Nick Alexander

January 28, 2015

Packets in Packets Analysis

The main purpose of the article *Packets in Packets* was to describe a vulnerability that exists in the way that radio devices interpret data. An exploiter of this vulnerability would be capable of passing malicious packets through a firewall as well as other intrusion detection systems. By creating a valid packet and storing the malicious packet *inside* the payload of the clean packet, the exploiter can trick the radio device into unpacking the malicious packet and sending it up the network stack. Furthermore, the exploiter does not need any type of heightened privileges, kernel compromises, or even physical access to the radio device. I am not convinced that a PIP attack is reliable attack mechanism, but I do believe the concept raises valid concerns with the current OSI model of networking.

Any time that a vulnerability is found, one must consider the access requirements. For example, if an attacker needs root-level access as well as access to a physical machine, it is much less viable than an exploit that only requires an IP address. In the case of packets in packets, the attacker does not need physical access to the radio network (e.g. does not need to be connected to target wireless router) nor does the attacker need any heightened privileges. Instead, maliciously stuffed PIP must only be hidden (or sent to a user) where the packets will travel along a radio network, such as WiFi. Due to the nature of the physical frames, unexpected network devices can read the malicious packet can be processed by an unsuspecting machine.

For example, suppose that a technical user is in an office building protected by WPA2 encryption which protects against the PIP vulnerability. However, suppose there is another user on an unprotected network, such as a home network or a public hotspot. If the unprotected user downloads a file embedded with PIPs, the protected user's wireless card may still read the malicious packet. This vulnerability is due to the inherent trust in radio frames; a radio receiver will ignore radio frames that are not addressed to itself but read frames that are addressed as broadcast. As an analogy, consider receiving a piece of mail in your post office box. The mail is not addressed to you, so you simply hand it back to your post master - this is what radio receivers will do normally. However, if the letter is vaguely addressed, "to whom it may concern" for example, you are going to open the letter - the radio receiver accepts broadcast frames. Inside the letter is some harmful pathogen, and you have just fallen for a packet in packet injection. Again, radio devices work on this implicit trust. [1]

While I hold the belief that this type of attack is fairly implausible, I believe the principal is enough for a call to arms. Consider an urban setting with many corporate entities nearby. Businesses like McDonalds and Starbucks - who pride themselves on their public hotspots - are

going to flock to these high density, high traffic areas. Therefore, as little Timmy downloads a file with PIPs at McDonalds, it is possible to penetrate a corporate environment. This idea becomes even more alarming when one considers that there are expected to be 5.8 million public wireless hotspots by the end of 2015 [2]. This fear becomes almost crippling as the prospect of a global wireless network emerges from the rising new company SpaceX [3]. If we have a massive wireless infrastructure, the packets in packets vulnerability could become much more prevalent.

The solution, however, is not exceedingly simple. It seems to be that the only way to functionally eliminate the possibility for a PIP injection is the completely reengineer the way that radio receivers handle broadcast packets - or how the entire OSI model handles broadcast packets. As a whole, broadcast packets are an extremely useful mechanism in causing chaos in a network. Here we have PIP injection relying on broadcast and another example of attacks relying on broadcast packets is a smurf attack. The Global Information Assurance Certification generally hold the stance that unnecessary broadcast packets should be suppressed [4]. As many know, early networking was designed for simplicity and proper functioning but not for security since networks were implicitly trusted.

Today, we are working towards a new Internet as a whole. New protocols like HTTP/2 [5] and IPv6 by the IETF as well as the Web 2.0 movement [6] by the World Wide Web Consortium are sparking a whole new Internet never before imagined. We are cultivating an Internet of Things with Smart Homes and Smart Cars [7]. With millions of devices connected to the Internet running on an outdated 7 layer model, should consumers simply sit back and accept the fact that their devices are vulnerable? Of course it is impossible for a networked machine to be completely safe, but I believe a number of vulnerabilities could be eradicated if the foundation of the Internet was refined. Similar to construction, a building is only as strong as its foundation.

References

1. Goodspeed, T. (2011, December 28). 802.11 Packets in Packets [28C3]. 802.11 Packets in Packets [28C3]. Retrieved from https://www.youtube.com/watch?v=I_XGUgpYGww
2. Wifi hotspots set to more than triple by 2015. (n.d.). Retrieved 30 January 2015, from http://www.informa.com/Media-centre/Press-releases--news/Latest-News/Wifi-hotspots-set-to-more-than-triple-by-2015/
3. Vance, A. (2015, January 16). Elon Musk and SpaceX Plan a Space Internet. *Bloomberg Business*. Bloomberg. Retrieved from http://www.bloomberg.com/news/articles/2015-01-17/elon-musk-and-spacex-plan-a-space-internet
4. Scheible, A. (n.d.). Global Information Assurance Certification. Retrieved 30 January 2015, from http://www.giac.org/paper/gsec/1838/network-security-wake-on-lan-technology/103243
5. IEFT. (n.d.). HTTP/2. Retrieved 30 January 2015, from https://http2.github.io
6. Beal, V. (n.d.). Web 2.0. Retrieved 30 January 2015, from http://www.webopedia.com/TERM/W/Web_2_point_0.html
7. Greenough, J. (2015, January 26). THE INTERNET OF EVERYTHING: 2015. *Business Insider*.