ePortfolio Activity

What does the article teach you about carrying out vulnerability scans using Kali?

The article mainly points out the importance of understanding that when you work with Kali Linux, you're not working with just the tools it provides, but the operating system itself. Safeguards and behaviors that common Linux distributions have, do not exist on Kali, and to have a frictionless experience when scanning for software vulnerabilities, ensure that you understand how Kali Linux is different from other Linux distributions, and what how the tools you use modify or interact with the operating system.

What issues might you encounter?

- If you're not careful, you can end up in a situation where Kali sends noise throughout the networks you're scanning for vulnerabilities, tainting the results that you will obtain.
- According to the article, Kali only has a root user. If you attempt to run an application that scans for vulnerabilities (or complete any other task), there is a chance that you can break your system if the application in question modifies any critical operating system files and you either type something incorrectly or run a command that you do not fully understand the consequences of.
- If you need any additional software, you need to be careful with the installation thereof, because if it's not on Kali's official repository, obtaining it from another Linux distribution's repository can pose a threat to the operating system. Despite being Debian-based, Kali Linux itself is not officially compatible with packages that originate from other operating systems, such as Ubuntu. Packages could still be installed from these distributions, however, there's a high risk that system stability could be compromised.

How would you overcome them?

- Doing research before using any tools or starting any scans. This would mean reading the documentation of the tools that I'm using, and in doing so, make sure I understand how they interact with a network, how they could impact the network, and how they interact with the operating system itself.
- For the second point, I tested this behavior, and found that what the article says is no longer true- getting root access requires additional steps now. You can create your own user account, but to enable the root user, you need to use `sudo su` or open the clearly marked root terminal emulator from the desktop, which requires your current password (OffSec, 2022a). This makes it harder to break the system, however, it's still important to understand why certain tools or commands might need root access, because giving root access can still pose a threat to your system.
- I would opt to avoid adding any packages that do not come from the official Kali Linux repository. If I need any other programs or tools, I would get another Linux distribution. I use Windows Hyper-V to use Linux, and thus can easily create another virtual machine for distribution such as Ubuntu.

How do their results compare with your initial evaluation?

Their results deal with specific tools that come preinstalled with Kali Linux (OffSec, 2022b). The findings do help provide a direction for selecting specific tools available on Kali Linux- evaluation categories listed such as "Easy" and "Well-documented" help mitigate the risk of damage I outlined previously, where the damage in question could be caused by running a command which is not fully understood, or mistyping an argument to a command-line tool.

What do you think of their criteria?

Their given criteria are well-selected. For the executive summary assignment, time is of the essence and convenience is of paramount importance because everyone in my team is from a different background, with different levels of experience, and different computing situations. To be efficient at working on the assignment, the tools which we select must be user-friendly, free, cross-platform, and popular (so it's easier to have questions answered if anyone has any). These criteria are discussed by the authors, which I agree with completely. However, I don't feel that "Acclaimed" should be a relevant criteria- that is a subjective area and arguably does not provide a good rationale for selecting a tool. I'd also make the argument that "Support" should be split into two categories: "Supported by Developers" and "Supported by Community". The reason I say this is because user-generated content such as tutorials, guides, and FAQs are highly useful for those looking to learn how to use a new tool. However, it is not the same as developer support- developer support would include contributions such as bugfixes, patches, and the development of new features. However, I can't tell the difference if the category is defined only as "Support". For my team's purposes, user-generated content is more important because we need to develop a functional knowledge of the tools we're using, and would be less concerned about excellent developer support because of the timeframe of the assignment. Having this distinction would make it much easier to make a decision for our use case.

What are the pros and cons of using Kali Linux vs. Nessus?

Nessus provides an extremely streamlined approach to vulnerability scanning. Specific tools used for vulnerability scanning are abstracted away from the user, and instead, they can merely set a target to scan and obtain a list of vulnerabilities which are already scanned. It is also free for students.

Has this changed your original evaluation score?

It hasn't- it appears that there's a large emphasis on using multiple tools in conjunction with each other. Nessus does not act as a replacement for Kali Linux, neither does Metasploit. In fact, Kali Linux comes with Metasploit preinstalled and the creators of Nessus have a tutorial on how to install it on Kali Linux (Tenable, 2017). The more important thing to consider is whether necessary to use all of these tools or stick only with basic command line tools which Kali provides. I now think that it will be good to go into more detail in tool selection in the executive summary- my team will do this by discussing comprehensive suites (such as Metasploit and Nessus) versus individual command line tools, and how they link to our need of finding vulnerabilities outlined by the OWASP Top Ten. This can also be tied to research which has been done by Wimal, who suggested the use of Burp Suite and OWASP ZAP.

References

OffSec. (2022a) Enabling Root. Available from: https://www.kali.org/docs/general-use/enabling-root/ [Accessed 16 January 2022].
OffSec. (2022b) Kali Tools. Available from: https://www.kali.org/tools/ [Accessed 16 January 2022].

Tenable. (2017) Getting Started with Nessus on Kali Linux. Available from: https://www.tenable.com/blog/getting-started-with-nessus-on-kali-linux [Accessed 16 January 2022].