Lacerba S.r.l.
P.IVA/C.F.: 02824220343
Sede Legale: Viale Solferino, 3/1, 43123 Parma (PR)
Sede Operativa: Via Friuli, 72, 20135 Milano (MI)

Lacerba SaaS Technical Documentation

January 4th 2019

App Security

1. Authentication

We have our own database of users. For each of our clients we use a dedicated database, which ensures that no user account is shared across our different customers.
The server never stores the password in plain text! We use the Bcrypt algorithm to hash them.

2. Session management

The back-end exposes its endpoints following the REST conventions, meaning it's stateless. The front-end (consuming the API) therefore sends its authentication token at each request for validation.
We provide an endpoint to generate these tokens when a user logs in.
Tokens are Json Web Tokens. They replace the session information usually stored in cookies.
The token is signed server-side using the HMAC-SHA256 algorithm with a highly secured secret of 40 characters. This secret is specific for each of our clients. This ensures that if the secret of one of our client is compromised, the rest of our clients remains safe.

SaaS Infrastructure Details

3. Hosting

Lacerba SaaS applications are all hosted on Heroku PaaS.

All the details of Heroku Infrastrcuture are documented on its official public documentation (https://www.heroku.com/policy/security)

The server are physically located in Europe

4. Infrastructure Security Policies

DDOS prevention

Heroku comes with a standard DDOS protection techniques including TCP Syn cookies and connection rate limiting in addition to maintaining multiple backbone connections and internal bandwidth capacity that exceeds the Internet carrier supplied bandwidth

Spoofing and Sniffing Protections

Managed firewalls prevent IP, MAC, and ARP spoofing on the network and between virtual hosts to ensure spoofing is not possible. Packet sniffing is prevented by infrastructure including the hypervisor which will not deliver traffic to an interface which it is not addressed to. Heroku utilizes application isolation, operating system restrictions, and encrypted connections to further ensure risk is mitigated at all levels.

Port Scanning

Port scanning is prohibited and every reported instance is investigated by our infrastructure provider. When port scans are detected, they are stopped and access is blocked.

5. Data Security

Heroku Postgres

Application data is stored in separate access-controlled database that require a unique username and password that is only valid for that specific database and is unique. Connections to Postgres databases require SSL encryption to ensure a high level of security and privacy. Stored data can be encrypted by customer applications in order to meet data security requirements.

6. Backups

Customer Applications

Applications deployed to the Heroku platform are automatically backed up as part of the deployment process on secure, access controlled, and redundant storage. Heroku uses these backups to deploy the application and to automatically bring the application back online in the event of an outage.

Postgres Databases

Continuous Protection keeps data safe on Heroku Postgres. Every change to data is written to write-ahead logs, which are shipped to multi-datacenter, high-durability storage. In the unlikely

event of unrecoverable hardware failure, these logs can be automatically 'replayed' to recover the database to within seconds of its last known state. We also store daily backups of the database.

7. Vulnerability Management

Heroku vulnerability management process is designed to remediate risks without impacting Lacerba SaaS application. Heroku is notified of vulnerabilities through internal and external assessments, system patch monitoring, and third party mailing lists and services. Each vulnerability is reviewed to determine if it is applicable to Heroku's environment, ranked based on risk, and assigned to the appropriate team for resolution.

8. Disaster Recovery

Customer Applications and Databases

Heroku platform automatically restores our application and Heroku Postgres database in the case of an outage. The Heroku platform is designed to dynamically deploy the application within the Heroku cloud, monitor for failures, and recover failed platform components.

Heroku Platform

The Heroku platform is designed for stability, scaling, and inherently mitigates common issues that lead to outages while maintaining recovery capabilities. The platform maintains redundancy to prevent single points of failure, is able to replace failed components, and utilizes multiple data centers designed for resiliency. In the case of an outage, the platform is deployed across multiple data centers using current system images and data is restored from backups. Heroku reviews platform issues to understand the root cause, and improve the platform and processes.

Rollbar Add-ons

We use Rollbar add-ons to monitor the status of the application. In case of outage or application error we receive immediate notification we details of the error.