

**Artemis Financial Vulnerability Assessment Report**

Document Revision History

| **Version** | **Date** | **Author** | **Comments** |
| --- | --- | --- | --- |
| **1.0** | **11/12/22** | **Devin Hunter** |  |

Client



Instructions

Submit this completed vulnerability assessment report. Replace the bracketed text with the relevant information. In the report, identify your findings of security vulnerabilities and provide recommendations for the next steps to remedy the issues you have found.

* Respond to the five steps outlined below and include your findings.
* Respond using your own words. You may also choose to include images or supporting materials. If you include them, make certain to insert them in all the relevant locations in the document.
* Refer to the Project One Guidelines and Rubric for more detailed instructions about each section of the template.

Developer

Devin Hunter

* **Interpreting Client Needs**

**What is the value of secure communications to the company?**

Artemis Financial, being a financial institution, handles sensitive customer information regularly. Customers communicate sensitive data to Artemis Financial and Artemis Financial is responsible for keeping that customer information private and secure. If a security breach were to occur it could cost Artemis Financial their reputation and customers.

**Does the company make any international transactions?**

The scenario outlined in the document does not specifically mention that Artemis Financial makes international transactions. Their API is web-based which generally means that it can be accessed world-wide, and given that the scenario does not specifically mention that Artemis Financial does not make international transactions I am inclined to assume that they do.

**Are there governmental restrictions about secure communications to consider?**

Some restrictions vary state by state in the United States, but as far as Federal restrictions go there are some to consider. The Electronic Communications Privacy Act (ECPA) passed in 1986, allows the United States government to access digital communications with a subpoena unless the communications are 180 days old or over. This means that Artemis Financial would legally have to supply the government with client's information if asked. The Computer Fraud and Abuse Act (CFAA) passed in the late 1980's criminalizes the act of accessing and sharing protected information such as sensitive customer data.

**What external threats might be present now and in the immediate future?**

As always, bad actors are looking to gather client’s sensitive information in order to steal their identities, gain access to their finances, or generally use blackmail to get what they want. Cybersecurity threats are constantly growing more and more advanced, and bad actors have a plethora of techniques that they can use to bypass an institution’s security measures. Cyber criminals are learning to be coordinated, and in just the last year have launched complex, multi-stage attacks on small businesses and corporations alike. Some techniques attacks use include phishing scams, trojan horse malware, ransomware attacks, DDoS attacks, SQL injection, and more.

**What are the modernization requirements that you must consider? For example:**

**The role of open-source libraries:**

Open-source libraries are public and free to use. Because of the public nature of open source, some have the idea that it is a large security risk to use these libraries in a project. In reality, open source libraries generally have large communities dedicated to monitoring the code, fixing bugs, and identifying and closing security gaps. “A 2021 survey by Red Hat found that 89 percent of IT leaders at companies worldwide believed that open source software was just as secure or more secure than proprietary software” (Tennant, 2022). Open source libraries are okay to use when building a project, but developers should be sure to use tools such as the OWASP dependency check to weed out vulnerabilities in their project’s dependencies.

**Evolving web application technologies:**

Web app technologies and architectures significantly change over short periods of time. As a web app evolves and is maintained and upgraded, developers would do well to keep in mind the OWASP Top Ten project, which was constructed based on over 40 security firm reports and industry surveys. It lists the top ten most critical web app security flaws. Although this list covers around 90% of common security threats to web applications, OWASP has supporting documents to review as well, like the OWASP Developer’s Guide, Code Review Guide, and Testing Guide. By following these guidelines, developers can make sure that the application is secure by design, so even as it evolves, it stays secure.

* **Areas of Security and Justification**

**Input Validation**

* “Input validation is often one of the ‘first lines of defense’” (Detlefsen & Manico, 2015) for any web application. In the case of Artemis Financial, input validation will be important because their API is a RESTful API which delivers information via HTTP. Because of this, attackers could add strings to parts of the HTTP request in order to get AF’s API to act in unexpected ways.

**APIs**

* Artemis Financial has a RESTful web API. This API will allow customers to share their financial data, and possibly even allow them to link their various financial accounts with Artemis Financial’s web app so that the company can more readily give their customers the sound financial advice that they’re known for providing. Secure API interactions will be essential to Artemis’ Financial.

**Cryptography**

* Let’s say that a third party organization is attempting to access a customer’s sensitive information via Artemis Financial’s API. AF should have the ability to send this information if the third party is a trusted party, but the information should be protected while being communicated. Hashing and salting strings of information could provide that protection.

**Code Error**

* Artemis Financial may want to connect to other banking institutions in order to create a complete picture of their customer’s financial situation so that they can more readily create an accurate financial plan for them, or Artemis financial may want to let other institutions access their API so that the customer can transfer financial information to those other institutions. Either way, there is myriad room for error when attempting to access an API. These errors should be handled gracefully, and should not provide any information that would-be attackers could potentially exploit.
* **Manual Review**

**API**

* Upon inspection this app is using http rather than https.

**Input Validation**

* (Found in class GreetingController) The request parameters do not seem to be validated.

**Code Error**

* (Found in read\_document) In the catch statement it seems that an error produces a stack trace that may reveal more information than is necessary.

**General**

* From what I can gather, the username and password in DocData are both identical.
* (Found in read\_document()) Both the username and password for the database are hardcoded in.
* **Static Testing: Dependencies and Recommendations**

**Bouncy Castle version 1.46 - Update to at least BC version 1.61**

⦁ CVE-2013-1624

⦁ CVE-2015-6644

⦁ CVE-2015-7940

⦁ CVE-2016-1000338

⦁ CVE-2016-1000339

⦁ CVE-2016-1000341

⦁ CVE-2016-1000342

⦁ CVE-2016-1000343

⦁ CVE-2016-1000344

⦁ CVE-2016-1000345

⦁ CVE-2016-1000346

⦁ CVE-2016-1000352

⦁ CVE-2017-13098

⦁ CVE-2018-5382

⦁ CVE-2020-0187

⦁ CVE-2020-26939

**logback version 1.2.4 - Update logback to at least version 1.2.7**

⦁ CVE-2021-42550

**Apache log4j API version 2.12.1 - Upgrade to at least version 2.17**

⦁ CVE-2020-9488

⦁ CVE-2021-44228

⦁ CVE-2021-44832

⦁ CVE-2021-45046

⦁ CVE-2021-45105

**SnakeYAML version 1.25 - Upgrade to at least version 1.32**

⦁ CVE-2017-18640

⦁ CVE-2022-25857

⦁ CVE-2022-38749

⦁ CVE-2022-38750

⦁ CVE-2022-38751

⦁ CVE-2022-38752

**Jackson databind version 2.10.2 - Upgrade to at least version 2.14.0**

⦁ CVE-2020-25649

⦁ CVE-2020-36518

⦁ CVE-2022-42003

⦁ CVE-2022-42004

**Apache Tomcat version 9.0.30 - Upgrade to at least version 10.0.27**

⦁ CVE-2019-17569

⦁ CVE-2020-11996

⦁ CVE-2020-13934

⦁ CVE-2020-13935

⦁ CVE-2020-13943

⦁ CVE-2020-17527

⦁ CVE-2020-1935

⦁ CVE-2020-1938

⦁ CVE-2020-8022

⦁ CVE-2020-9484

⦁ CVE-2021-24122

⦁ CVE-2021-25122

⦁ CVE-2021-25329

⦁ CVE-2021-30640

⦁ CVE-2021-33037

⦁ CVE-2021-41079

⦁ CVE-2021-43980

⦁ CVE-2022-29885

⦁ CVE-2022-34305

⦁ CVE-2022-42252

**Spring Boot version 2.2.4 - Upgrade to at least version v2.2.11**

⦁ CVE-2022-27772

**Hibernate’s Bean Validation version 6.0.18 - Upgrade to at least version 6.1.3**

⦁ CVE-2020-10693

**Spring Core version 5.2.3 - Upgrade to latest version 5.2.22**

⦁ CVE-2016-1000027

⦁ CVE-2020-5421

⦁ CVE-2021-22060

⦁ CVE-2021-22096

⦁ CVE-2021-22096

⦁ CVE-2021-22118

⦁ CVE-2022-22950

⦁ CVE-2022-22965

⦁ CVE-2022-22968

⦁ CVE-2022-22970

⦁ CVE-2022-22971

* **Mitigation Plan**

Actions to take in order to fix vulnerabilities discovered in code base:

⦁ Update the dependencies in section 4 to the versions recommended.

⦁ Begin to use HTTPS protocol rather than HTTP.

⦁ Ensure database password is different from database username.

⦁ Remove the hard-coded database username and password.

⦁ Create a custom error message rather than displaying a stack trace.

⦁ Validate request parameters (Spring makes this easy to do).

**Resources:**

Erşahin, B., & Erşahin, M. (2022). Web application security. *South Florida Journal of Development*, *3*(4), 4194–4203. https://doi.org/10.46932/sfjdv3n4-002

Manico, J., & Detlefsen, A. (2015). *Iron-clad java: Building secure web applications ; best practices for secure Java Web Application Development*. Oracle Press.

NortonLifeLock Employee. (2018, January 18). *What are some of the laws regarding internet and data security?* What Are Some of the Laws Regarding Internet and Data Security? Retrieved November 12, 2022, from https://us.norton.com/blog/privacy/laws-regarding-internet-data-security#

Sullivan, T. (2022, February 2). *What is a financial api? how apps and accounts connect*. Plaid. Retrieved November 12, 2022, from https://plaid.com/resources/api/what-is-a-financial-api/

Tennant, D. (2022, August 17). *How open source software benefits businesses*. The Agile Content Platform | WordPress VIP. Retrieved November 12, 2022, from https://wpvip.com/2022/05/12/how-open-source-software-benefits-businesses/