Security*, pp. 1-2, 2013.

[13] Samay S Omanwar, Kailas Patil and Narendra P. Pathak, "Flexible and Fine-Grained Optimal Network Bandwidth Utilization Using Client Side Policy", *International Journal of Scientific and Engineering Research*, Vol. 06, No. 7, pp. 692-698, 2015.

[14] Dnyaneshwar K Patil and Kailas Patil, "Automated Client-side Sanitizer for Code Injection Attacks", *International Journal of Information Technology and Computer Science*, Vol. 8, No. 4, pp. 86-95, 2016.