Bangladesh University of Engineering and Technology

Department of Computer Science and Engineering

Academic Year 2023–2024

CSE 406
Computer Security Sessional

Wazuh: A Comprehensive Look at its XDR and SIEM Capabilities for Enhanced Security

---

Submitted by:

1905001 — Mohammad Sadat Hossain

1905004 — Asif Azad

1905005 — Md. Ashrafur Rahman Khan

Supervisor: Abdur Rashid Tushar

Submission date: March 8, 2024

Contents

Wazuh: A Comprehensive Look at its XDR and SIEM Capabilities for Enhanced Security

1 Introduction

Securing IT infrastructure in today's complex environments requires robust and comprehensive security solutions. Wazuh, a free and open-source security platform, offers a powerful combination of Extended Detection and Response (XDR) and Security Information and Event Management (SIEM) capabilities. This empowers organizations and individuals to safeguard their data assets across various environments, including on-premises, virtualized, containerized, and cloud-based ones. Wazuh's popularity is a testament to its effectiveness, with thousands of organizations worldwide, from small businesses to large enterprises, relying on it for data security.

1.1 Brief Background on Wazuh

Wazuh is a free, open-source security platform that combines XDR and SIEM capabilities. This allows it to protect workloads across various environments, including on-premises, virtualized, containerized, and cloud-based ones. By using Wazuh, organizations and individuals can safeguard their data assets from security threats. Its popularity is evident, with thousands of organizations worldwide, from small businesses to large enterprises, relying on it for data security.

2 Overview of Ghidra

2.1 History and development

Ghidra was developed by the National Security Agency as an internal tool for analyzing malware, viruses, and other executable files. After years of internal use, the NSA decided to release Ghidra as an open-source project in 2019. This allowed the software reverse engineering community to collaborate in enhancing and expanding Ghidra's capabilities.

2.2 Core capabilities

Ghidra provides a comprehensive set of static and dynamic analysis features centered around disassembly and decompilation. Key capabilities include:

1. Disassembly - Extracting assembly code from executable binaries
2. Decompilation - Reconstructing high-level source code structures

3. Control flow graphs - Visually mapping program execution flows
4. Data type analysis - Inferring variable types and data structures
5. Patching binaries - Modifying compiled code and re-exporting it
6. Scripting API - Support for headless analysis using Python and Java
7. Structural analysis - Identifying objects, classes, and complex data structures
8. Import analysis - Determining imported libraries and external functions called

2.3 Supported processors and file formats

Ghidra supports a wide variety of processor instruction sets including X86, ARM, MIPS, PowerPC, Sparc, and more. It can analyze executable file formats like ELF and PE. Ghidra's module architecture allows support for additional processors and file types to be added.

2.4 Plugins and extensions

Ghidra has a plug-in system that lets users add new functionality. There are both analysis plugins that unlock new reverse engineering capabilities, as well as UI plugins for improved usability. Popular plugins include decompiler extensions, vulnerability detectors, and import/export utilities.

2.5 Comparison to other tools

Ghidra is often compared to IDA Pro as one of the most fully-featured reverse engineering platforms. Unlike IDA, Ghidra is free and open-source. Ghidra excels in areas like decompilation and data type analysis. Other tools like Radare2 provide scriptable command line interfaces. Ghidra balances both GUI and headless usage models.

3 Conclusion

This report has covered Ghidra, an open-source reverse engineering tool developed by the NSA. We looked at its main features, including how it disassembles, decompiles, analyzes, and modifies binary files.

Ghidra's design is modular, which means it combines different functions like disassembly and decompilation in one place. This makes it easier to use and understand. We showed how Ghidra

can turn machine code back into assembly or even high-level source code, which is useful for understanding how a program works.

Some of the best parts of Ghidra are its call graphs, which show how functions in a program interact, its ability to figure out complex data structures, and its tools for following complicated code. Ghidra also lets users both manually analyze code and automate some tasks with scripts. Plus, one can edit and change binary files directly in Ghidra.

While Ghidra is powerful, it's not a replacement for looking at the original source code of a program. But it does give a lot of insight into compiled programs. Ghidra is flexible and can be adjusted to fit different needs, and it's useful for both beginners and experts.

In today's world, where security risks might be hidden in binary files, Ghidra is a helpful tool. We hope this report has given a good starting point for using Ghidra. It's a key tool for anyone working in security, malware research, or software development.