July 17, 2024

SECURITY AUDIT REPORT FOR WeaUp AWS WEB SERVER

WMDD – 4950 Security & Cloud & Server Admin

Wonnyo Hamester | Lead Developer
Bethleen Baral (Beck) | Developer
Wilm Reinhardt Botha | Developer
Victor Portus | Developer

snəẃeyəɬ leləm̓.
THE COLLEGE OF HIGHER LEARNING.

Langara.
THE COLLEGE OF HIGHER LEARNING.

Contents

1 Executive Summary

The security audit of the WeaUp AWS web server revealed several vulnerabilities of varying severity, including potential exposure of cloud metadata, accessible sensitive files, and information leaks. The most critical issue identified was a misconfigured NGINX server that could potentially allow unauthorized access to instance metadata, posing a high risk to system security. Additionally, medium-risk vulnerabilities were found related to exposed Git configuration files and accessible .htaccess files, which could leak sensitive information.

To address these issues and enhance overall security, we recommend implementing a series of actions focused on improving server configuration, AWS infrastructure security, and monitoring practices. Key recommendations include rectifying the NGINX configuration, securing Git repositories, implementing AWS best practices such as encryption and proper access controls, and enhancing monitoring through AWS CloudWatch and CloudTrail. By addressing these vulnerabilities and implementing the suggested improvements, WeaUp can significantly strengthen its security posture and reduce the risk of potential breaches or unauthorized access.

2 Overview of WeaUp AWS Server

The backend API server of WeaUp is hosted on an Amazon AWS EC2 instance. The instance is a t2.micro (1 vCPU, 1 GB RAM, 20 GB HDD) running Linux Ubuntu. The server stack includes Node.js and Express, managed by PM2, with NGINX as the reverse proxy. Communication with the presentation layer is secured using JWT authorization for private endpoints. Additionally, Google Authentication is implemented to facilitate user authentication.

3 Methodology

We performed the OWASP ZAP test on our endpoints and API, we then compared our server against the AWS Guidelines and analysed our Google Authentication implementation.

4 OWASP ZAP Test Findings

4.1 About the test

4.1.1 Contexts

No contexts were selected, so all contexts were included by default.

4.1.2 Sites

The following sites were included:
http://dev.api.weaup.io

4.1.3 Risk levels

Included: High, Medium, Low, Informational
Excluded: None

4.1.4 Confidence levels

Included: User Confirmed, High, Medium, Low
Excluded: User Confirmed, High, Medium, Low, False Positive

4.2 Summaries

4.2.1 Alert counts by risk and confidence

This table shows the number of alerts for each level of risk and confidence included in the report.

(The percentages in brackets represent the count as a percentage of the total number of alerts included in the report, rounded to one decimal place.)

		Confidence				
		User Confirmed	High	Medium	Low	Total
Risk	High	0 (0.0%)	0 (0.0%)	0 (0.0%)	1 (25.0%)	1 (25.0%)
	Medium	0 (0.0%)	1 (25.0%)	1 (25.0%)	0 (0.0%)	2 (50.0%)
	Low	0 (0.0%)	0 (0.0%)	0 (0.0%)	0 (0.0%)	0 (0.0%)
	Informational	0 (0.0%)	0 (0.0%)	1 (25.0%)	0 (0.0%)	1 (25.0%)
	Total	0 (0.0%)	1 (25.0%)	2 (50.0%)	1 (25.0%)	4 (100%)

4.2.2 Alert counts by site and risk

This table shows, for each site for which one or more alerts were raised, the number of alerts raised at each risk level. Alerts with a confidence level of "False Positive" have been excluded from these counts.

(The numbers in brackets are the number of alerts raised for the site at or above that risk level.)

		Risk			
		High (= High)	Medium (>= Medium)	Low (>= Low)	Informational (>= Informational)
Site	http://dev.api.weaup.io	1 (1)	2 (3)	0 (3)	1 (4)

4.2.3 Alert counts by alert type

This table shows the number of alerts of each alert type, together with the alert type's risk level.

(The percentages in brackets represent each count as a percentage, rounded to one decimal place of the total number of alerts included in this report.)

Alert type	Risk	Count
Cloud Metadata Potentially Exposed	High	1 (25.0%)
.htaccess Information Leak	Medium	5 (125.0%)
Hidden File Found	Medium	6 (150.0%)
User Agent Fuzzer	Informational	96 (2400.0%)
Total		4

4.3 Alerts

4.3.1 Risk=High, Confidence=Low (1)

Cloud Metadata Potentially Exposed(1)
GET http://dev.api.weaup.io/latest/meta-data/

Alert tags	▪ OWASP_2021_A05 ▪ OWASP_2017_A06
Alert description	The Cloud Metadata Attack attempts to abuse a misconfigured NGINX server in order to access the instance metadata maintained by cloud service providers such as AWS, GCP and Azure. All of these providers provide metadata via an internal unroutable IP address '169.254.169.254' - this can be exposed by incorrectly configured NGINX servers and accessed by using this IP address in the Host header field.
Other info	Based on the successful response status code cloud metadata may have been returned in the response. Check the response data to see if any cloud metadata has been returned. The meta data returned can include information that would allow an attacker to completely compromise the system.
Request	Request line and header section (206 bytes) GET http://dev.api.weaup.io/latest/meta-data/ HTTP/1.1 host: 169.254.169.254 user-agent: Mozilla/5.0 (Windows NT 10.0; rv:125.0) Gecko/20100101 Firefox/125.0 pragma: no-cache cache-control: no-cache Request body (0 bytes)
Response	Status line and header section (348 bytes) HTTP/1.1 200 OK X-Content-Type-Options: nosniff Surrogate-Control: no-store cache-control: private, max-age=0 Pragma: no-cache Expires: 0

	expo-protocol-version: 0 expo-sfv-version: 0 content-type: text/plain Vary: Accept-Encoding Date: Tue, 23 Jul 2024 19:42:58 GMT Connection: keep-alive Keep-Alive: timeout=5 content-length: 2990 Response body (2990 bytes) ** Actual response is excluded from this report*
Attack	169.254.169.254
Solution	Do not trust any user data in NGINX configs. In this case it is probably the use of the $host variable which is set from the 'Host' header and can be controlled by an attacker.

4.3.2 Risk=Medium, Confidence=High (1)

Hidden File Found(1)
GET http://dev.api.weaup.io/.git/config

Alert tags	▪ OWASP_2021_A05 ▪ OWASP_2017_A06 ▪ CWE-538 ▪ WSTG-v42-CONF-05
Alert description	A sensitive file was identified as accessible or available. This may leak administrative, configuration, or credential information which can be leveraged by a malicious individual to further attack the system or conduct social engineering efforts.
Other info	git_dir

Request	Request line and header section (201 bytes) GET http://dev.api.weaup.io/.git/config HTTP/1.1 host: dev.api.weaup.io user-agent: Mozilla/5.0 (Windows NT 10.0; rv:125.0) Gecko/20100101 Firefox/125.0 pragma: no-cache cache-control: no-cache Request body (0 bytes)
Response	Status line and header section (423 bytes) HTTP/1.1 200 OK X-Content-Type-Options: nosniff Surrogate-Control: no-store Cache-Control: no-store, no-cache, must-revalidate, proxy-revalidate Pragma: no-cache Expires: 0 Accept-Ranges: bytes Last-Modified: Mon, 22 Jul 2024 01:02:44 GMT ETag: W/"1153-190d7f65994" Content-Type: application/octet-stream Content-Length: 4435 Date: Tue, 23 Jul 2024 18:46:09 GMT Connection: keep-alive Keep-Alive: timeout=5 Response body (4435 bytes) ** Actual response is excluded from this report*
Evidence	HTTP/1.1 200 OK
Solution	Consider whether or not the component is actually required in production, if it isn't then disable it. If it is then ensure access to it requires appropriate authentication and authorization, or limit exposure to internal systems or

specific source IPs, etc.

4.3.3 Risk=Medium, Confidence=Medium (1)

.htaccess Information Leak (1)

GET http://dev.api.weaup.io/.git/.htaccess

Alert tags	▪ CWE-94 ▪ OWASP_2021_A05 ▪ OWASP_2017_A06 ▪ WSTG-v42-CONF-05
Alert description	htaccess files can be used to alter the configuration of the Apache Web Server software to enable/disable additional functionality and features that the Apache Web Server software has to offer.
Request	Request line and header section (204 bytes) GET http://dev.api.weaup.io/.git/.htaccess HTTP/1.1 host: dev.api.weaup.io user-agent: Mozilla/5.0 (Windows NT 10.0; rv:125.0) Gecko/20100101 Firefox/125.0 pragma: no-cache cache-control: no-cache Request body (0 bytes)
Response	Status line and header section (348 bytes) HTTP/1.1 200 OK X-Content-Type-Options: nosniff Surrogate-Control: no-store cache-control: private, max-age=0 Pragma: no-cache

	Expires: 0 expo-protocol-version: 0 expo-sfv-version: 0 content-type: text/plain Vary: Accept-Encoding Date: Tue, 23 Jul 2024 19:42:59 GMT Connection: keep-alive Keep-Alive: timeout=5 content-length: 2996 Response body (2996 bytes) ** Actual response is excluded from this report*
Evidence	HTTP/1.1 200 OK
Solution	Ensure the .htaccess file is not accessible.

4.3.4 Risk=Informational, Confidence=Medium (1)

User Agent Fuzzer (1)
GET http://dev.api.weaup.io/api/.htaccess

Alert tags	
Alert description	Check for differences in response based on fuzzed User Agent (eg. mobile sites, access as a Search Engine Crawler). Compares the response statuscode and the hashcode of the response body with the original response.
Request	Request line and header section (185 bytes) GET http://dev.api.weaup.io/api/.htaccess HTTP/1.1 host: dev.api.weaup.io user-agent: Mozilla/4.0 (compatible; MSIE 8.0; Windows NT 6.1) pragma: no-cache cache-control: no-cache

	Request body (0 bytes)
Response	Status line and header section (348 bytes) HTTP/1.1 200 OK X-Content-Type-Options: nosniff Surrogate-Control: no-store cache-control: private, max-age=0 Pragma: no-cache Expires: 0 expo-protocol-version: 0 expo-sfv-version: 0 content-type: text/plain Vary: Accept-Encoding Date: Tue, 23 Jul 2024 19:43:04 GMT Connection: keep-alive Keep-Alive: timeout=5 content-length: 2996 Response body (2996 bytes) ** Actual response is excluded from this report*
Parameter	Header User-Agent
Attack	Mozilla/4.0 (compatible; MSIE 8.0; Windows NT 6.1)

4.4 Additional Information

4.4.1 Alert types

This section contains additional information on the types of alerts in the test report.

4.4.1.1 Cloud Metadata Potentially Exposed

Source	raised by an active scanner (Cloud Metadata Potentially Exposed)
Reference	▪ https://www.nginx.com/blog/trust-no-one-perils-of-trusting-user-input/

4.4.1.2 htaccess Information Leak

Source	raised by an active scanner (.htaccess Information Leak)
CWE ID	94
WASC ID	14
Reference	▪ https://developer.mozilla.org/en-US/docs/Learn/Server-side/Apache_Configuration_htaccess ▪ https://httpd.apache.org/docs/2.4/howto/htaccess.html

4.4.1.3 Hidden File Found

Source	raised by an active scanner (Hidden File Finder)
CWE ID	538
WASC ID	13
Reference	▪ https://blog.hboeck.de/archives/892-Introducing-Snallygaster-a-Tool-to-Scan-for-Secrets-on-Web-Servers.html ▪ https://git-scm.com/docs/git-config

4.4.1.4 User Agent Fuzzer

Source	raised by an active scanner (User Agent Fuzzer)
Reference	▪ https://owasp.org/wstg

5 AWS Security Guidelines

5.1 AWS Responsibilities

AWS is responsible for protecting the infrastructure that runs AWS services in the AWS Cloud. AWS also provides you with services that you can use securely. Third-party auditors regularly test and verify the effectiveness of our security as part of the AWS Compliance Programs.

5.2 Cloud Manager Responsibilities

Our responsibility includes the following areas:

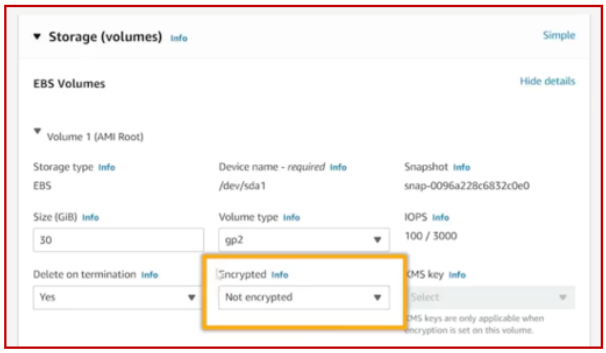
- Controlling network access to your instances
- Managing the credentials used to connect to your instances.
- Managing the guest operating system and software deployed to the guest operating system, including updates and security patches.
- Configuring the IAM roles attached to the instance and the permissions associated with them.

5.3 Best practices

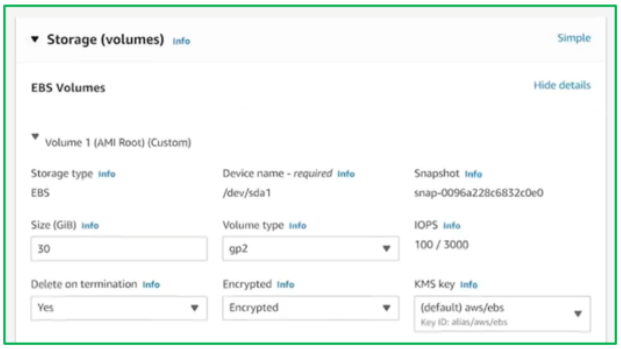
5.3.1 Encrypt Electronic Block Storage (EBS) Volumes and Snapshots

Encryption ensures security on both data rest and data in transit between an instance and its attached EBS storage.

Not encrypted



Encrypted



5.3.2 Regularly backup EBS using Amazon EBS snapshots.

AWS does not automatically back up data stored on Amazon EBS volumes. For backup and disaster recovery it is a cloud costumer's responsibility to create regular backups using Amazon EBS snapshots.

5.3.3 Restrict access to your EC2 instance using security groups

Define the network access with Inbound rules, meaning traffic coming into your instance and outbound meaning traffic going outside.

5.3.4 Regularly patch Elastic Compute Cloud (EC2) instances

You can use the AWS System Manager Patch Manager to automate the patch process.

5.3.5 Configure Virtual Private Cloud (VPC) flow logging settings to monitor the network traffic reaching EC2 instances

Use VPC to flow logs, that's the virtual private cloud flow logs, to monitor network traffic that reaches the EC2 instance

5.4 How to apply these best practices

5.4.1 CloudWatch

CloudWatch allows you to view data from all your AWS services in a single console providing a unified view of your operational health. Data from different services is organized into namespaces. There are multiple options available for metrics, statistic type and monitoring time range, giving you the flexibility to track in a way that works for you. CloudWatch also allows you to create alarms to get notified when critical metrics breach configure thresholds. A robust monitoring solution must also support automation. CloudWatch Events, now known as Amazon EventBridge, allows you to continuously monitor event patterns and trigger remediation actions using Lambda function.

Amazon CloudWatch integrates with AWS IAM, allowing you to specify which CloudWatch actions a user can perform. But IAM cannot control access to CloudWatch

data for specific resources when permissions are granted to the AIM, they will be able to see all cloud resources you use with CloudWatch.

5.4.2 CloudWatch Alarms

Given a set of parameters, we can define rules to send out notifications when the conditions are met. One of the indispensable configurations is the Billing alarms to monitor monetary cost fluctuations due to our resources' usage.

CloudWatch logs allows you to collect and store logs from your resources, applications, and services in near real time.

Here are the steps to set up a CloudWatch alarm:

- Create a log group
- Create a VPC flow logs linked to an IAM role that has permissions to the EC2 instance
- Create custom flow log metric filter for the logs that interest us the most, for example connections to our EC2 via SSH
- Create CloudWatch alarm associated with the custom metric

5.4.3 CloudWatch Logs Insights

It allows you to search and analyze your data in CloudWatch Logs interactively. Logs Insights automatically generates fields from your logs and provides a purpose-built language to query these fields.

For example, you can query information like the top 10 IPs with the highest number of TCP sessions.

5.4.4 CloudWatch Agent

- Collect internal system-level metrics from Amazon EC2 instances.
- Collect system-level metrics from on-premises services
- Collect logs from Amazon EC2 instances and on-premises servers
- Collect custom metrics from your applications for quickly spotting problems

Metrics collected by the CloudWatch agent are stored in the namespace called CWAgent.

5.4.5 CloudTrail

AWS CloudTrail allows you to log and retain account activity enabling governance, compliance, and risk audition across your AWS infrastructure. It provides event history for actions performed through the AWS Management Console, AWS SDKs, command-line tools, and other AWS services.

This tool is mostly used to record and view the actions that occurred on the AWS Console by other users, combined with CloudWatch logs we can set up alarms to different actions to have a robust and secured monitoring system over our AWS environment.

5.5 Action items to enhance WeaUp AWS infrastructure and monitoring

5.5.1 Server backups

- Attach an encrypted EBS volume to our EC2 instance
- To regularly back up EBS volumes using Amazon EBS snapshots, we can use AWS Data Lifecycle Manager (DLM). AWS DLM automates the creation, retention, and deletion of EBS snapshots

5.5.2 Network

- Restrict the access to our EC2 instance using security groups. As we have a GitHub Action Workflow handling the continuous deployment of our API, we can limit the SSH traffic to only allow the IP addresses we want, in this case, the GitHub Actions IP Ranges. GitHub maintains a list of their IP addresses in a JSON file which you can fetch from their official endpoint. You can find the list here: GitHub Meta API.

- Create a VCP to monitor the network traffic reaching our EC2 instance. We can also set up a CloudWatch alarm to notify SSH's failed attempts to connect. This gives us more visibility who is trying to connect to our resources

5.5.3 Financial and Growth

- Set up a Billing Alarm

5.5.4 Performance

- Install and create a CloudWatch Agent to monitor the CPU and performance of the EC2 instance. A basic level can be enough for our project.
- We can also enable CloudWatch Agent to register the logs of a file inside our EC2 for example the PM2 service or NGINX logs to monitor our API from the AWS Console.

5.5.5 AWS Security and Monitoring

- All the resources should have an IAM with dedicated access to each resource.
- CloudTrail allows us to monitor the actions performed on our AWS console by all IAM users in our organization.

6 Google Authentication

The application utilized Google authentication, which provides a secure and convenient way for users to access the application using their Google accounts. This was implemented along with JWT, to facilitate secure communication between the application and backend API.

6.1 Implementation

6.1.1 Google Authentication Flow

The implementation follows the OAuth 2.0 Authentication Code Flow with PKCE (Proof Key for Code Exchange)

- **Authorization Request:** Initiates authentication by redirecting users to Google's OAuth consent screen.
- **User Consent:** Users grant permissions to the application, specifying which Google account can be accessed.
- **Token Exchange:** Upon user consent, the application exchanges an authorization code for an access token

6.1.2 JWT Usage

JWT is used for secure communication between the application and backend API. Upon successful Google authentication, the backend issues a JWT containing user claims.

6.1.3 Libraries Used

- **expo-auth-session**: Used for handling OAuth flows securely in the React Native environment.
- **expo-auth-session/providers/google**: Used to integrate with Google's OAuth services to handle authentication requests and responses.
- **jsonwebtoken:** Enables encoding, decoding, and verification of JWTs to ensure secure handling of tokens and facilitate token-based authentication between the app and server.

6.2 Recommendations

Below are some recommendations for future implementation to further enhance security:

- Implement multi-factor authentication (MFA).
- Regularly review and update JWT expiration policies and token refresh mechanisms.
- Conduct regular security audits and vulnerability assessments.

7 Discovered Vulnerabilities Details

Cloud Metadata Potentially Exposed (Risk: High, Confidence: Low)

- The Cloud Metadata Attack attempts to abuse a misconfigured NGINX server to access instance metadata.
- This could potentially allow an attacker to completely compromise the system.

Hidden File Found (.git/config accessible) (Risk: Medium, Confidence: High)

- A sensitive file (.git/config) was identified as accessible.
- This may leak administrative, configuration, or credential information.

.htaccess Information Leak (Risk: Medium, Confidence: Medium)

- The .htaccess file is accessible, which can reveal server configuration details.

User Agent Fuzzer (Risk: Informational, Confidence: Medium)

- Differences in response based on fuzzed User Agent were detected.

8 Action Items

NGINX Configuration:

- Review and correct NGINX configuration to prevent Cloud Metadata exposure.
- Ensure that user data is not trusted in NGINX configs, especially the use of the $host variable.

Git Repository Security:

- Remove or secure access to the .git directory on the production server.
- Implement proper .gitignore rules to prevent sensitive files from being committed.

.htaccess Protection:

- Ensure .htaccess files are not accessible from the web.
- Configure server to deny direct access to .htaccess files.

AWS Security Enhancements:

- Encrypt EBS Volumes and Snapshots.

- Set up regular backups of EBS volumes using Amazon EBS snapshots.
- Restrict access to EC2 instances using properly configured security groups.
- Implement regular patching for EC2 instances, potentially using AWS Systems Manager Patch Manager.
- Configure VPC flow logging to monitor network traffic to EC2 instances.

Monitoring and Alerting:

- Set up CloudWatch alarms for critical metrics.
- Implement CloudWatch Logs Insights for log analysis.
- Install and configure CloudWatch Agent on EC2 instances for enhanced monitoring.
- Set up a Billing Alarm to monitor AWS costs.

Network Security:

- Limit SSH access to specific IP ranges, such as GitHub Actions IP ranges for CI/CD.
- Create a VPC to better control and monitor network traffic.

IAM and Access Management:

- Review and refine IAM roles and permissions for all AWS resources.
- Implement the principle of least privilege for all IAM users and roles.

Logging and Auditing:

- Enable CloudTrail for comprehensive AWS account activity logging.
- Set up log retention and analysis procedures.

Google Authentication Enhancements:

- Implement multi-factor authentication (MFA).
- Regularly review and update JWT expiration policies and token refresh mechanisms.

Security Audits:

- Conduct regular security audits and vulnerability assessments.
- Perform periodic penetration testing of the application and infrastructure.

9 References

https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/ec2-security.html

https://www.linkedin.com/learning/aws-monitoring-logging-and-remediation/monitoring-logging-and-remediation-in-aws?resume=false&u=57075641

https://www.linkedin.com/learning/cloud-security-and-audit-fundamentals-aws-microsoft-azure-and-google-cloud/security-basics-in-aws-elastic-compute-cloud-ec2?resume=false&u=57075641

https://www.zaproxy.org/getting-started/

https://docs.expo.dev/versions/latest/sdk/auth-session/

https://datatracker.ietf.org/doc/html/rfc7519