GLOBALRAIN

Artemis Financial Vulnerability Assessment Report

Table of Contents

Document Revision History

Version	Date	Author	Comments
1.0	11/12/2022	Glenn Lehman	

Client

Developer
Glenn Lehman

1. Interpreting Client Needs

Global Rain has been contracted to provide recommendations to Artemis Financial about external security threats as they modernize their operations. Global Rain has reviewed the existing RESTful web application interface (API). While it is difficult to associate a specific dollar amount with security, having a secure interface is vital to what Artemis provides to its clients.

Any security requirements specified by the government are difficult to discover without clarification on which countries Artemis is anticipating customers to come from.

Security attacks can come from many places, as indicated in this table from NIST 800-82.

The following table is an excerpt from NIST 800-82, "Guide to Supervisory Control and Data Acquisition (SCADA) and Industrial Control System Security (SME draft), provides a description of various threats to CS networks:

Threat	Description
Bot-network operators	Bot-network operators are hackers; however, instead of breaking into systems for the challenge or bragging rights, they take over multiple systems in order to coordinate attacks and to distribute phishing schemes, spam, and malware attacks. The services of these networks are sometimes made available in underground markets (e.g., purchasing a denial-of-service attack, servers to relay spam, or phishing attacks, etc.).
Criminal groups	Criminal groups seek to attack systems for monetary gain. Specifically, organized crime groups are using spam, phishing, and spyware/malware to commit identity theft and online fraud. International corporate spies and organized crime organizations also pose a threat to the United States through their ability to conduct industrial espionage and large-scale monetary theft and to hire or develop hacker talent.
Foreign intelligence services	Foreign intelligence services use cyber tools as part of their information-gathering and espionage activities. In addition, several nations are aggressively working to develop information warfare doctrine, programs, and capabilities. Such capabilities enable a single entity to have a significant and serious impact by disrupting the supply, communications, and economic infrastructures that support military power - impacts that could affect the daily lives of U.S. citizens across the country.
Hackers	Hackers break into networks for the thrill of the challenge or for bragging rights in the hacker community. While remote cracking once required a fair amount of skill or computer knowledge, hackers can now download attack scripts and protocols from the Internet and launch them against victim sites. Thus while attack tools have become more sophisticated, they have also become easier to use. According to the Central Intelligence Agency, the large majority of hackers do not have the requisite expertise to threaten difficult targets such as critical U.S. networks. Nevertheless, the worldwide population of hackers poses a relatively high threat of an isolated or brief disruption causing serious damage.
Insiders	The disgruntled organization insider is a principal source of computer crime. Insiders may not need a great deal of knowledge about computer intrusions because their knowledge of a target system often allows them to gain unrestricted access to cause damage to the system or to steal system data. The insider threat also includes outsourcing vendors as well as employees who accidentally introduce malware into systems.
Phishers	Individuals, or small groups, who execute phishing schemes in an attempt to steal identities or information for monetary gain. Phishers may also use spam and spyware/malware to accomplish their objectives.
Spammers	Individuals or organizations who distribute unsolicited e-mail with hidden or false information in order to sell products, conduct phishing schemes, distribute spyware/malware, or attack organizations (i.e., denial of service).
Spyware/malware authors	Individuals or organizations with malicious intent carry out attacks against users by producing and distributing spyware and malware. Several destructive computer viruses and worms have harmed files and hard drives, including the Melissa Macro Virus, the Explore.Zip worm, the CIH (Chernobyl) Virus, Nimda, Code Red, Slammer, and Blaster.
Terrorists	Terrorists seek to destroy, incapacitate, or exploit critical infrastructures in order to threaten national security, cause mass casualties, weaken the U.S. economy, and damage public morale and confidence. Terrorists may use phishing schemes or spyware/malware in order to generate funds or gather sensitive information.

Source: Government Accountability Office (GAO), Department of Homeland Security's (DHS's) Role in Critical Infrastructure Protection (CIP) Cybersecurity, GAO-05-434 (Washington, D.C.: May, 2005).

While there are lots of moving parts to consider, the best starting place is the implementation of a Zero Trust Architecture (ZTA) is now mandated by executive order in the US (Kost, 2022)

2. Areas of Security

Outside of the scope of the standard Vulnerability Assessment Process Flow (VAPF), security is broken down into a few areas that directly apply to Artemis. These include user validation, design of the RESTful API, and verification of all access privileges for users and devices. Within the RESTful API itself, consideration must be given to input validation, code test, and quality, as well as encapsulation. Encryption must be evaluated to ensure that all account numbers are protected as well as the customers' private information.

It goes without saying that to implement security, we must first know who is using the system. Additionally, once verified, it is crucial to ensure that the user has the correct permissions to perform the operation they are requesting. For example, even though both a customer service rep and a client can access a client's account, only the customer should be able to initiate a transaction.

The API itself should verify all information it receives before taking action. Following the model of ensuring the user has permission and has provided the expected data

3. Manual Review

A manual review of the current code has shown the following security weakness.

- The current user cannot be passed to a method as such permission to execute cannot be verified.
- Implementation of setMyDateTime in myDateTime.java creates a large opening. Immediately you must deal with validation of all input, you would also need to expand to get time zone information. (Failure to do this can create issues with serialization) – consider using a standard library instead of this class.
- DocData takes connection information is hard codes and does not support easy deployment in multiple environments.

4. Static Testing

Static testing performed indicates the following issues with the current dependencies:

Dependency	Highest Severity	CVE Count	Conf	Evidence Count
spring-boot-2.2.4.RELEASE.jar	CRITICAL	13	Highest	32
log4j-api-2.12.1.jar	CRITICAL	5	Highest	46
tomcat-embed-core-9.0.30.jar	CRITICAL	20	Highest	39
spring-core-5.2.3.RELEASE.jar	CRITICAL	10	Highest	30
bcprov-jdk15on-1.46.jar	HIGH	16	Highest	37
snakeyaml-1.25.jar	HIGH	6	Highest	28
jackson-databind-2.10.2.jar	HIGH	4	Highest	39
spring-boot-starter-validation-2.2.4.RELEASE.jar	HIGH	1	Highest	28
logback-core-1.2.3.jar	MEDIUM	1	Highest	32
hibernate-validator-6.0.18.Final.jar	MEDIUM	1	Highest	36

The entire report is attached with details of all issues.

A sample issue is spring-boot-2.2.4.RELEASE.jar which is defined as critical. The documentation indicates that prior to v2.2.11 the temporary directory could be hijacked. Specifically this vulnerability only impacted the org.springframework.boot.web.server.AbstractConfigurableWebServerFactory.createTempDir method. This was published on March 30, 2022. Thirteen prior issues are listed for this .jar

5. Mitigation Plan

The following mitigation plans should be implemented:

a. Software Architect should determine if it is easier to update the spring framework currently being used or to design the software to ensure that the exploits are not exposed.
b. Software design patterns should be adjusted to support Zero Trust Architecture (ZTA).
c. Automated testing should be developed to exploit these vulnerabilities to confirm the RESTful API is protected. This includes ensuring the proper permissions to execute.
d. A standard review process should be created to verify any newly discovered issues with dependencies do not present a risk.
e. Additional time should be allocated during the estimating process of the various agile stories to ensure these checks are made during each sprint.

References

Kost, E. (2022, October 20). *Zero trust as a defence against supply chain attacks: Upguard*. RSS. Retrieved November 12, 2022, from https://www.upguard.com/blog/prevent-supply-chain-attacks-with-zero-trust-architecture