Beep security report

What is Beep?

Beep is a discord-like web application that allows users to communicate in real-time through text, voice, and video.

Project deployment diagram

The following diagram illustrates the deployment architecture of Beep, including its components and their interactions:

Threats & security risk assessment

Currently, Beep does not have security risk assessment mechanisms in place. To address this issue, we have conducted a security risk assessment of Beep's API, web UI, and infrastructure defined as code, as well as automated such processes in CI. Conducting this has allowed us to identify potential threat vectors and Common Vulnerabilities and Exposures (CVEs). The assessment was performed using various tools, including IriusRisk for threat modeling and risk assessment, Snyk for static code analysis, and trivy for container image scanning. Dependencies were also analyzed for known vulnerabilities.

Security risk assessment with IriusRisk

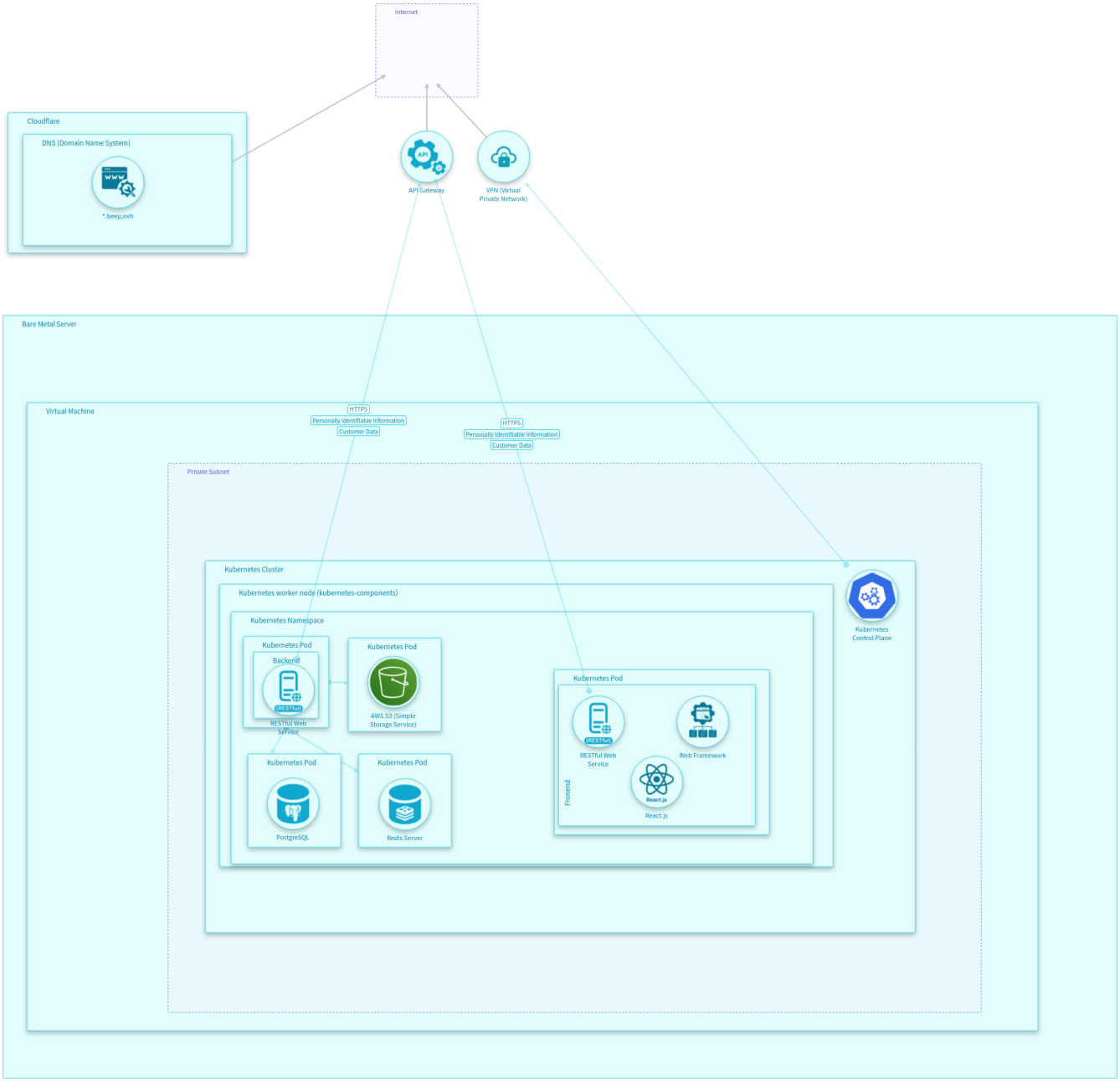
IriusRisk was used to perform threat modeling and security risk assessment of the Beep platform, identifying potential vulnerabilities across all application components.

Methodology

Our assessment followed a four-step approach: Asset Identification of critical systems and data; Threat Modeling using IriusRisk templates for web applications; Risk Evaluation based on likelihood and impact; and Countermeasure Planning for significant threats.

Key Findings

The threat modeling identified several high-priority concerns including unauthorized administrative access, DNS vulnerabilities (spoofing and unencrypted traffic), insufficient logging and monitoring, access control weaknesses, and infrastructure security gaps in Kubernetes deployment.



Critical Threats

Our assessment revealed several critical security threats to the Beep platform:

Authentication Issues: Unauthorized administrative access poses significant risk of configuration changes and data breaches. Inadequate RBAC implementation allows users to potentially access unauthorized resources.

Network Vulnerabilities: DNS spoofing could redirect users to malicious websites, while unencrypted DNS traffic exposes user browsing patterns. Insecure Kubernetes PKI key file permissions could enable unauthorized system access.

Operational Weaknesses: Insufficient logging prevents detection of malicious activities. Lack of rate limiting exposes services to denial-of-service attacks, particularly affecting Redis Server performance.

Data Security Concerns: Insecure file permissions on PostgreSQL databases and inadequate

protection of secrets management increase risk of data exposure.

Potential Mitigations

Short-term Actions: Our immediate recommendations focus on critical security controls. Implementing multi-factor authentication for administrative interfaces will significantly reduce unauthorized access risk. Applying proper file permissions (600) to Kubernetes PKI keys is essential to prevent unauthorized system access. Enabling comprehensive logging across all critical services will help detect suspicious activities. Encrypting DNS traffic using DoH or DoT protocols will protect against DNS-based attacks and eavesdropping.

Medium-term Strategy: For sustained security improvement, we recommend deploying Role-Based Access Control throughout the application stack to enforce the principle of least privilege. Implementing rate limiting and resource throttling will protect services from denial-of-service attacks. Establishing secure secrets management workflows will safeguard sensitive credentials. Setting up real-time monitoring and alerting systems will enable rapid response to security incidents.

Long-term Approach: To build a security-focused culture, we suggest developing comprehensive security training for all developers to ensure secure coding practices. Establishing automated security testing in the CI/CD pipeline will catch vulnerabilities early in development. Creating detailed incident response procedures for various threat scenarios will improve recovery time. Implementing regular security assessments and penetration testing will continuously identify and address new vulnerabilities.

The implementation of these mitigations, prioritizing the highest-risk items first, will substantially improve Beep's security posture and reduce vulnerability to attacks.

Security risk assessment with Snyk

Security risk assessment with Trivy