Payatu

Payatu Casestudy

Secure Code Review for One of the Largest Online Payments Service Providers

Payatu Security Consulting Pvt. Ltd

Project Overview

In the ever-evolving landscape of financial technology, where transactions unfold at the speed of a click, the assurance of a secure and reliable payments processing application stands as the bedrock of user confidence. While competition increases, one thing that can make a player stand out from its contemporaries is its commitment to cybersecurity.

A multi-million-dollar payments application that has redefined the way we engage with digital transactions approached Payatu to get a source code review done.

At the intersection of innovation and financial convenience, this industry-leading platform has become synonymous with seamless, secure, and transformative digital payments experiences.

With multiple modules to test and review, Payatu subjected the client's source code to a meticulous and comprehensive review.

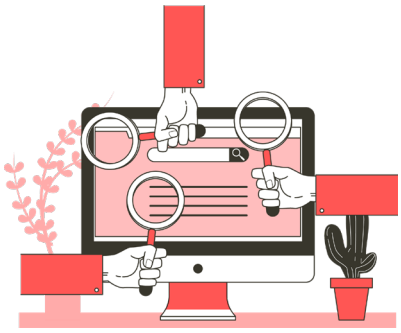
Let's take a look at this Source Code Review's journey!

The Scope

The client subjected the source code of 18 different modules to be reviewed manually and automatically using advanced in-house tools.

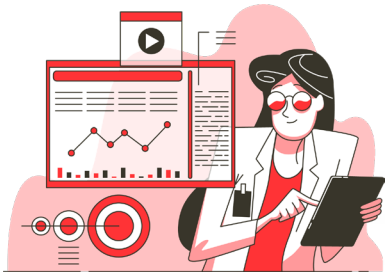
Security Assessments to be performed considering OWASP secure coding principles and Payatu's custom checklist.

Process



1. Information Gathering

The very first thing that the experts at Payatu do is systematically collect relevant data and insights to comprehensively understand the client's codebase, its structure, dependencies, and potential security risks. This phase is crucial for identifying potential vulnerabilities, ensuring compliance with coding standards, and improving overall code quality.



2. Understanding the Application and its Components

Here, the bandits understand how the codebase is organized, including the directory structure, modular components, and interdependencies.
This helps them in reviewing the code more thoroughly.

3. Establish a Review Environment

An isolated secure environment is created to conduct the review process to avoid unintentional exposure of sensitive information during the review.

4. Code Analysis

The code is now manually reviewed to identify security vulnerabilities, coding standard violations, and potential weaknesses.

The vulnerabilities covered are -

- Injection Attacks
- Broken Authentication
- Sensitive Data Exposure
- External XML Entity (XXE) Attack
- Broken Access-Control
- Security Misconfiguration
- Cross-Site Scripting
- Insecure Deserialization
- Using Component with Known Vulnerabilities
- Server-Side Request Forgery

5. Reporting and Documentation

Once the code is assessed, all findings are listed down systematically with explanation and mitigation strategies, to help the client understand the security gaps and take appropriate measures.

Findings

	High	Vulnerability
1.	Application is Vulnerable to Log4shell and other CVE's	**Vulnerable and Outdated Components**
2.	Possible Expression Language Injection	**Injection**
3.	Path Traversal Leads to File Disclosure	**Injection**
4.	Authentication is not configured for some of the Controllers	**Identification and Authentication Failures**
5.	Admin + Code Injection Using the System	**Injection**
6.	Admin + Code Injection Using tilde (``)	**Injection**
7.	Admin + Code Injection Using Exec	**Injection**

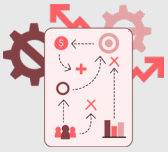
	Medium	Vulnerability
1.	Possible Path Traversal Inside Internal API Call	**Security Misconfiguration**
2.	Logging of Sensitive Details	**Security Logging and Monitoring Failures**
3.	Payment Session Timeout Value Can Be Manipulated	**Security Misconfiguration**
4.	URL Regex Possible Expression Language	**Injection**
5.	Injection Bypass	**Security Misconfiguration**
6.	Blank CSRF Tokens on Front-End Web Pages	**Security Misconfiguration**

	Medium	Vulnerability
7.	Insecure Random Function is Being Used for the Generation of the Session ID	**Cryptographic Failures**
8.	Server Name can be Controlled Using the Host Header Value	**Security Misconfiguration**
9.	Potentially Sensitive Information (VPA) Masked Client-Side	**Insecure Design**
10.	Bank Account Number Masked Client-Side	**Insecure Design**
11.	CSRF Check is Disabled	**Security Misconfiguration**
12.	S3 File Upload Overwrite	**Security Misconfiguration**
13.	JMX Endpoint Enabled	**Security Misconfiguration**
14.	Missing Authorization Check on Endpoints	**Broken Access Control**
15.	Potentially Sensitive Token Logged on Home Page	**Sensitive information disclosure**
16.	Admin + File Path Traversal	**Broken Access Control**
17.	Admin + Creating or Overwriting Existing Files	**Broken Access Control**

Many other **low** and **info** vulnerabilities were found as a part of this assessment.

Challenges

Without proper documentation, understanding the application architecture became a time-consuming activity



Expansion of Scope

Remediation

1. Upgrade the packages to the latest version.
2. Sanitize the input before passing it into any of the proxy services.
   Follow the whitelisting-based approach allowing only permissible characters.
3. Do not log sensitive details into the application.
4. Do not pass the user input directly into the message template.
5. Return the time left instead of returning the hardcoded duration.
6. Sanitize user inputs and ensure only alphanumeric characters are allowed, within the path.
7. Configure the authentication properly in the controllers.
8. Implement a proper check for CSRF protection.
9. Ensure that CSRF tokens are being used & validated on the server side.
10. Use SecureRandomGenerator for creating the session id .
11. Mask the VPA on the server, to ensure that unmasked values can't be read by the client.
12. Mask the bank account number on the server, to ensure that unmasked values can't be read by the client by any means.
13. Use DateTimeFormatter which is thread safe instead of SimpleDateFormat.
14. Disable/Secure the JMX metric endpoint.
15. Proper authorization check should be implemented for all the endpoints based on different privileges required in the application.
16. Remove the token from the client-side source code.
17. Avoid incorporating user-controllable data into dynamically evaluated code.
18. Data should be strictly validated/sanitized. Ideally, a whitelist of specific accepted values should be used.

Payatu

About Payatu

Payatu is a Research-powered cybersecurity services and training company specialized in IoT, Embedded Web, Mobile, Cloud, & Infrastructure security assessments with a proven track record of securing software, hardware and infrastructure for customers across 20+ countries.

Code Review

Payatu's Secure Code Review includes inspecting, scanning and evaluating source code for defects and weaknesses. It includes the best secure coding practices that apply security consideration and defend the software from attacks.

Web Security Testing

Internet attackers are everywhere. Sometimes they are evident. Many times, they are undetectable. Their motive is to attack web applications every day, stealing personal information and user data. With Payatu, you can spot complex vulnerabilities that are easy to miss and guard your website and user's data against cyberattacks.

DevSecOps Consulting

DevSecOps is DevOps done the right way. With security compromises and data breaches happening left, right & center, making security an integral part of the development workflow is more important than ever. With Payatu, you get an insight to security measures that can be taken in integration with the CI/CD pipeline to increase the visibility of security threats.

Payatu

Product Security

Save time while still delivering a secure end-product with Payatu. Make sure that each component maintains a uniform level of security so that all the components “fit” together in your mega-product.

Cloud Security Assessment

As long as cloud servers live on, the need to protect them will not diminish. Both cloud providers and users have a shared. As long as cloud servers live on, the need to protect them will not diminish.

Both cloud providers and users have a shared responsibility to secure the information stored in their cloud Payatu’s expertise in cloud protection helps you with the same. Its layered security review enables you to mitigate this by building scalable and secure applications & identifying potential vulnerabilities in your cloud environment.

IoT Security Testing

IoT product security assessment is a complete security audit of embedded systems, network services, applications and firmware. Payatu uses its expertise in this domain to detect complex vulnerabilities & security loopholes to guard your IoT products against cyberattacks.

Red Team Assessment

Red Team Assessment is a goal-directed, multidimensional & malicious threat emulation. Payatu uses offensive tactics, techniques, and procedures to access an organization’s crown jewels and test its readiness to detect and withstand a targeted attack.

Mobile Security Testing

Detect complex vulnerabilities & security loopholes. Guard your mobile application and user's data against cyberattacks, by having Payatu test the security of your mobile application.

Critical Infrastructure Assessment

There are various security threats focusing on Critical Infrastructures like Oil and Gas, Chemical Plants, Pharmaceuticals, Electrical Grids, Manufacturing Plants, Transportation systems etc. and can significantly impact your production operations. With Payatu's OT security expertise you can get a thorough ICS Maturity, Risk and Compliance Assessment done to protect your critical infrastructure.

CTI

The area of expertise in the wide arena of cybersecurity that is focused on collecting and analyzing the existing and potential threats is known as Cyber Threat Intelligence or CTI. Clients can benefit from Payatu's CTI by getting – social media monitoring, repository monitoring, darkweb monitoring, mobile app monitoring, domain monitoring, and document sharing platform monitoring done for their brand.

More Services Offered

- AI/ML Security Audit
- Trainings

More Products Offered

- EXPLIoT
- CloudFuzz

Payatu

Payatu Security Consulting Pvt. Ltd.

www.payatu.com

info@payatu.io

+91 20 41207726