Security Analysis of Captive Portals

Omkar Parkhe, Sri Sai Bhargav Tiruveedhula, Nida Syed

July 3, 2019

Due to the proliferation of public WiFi networks such as university, coffee shops, etc., authentication in these networks is performed using captive portals. These portals have largely evolved in terms of security, usability and interoperability. They are widely used in large-scale campus-wide networks using firewall and multiple authentication servers to prevent theft of service. A malicious user can perform war driving and war chalking to locate open WiFi and hotspots [1]. There are many security issues in large-scale captive portals as it operates in the physical and data link layers.

Many portals only encrypt usernames and passwords during the authentication phase, and transmit all user data in the clear. Some portals do not even encrypt usernames and passwords. Many hotspots use such portals to obtain credentials and payment and then leave it upto the users to protect their own data, sometimes without even informing them. Moreover, most organizations do not block all the outbound ports like TCP/3128 for a web proxy, TCP/22 for SSH which can be used for bypassing captive portals [2].

A captive portal can also be used for stealing the browser history using Cross-site request forgery (CSRF) using the HTTP Strict Transport Security (HSTS) headers and long-term cookies [3]. This is done by invisibly placing vast amounts of specially crafted references into these portal pages, we can lure the browser into revealing a users browsing history by either reading stored persistent (long-term) cookies or evaluating responses for previously set HSTS headers.

Other attacks to circumvent the security of captive portals include DNS tunneling, Man in the Middle, session hijacking and freeloading attacks. In DNS tunneling attacker creates a tunnel through the payload of DNS packets using available tools to gain access to the network without authentication. It is observed that Iodine and Tuns are efficient tools as compared to the other state-of-the-art tools to perform DNS tunneling in networks using captive portals [4].

A malicious user can perform a MitM attack using a Rogue AP having more signal strength than the original AP [5]. The attacker can force an authenticated user to break the connection with original AP and connect with a Rouge AP and obtain the login credentials using tools to crack the internal authentication protocol. This attack can be countered by using hash of hash of password and a challenge generated by a cryptographically secure PRG to authenticate a user.

Session hijacking includes stealing of authenticated users credentials and other configurations by IP spoofing, ARP spoofing and by taking advantage of incomplete firewall rules to gain access to the network[6]. This attack can be countered by the use of session id and periodical authentication of users in order to detect the session hijackers. Similarly, freeloading leverages the presence of personal firewalls to gain free access to the network. Incrementing MAC sequence number each time a datagram is send will help the Access Point (AP) to detect freeloading attack by noticing the trend line of MAC sequence numbers [7].

Counter measures of [7] can be used to develop secure opportunistic hotspots for commercial as well as non-commercial purposes. These hotspots make use of both captive portals and 802.1x/802.11i IEEE standards for visitors and organization members to provide access to internet and intranet, respectively

[8]. The AP along with firewalls will make sure that the visitors are not given access to intranet. The captive portals can make use of online payment servers (OPS) such as PayPal and Boingo to perform billing operation.

Captive portals are also used along with other authentication mechanism such as WEP, WPA and WPA2. Many small public networks use WEP with captive portals which can be easily cracked using WEP cracking tools. As WPA and WPA2 are quite robust in environments, such as home, small business, and enterprise settings, where the base station and client securely share a key. However, they cannot provide the same properties in open hotspot environments due to the lack of a properly shared secret key.

Another solution for securing the public networks is using VPNs. However, this is not feasible as many servers do not support VPNs [9]. Implementation of RADIUS server for authentication and pfsense firewall on the captive portal server in small size public networks [10] can improve the security of the public network. Also, IEEE has developed advanced authentication and encryption protocol called 802.1X to solve the vulnerabilities found in WPA2. However, the 802.1X standard needs devices that work with the protocol, making it complicated than Captive Portal. Therefore, 802.1X is not widely deployed in WLAN.

Securing large size public networks can be done using key distribution using hierarchical identity based cryptography can be used. Key establishment between an unproven user and an access point using identity-based cryptography [9]. This eliminates MAC spoofing and detects rogue access points. Installation of antivirus and firewall on the server of captive portal webpage further enhances the security of users. Denial of service attacks on captive portals can also be reduced by bandwidth throttling [3]. Authenticating users with the use of stateless HTTP redirector will help in prevention of SYN-flooding attacks [11].

In this work, we utilize a multitude of well-established attacks over captive portals which substantiate concerns about public Wi-Fi networks. We demonstrate the damage that an public Wi-Fi with captive portal can inflict upon the users.

References

[1] K. J. Hole, E. Dyrnes, and P. Thorsheim. Securing wi-fi networks. *Computer*, 38(7):28–34, July 2005.

[2] Marc Laliberte. Lessons from DEFCON 2016 Bypassing Captive Portals. Report, Secplicity, 2016.

[3] A. Dabrowski, G. Merzdovnik, N. Kommenda, and E. Weippl. Browser history stealing with captive wi-fi portals. In *2016 IEEE Security and Privacy Workshops (SPW)*, pages 234–240, May 2016.

[4] Alessio Merlo, Gianluca Papaleo, Stefano Veneziano, and Maurizio Aiello. *A Comparative Performance Evaluation of DNS Tunneling Tools*, pages 84–91. Springer Berlin Heidelberg, Berlin, Heidelberg, 2011.

[5] M. Cai, Z. Wu, and J. Zhang. Research and prevention of rogue ap based mitm in wireless network. In *2014 Ninth International Conference on P2P, Parallel, Grid, Cloud and Internet Computing*, pages 538–542, Nov 2014.

[6] Pierre Anderberg and Erik Thorselius. How to circumvent a captive portal.

[7] Haidong Xia and José Brustoloni. *Detecting and Blocking Unauthorized Access in Wi-Fi Networks*, pages 795–806. Springer Berlin Heidelberg, Berlin, Heidelberg, 2004.

[8] Haidong Xia and J.C. Brustoloni. Secure and flexible support for visitors in enterprise wi-fi networks, 01 2005.

[9] J. Choi, S. Y. Chang, D. Ko, and Y. C. Hu. Secure mac-layer protocol for captive portals in wireless hotspots. In *2011 IEEE International Conference on Communications (ICC)*, pages 1–5, June 2011.

[10] Felix Larbi Aryeh, M Asante, and AEY Danso. Securing wireless network using pfsense captive portal with radius authentication–a case

study at umat. *Ghana Journal of Technology*, 1(1):40–45, 2016.

[11] K. Koht-Arsa, A. Phonphoem, and S. Sanguanpong. Architectural design for large-scale campus-wide captive portal. In *43rd Annual 2009 International Carnahan Conference on Security Technology*, pages 72–76, Oct 2009.