“Design of the Kali Linux Operating System Performance Evaluation and Analysis.”

By Eric Webb

**Abstract:**

Every so often there is a new flavor of Linux developed that sets the standard of what an Operating System can be. This literature discusses one of those flavors, Kali Linux. It also begins to cover specific technical analyses of default programs embedded within the Kali operating system and evaluates their performances. This information was gathered through peer reviewed research provided from the IEEE and ACM journal digital libraries. Academic studies were applied to previously written content and projects. From this research an educated analysis and evaluation of the programs and their subject matters were concluded. Following that with an opinionated discussion about the future of this open source project and its embedded programs.

**Introduction:**

Kali, a distribution of Linux designed for penetration testing and digital forensics, is a powerful tool across many technical spectrums. This security-focused operating system is managed through Offensive Security LTD and is essentially an upgrade from its predecessor called Backtrack (Gaddam & Nandhini, 2017. Pg. 3). Mastery in many of these technical facets can be compared to a very experienced driver behind a very fun, but dangerous well-tuned car. Meaning, Kali Linux can be great for learning purposes and offensive security practices within ethical hacking, but can ultimately be used for malicious intent with the wrong user.

This flavor of Linux is popular within the ethical hacking community because it has the ability to boot off of an external device such as a USB device, CD-ROM, or a Virtual Machine. This is beneficial because security professionals can now carry these tools within Kali’s operating system anywhere with them. This ability to boot up Kali and conduct research from a multitude of devices, instead of a single dedicated device, makes testing more portable and available in an emergency.

The tools already embedded within Kali’s Debian based operating system contain many techniques and practices for digital forensics and network security. Giving Kali so many options across these spectrums gives it the flexibility to meet the demands of the penetration testing market. This literature will now discuss these features of Kali Linux, where they currently fit in society, why they are important, and potential future endeavors of the foresaid technologies.

**Use Case 1: Packet Crafting with Hping3.**

The Kali operating system comes with the ability to packet craft right out of the box. One means of achieving this is using the Hping3 tool that is already pre-installed. Hping3 allows for the creation of a multitude of packets to be sent out in different configurations. This is important to the network security and ethical hacking realm because it allows administrators to test the network for entry points that can be exploited.

A real world case example is demonstrated in a network lab where a virtual machine running Kali Linux can create multiple TCP packets via a Spoofed IP address using Hping3. Multiple packets try to create multiple connections to a TCP port. Doing this TCP flood fast enough can exhaust resources if the network is not properly configured and can cause a denial-of-service attack to occur. This DOS attack is accomplished by Hping3 packet crafting and is a powerful tool that can be taken advantage of in the Kali Linux operating system. It is important because it allows administrators to offensively probe the project’s security scope, but can also be used nefariously to bring down systems (Cui, Liu, Zheng, & Huang, 2018. Pg. 3).

As packet crafting software becomes more popular administrators will have to design networks to protect against high amounts of traffic. The installation and use of load balancers and advanced firewall filters will become ever more present in network design so that DOS type attacks such as a TCP flood doesn’t affect the network.

**Use Case 2: Intrusion Detection with Snort.**

Not only does Kali offer strictly offensive security tools embedded within it’s out of the box operating system, but it also offers techniques for network security as well. This is apparent in a tool called Snort. Snort is an intrusion detection system that allows the administrator to set up rule tables to allow alerts of nefarious activities. These alerts can also trigger events down the line to execute procedures to mitigate these attacks. Alerts can also be set as logs for auditing purposes.

This can be demonstrated by using Snort to set up rules to alert administrators of possible harmful traffic within a network. For example, someone can set up Snort on a server to allow administrators to get notifications of events on the network such as OS fingerprinting attempts, buffer overflows, stealthy port scans, and more (Gaddam & Nandhini, 2017. Pg. 3). One can use the Intrusion detection in many ways. One example is creating Snort based rules based off of a honeypot result set. Another is to create rules based off of efficient port scanning detection rules by getting the results of port scans. Snort is not just limited to network and host based intrusion detection, but can also be applied in cloud environments (Gaddam & Nandhini, 2017. Pg. 4).

This is important because a network would typically have a statefull Snort IDS behind the stateless firewall. The stateless firewall handles and filters a lot of traffic independent of what is inside the packets. Once the initial filtering is complete the importance of the Snort IDS is applied. Snort will then take those accepted packets and perform a statefull analysis fully dissecting the contents of the packet. A lot of times using signature or anomaly based detection, Snort can detect a nefariously crafted packet that made it past the firewall. That is why it is important for the IT infrastructure to actively have traffic passing through an IDS to protect against malicious packets intruding the network. Therefore, Kali Linux supporting Snort right out of the box is a huge benefit from a security standpoint.

As packet crafting becomes more prevalent and new ways are created to breach networks, IDS’s will have to grow. This will most likely mean larger and more cumbersome libraries of known signatures. Since Snort can use a signature based libraries to detect known intrusion it will no doubt increase in size as time goes on, making scans more resourceful and time exhausting. Anomaly based detection is predicted to become more efficient and prevalent as the algorithms become more fined tuned for detecting fraud.

**Use Case 3: Network Attacks with Ettercap.**

Another powerful tool in the Kali Linux OS distributions’ arsenal is a multipurpose tool, called Ettercap. This tool has many capabilities including, but not limited to packet sniffing, packet filtering, DNS spoofing, forging SSL certificates, and even offers a suite for attacking the Point-to-Point Tunneling Protocol. Because this tools has many capabilities, it is known to discover exploits in networking technologies, which leads to the advancement of best practices in security. The following literature will look at a use case examples using Ettercap.

One documented example of Ettercap with another Kali Linux tool, called SSL-strip, were used to exploit an implementation of PPTP. Essentially SSL-strip forces requests to HTTP pages that would normally be redirected to HTTPS pages, to stay pointed at their non-secure HTTP counterparts. This means that the login information would no longer be sent over the secure PPTP tunnel, but now be sent via the insecure methods of HTTP. After the SSL-strip removed the secure redirect, then Ettercap was able to listen in and monitor network traffic in clear text revealing the credentials (Pingle, Mairaj, & Javaid, 2018. Pg. 2). This is important because it revealed to the world a vulnerability that you did not have to overcome the PPTP tunnel, but simply had to stop the user from implementing it by stopping its redirects. It is also a great example of how one of the Kali Linux operating system’s pre-downloaded features is used to help benefit the advancement of network security. Because of the flaw of not checking for redirects, HTTPS standards were introduced to further check if logins are sent to their HTTPS secure location redirects or not. (Pingle, Mairaj, & Javaid, 2018. Pg. 2).

As the future of MITM attacks unfolds, more and more protocols and standards will be developed to mitigate them. Every time an exploit is discovered a new procedure will be instilled to protect users against it. This will lead to more cumbersome libraries, procedures, and protocols as time goes by but will be needed for the sake of security.

**Use Case 4: WarDriving with Airmon, BlueNMEA and Kismet.**

As the development of wireless local area networks becomes all more increasing, the number of access points left open for vulnerabilities and exploits becomes more prevalent, thus leaving attackers with more options to choose from. With some of the tools available in the Kali operating system it is possible to perform war driving reconnaissance and enumeration in a geographical area. An example of war driving performed in the Kali Linux OS is when exploiters were able to use Airmon, BlueNMEA, and Kismet to drive around and gather WLAN information from local access points. This was done by initially booting up Kali Linux and enabling monitor mode to sniff wireless packets using airmon-ng. Blootooth and GPS sensors were paired to the instance of Kali. The GPS device was sending GPS information to the Bluetooth device and was being relayed back to the Kali instance vie BlueNMEA an application for sending GPS information. This information was being recorded and read via Kismet to extract this data. From doing this WarDriving exercise the ones administering this were able to find a multitude of WLAN networks and clients. While most of the clients were protected by the secure WPA2, some environments did not have strong encryption implementations and could be taken advantage of. (Akram, Saeed, & Daud, 2018 Pg. 3)

This WarDriving performance shows how administrators can set up a scenario using Kali Linux where they were able to drive around the network perimeter, probing the network for weak spots. It also shows how a criminal could theoretically drive around and probe areas for easy entry points with little to no security, essentially WarDriving for low bearing fruit. Another example is how it can be applied in a militaristic or police stand point, where authorities could probe the area and get an idea of the number of devices belonging to a target to get an idea of how many communication channels are being used within that targeted network.

In the future, WarDriving practices will become more prevalent and will have its complexity deteriorate as pre-written graphical user interfaces become easier for the novice user to understand. Systems will have to develop self-identifying technology that will have to identify when they are being probed. As Kali Linux evolves there will be great strides in not just improving war driving technologies but also preventing it.

**Use Case 5: Structured Query Language Attack and Defense.**

After all the advancements of good coding practices and knowledge available about network security, mistakes are still made and negligence is still applied. Because of these, concepts of attacks like SQL injection are still a discussion. SQL injection allows for information to be obtained or SQL commands to be executed that should not be, creating a huge security threat that must be dealt with accordingly. In the following literature, there are examples Kali Linux providing offensive security and automated protection against SQL based attacks.

One example of a SQL injection attack happens when there is no sanitization between the user input and the database. The information needs to be validated and the user input verified as well. One infamous example using SQL injection can be seen where no sanitation occurs and a user can enter 1=1 in an input field. In SQL 1=1 equals true. So regardless of the question if 1=1, then the answer is true. This is used in the login where 1=1 is used in the ID field and returns true to being an admin (Gudipati, Venna, Subburaj, & Abuzaghleh, 2016 Pg. 3). This is of course an obscure over simplification but still shows the power of SQL injection.

Kali builds from this SQL injection prowess and incorporates automated SQL injections tools. One tool in particular is called SQLmap and comes preinstalled on the Kali Linux OS. SQLmap can be thought of similar to Nmap, in that it can be used to enumerate databases and retrieve data such as names, servers, and operating systems. Very much like how Nmap enumerates network information, SQLmap queries database information. SQLmap is important because it helps automate the task of enumerating a database and gathering information making SQL injection that much quicker to apply since there is a better understanding the target. This can also help administrators probe databases for weak points (Gudipati, Venna, Subburaj, & Abuzaghleh, 2016 Pg. 6).

Protecting against sequel injection is important, as applications susceptible to SQL injection, can cause much agony for an organization. Having records and files with the vulnerability to be copied, manipulated, or deleted at will is a huge flaw in an IT organization. Therefore, actively probing the network for spots for injection is important to maintain a proper security posture.

In the future, as more applications are developed with user inputs communicating to the back end, Kali Linux and other platforms will incorporate more automated injection and scanning techniques. These techniques will make it easier for users to find and peruse applications that are susceptible to SQL injection. Of course as these advanced scanning and injecting techniques are developed and understood, more sophisticated procedures will be implemented to mitigate the threat of SQL injection. It is only a matter of self-discipline if these procedures are followed.

**Use Case 6: Attacking SIP**

Multimedia is a big part of people’s everyday lives with the technical aspects of multimedia being abstracted away from the average user’s point of view. An example of this is how VOIP services use the Session Initiation and Real Time protocols for voice traffic, which is not common knowledge to most people. The Kali Linux OS offers many tools from installation that are used to gather and exploit vulnerabilities dealing with VOIP infrastructures.

Tools in Kali Linux are used for attacking the SIP protocol. One specific instance is using a tool called SIPflood to dump SIP session content and RTP session contents to be later reviewed and analyzed. Another powerful tool for flooding is Sipack, which is used to create invite, register, and UDP type floods. These floods create DOS attacks on VOIP networks. Lastly, tools like John the Ripper, Authtool, and Svcrack are used for password cracking to recover encrypted passwords. (Rehman & Abbasi, 2014 Pg. 5)

Defending VOIP protocols is important because they are used for everyday phone calls over the internet. Since Kali Linux offers a suite of tools for attacking VOIP protocols, it is beneficial to a network administrator by actively seeing if calls can be monitored, tampered, or Denied. This is important to a non-technical user because it shows how this open source operating system can be used out of the box to help listen in on phone calls.

As future tools are developed to focus attacks on VOIP technologies, there will be more standards and procedures in place not only on the client side of the VOIP infrastructure, but more rules and guidelines used and administered by the ISP’s as well.

**Conclusion:**

From these articles one can see how the Kali Linux operating system is a very powerful tool. Not only does it have the capabilities right out of the box to actively probe networks for vulnerabilities, but also that the ability to exploit those vulnerabilities. This can obviously be used for offensive security or for more nefarious measures. These examples show how Kali performs various techniques such as packet crafting, providing intrusion detection, taking advantage of HTTP to HTTPS redirects, performs WarDriving, uses automated mapping tools for easier SQL injection, as well as tools to attack VOIP protocols such as SIP and RTP. This is important because this open source operating system is available for free to anyone on the web and can trigger a spark for growth of knowledge, but can also be used very dangerously in the wrong hands. As Kali Linux continues to grow it will meet the demands of security for that era. New tools will be added and deleted from its library, while fine tuning current tools with more capabilities and features.

**Work Cited.**

Akram, Z. A., Saeed, M. A., & Daud, M. A. (2018, April 6). *Wardriving and its Application in Combating Terrorism.* doi: 10.1109/CAIS.2018.8442035

Cui, Y., Liu, Q., Zheng, K., & Huang, X. (2018, December 27). *Evaluation of Several Denial of Service Attack Methods for IoT Systems.* doi: 10.1109/ITME.2018.00179

Gaddam, R. T., & Nandhini, M. (2017, July 17). *An analysis of various snort based techniques to detect and prevent intrusions in networks proposal with code refactoring snort tool in Kali Linux environment.* doi: 10.1109/ICICCT.2017.7975177

Gudipati, V. K., Venna, T., Subburaj, S., & Abuzaghleh, O. (2016, October 15). *Advanced automated SQL injection attacks and defensive mechanisms.* doi: 10.1109/CT-IETA.2016.7868248

Pingle, B., Mairaj, A., & Javaid, A. Y. (2018, October 22). *Real-World Man-in-the-Middle (MITM) Attack Implementation Using Open Source Tools for Instructional Use.* doi: 10.1109/EIT.2018.8500082

Scott, B., Xu, J., Zhang, J., Brown, A., Clark, E., Yuan, X,Williams, K. (2017, December 14). *An interactive visualization tool for teaching ARP spoofing attack.* doi: 10.1109/FIE.2017.8190531

Rehman, U. U., & Abbasi, A. G. (2014, December 9). *Security analysis of VoIP architecture for identifying SIP vulnerabilities.* doi: 10.1109/ICET.2014.7021022