SECURITY REQUIREMENTS

Common Security Requirements – Encircle Qualifications

1. **The Provider must have a documented Incident Response Plan that includes immediate notification to customer upon qualification of a breach**

   Yes, Encircle has a documented Incident Response plan in place today with notifications set to be issued within 4 hours of any incident.

2. **The Provider must ensure that all Provider personnel with access to the service systems are trained and qualified to operate the systems that provide the service. Each provider employee must be provided sufficient security awareness training. Provider employees must have signed confidentiality and conflict of interest agreements.**

   Yes, Encircle has a documented Incident Response plan in place today with notifications set to be issued within 4 hours of any incident.

3. **The Provider must have a dedicated and qualified, accountable individual named as the privacy officer**

   Ronuk Raval, CTO, also acts as the privacy officer of the company.

4. **The Provider must demonstrate that they have a robust change management capability for the systems providing the service.**

   Encircle has a robust change management process in place to ensure that only validated software is released from development to staging and then ultimately to production.

5. **The Provider must demonstrate commitment to best practices for enterprise architecture including the implementation of a tiered architecture for the services delivered.**

   The Encircle platform is multi-tiered and our server infrastructure is also multi-tiered.

6. **The Provider must demonstrate that they have a web application security assessment performed (internally or externally) for the services provided as appropriate.**

   Encircle is currently in the process of having an external penetration test performed on our infrastructure. We will be happy to provide any customer with a copy of the final report.

7. **All user credentials maintained by the Provider must be protected with the currently accepted encryption controls. Passwords must be hashed and only the hashes stored on a separate system from the application. Password management for self-service reset requires user authentication using known information (email or phone number).**

   Encircle has chosen to protect user credentials using bcrypt (https://en.wikipedia.org/wiki/Bcrypt) as our research shows it is more secure and reliable for generating keys from passwords.

8. **The Provider must store confidential material in an encrypted form while not processing it. Acceptable encryption algorithms are AES256 and higher or other NIST approved algorithms. Key Management processes and policies must be in place.**

   All data is encrypted via TLS during transit and encrypted for backups via AES. Data in the database would remain in the processing state and would not be encrypted.

9. **The Provider is expected to operate a centralized monitoring and logging solution to manage Security information and events (SIEM).**

   Encircle does not currently have a centralized monitoring and logging solution in place today but rather is leveraging a number of tools to provide monitoring and logging. We are investigating the effort involved to move towards a centralized solution.

10. **The Provider must have a policy and practice in place to ensure that the systems composing the solution delivering service is patched regularly and maintained at an N-1 state.**

    Confirmed. Encircle ensures patches are regularly performed and maintained.

11. **The Provider must ensure that physical security of their data center and all office space is sufficient to control access to the sites. Only authorized personnel are to be permitted access.**

    Encircle's infrastructure is hosted on the Microsoft Azure platform where only key employees of Microsoft are provided with physical access to the servers. Documentation can be retrieved from Microsoft to confirm their processes as necessary.

12. **The Provider must inform clients of the location (Country, State/Province/District, and city) of the data centers where data is to be stored and processed.**

    Encircle's servers are currently located in the "Canada East" zone of Microsoft Azure.

SQL INJECTION

Encircle uses an Object Relational Mapper called SQLAlchemy to formulate all database queries. SQLAlchemy automatically separates query parameters from the query itself. It is thus impossible for user provided input to escape the query parameter and perform a SQL Injection.escape the query parameter and perform a SQL Injection.

CROSS-SITE SCRIPTING (XSS)

Encircle uses tools for the server (Tornado Templates) and clients (React) that automatically sanitize user input from the structure of the web page, preventing cross-site scripting entirely. As an additional layer of protection, Content Security Policy headers are given, disabling the loading of scripts or content from any domain not specifically whitelisted.

TIMING ATTACKS

Timing attacks are a specific breed of side channel attacks where an attacker exploits the time taken by the server to respond to reason about various code paths.

All authentication for the Encircle API is handled by a single core layer that is independent of timing. Each code path is the same length and therefore no sensitive information can be inferred to attack the authentication layer.

CRIME AND BREACH

The BREACH (and its predecessor CRIME) attacks exploit compression artifacts in the core connection protocols of the internet to reveal sensitive information. Announced in August 2013, it is comparable in scope to Heartbleed, though it is not as easy to exploit. However, a large number of services continue to be vulnerable to this.

BREACH relies on sensitive information being in the document for every request an active attacker makes. Encircle generates a random nonce for each request and only responds with the secure token exclusively OR'd with the nonce. This way, the token is represented differently for each request, which mitigates BREACH but still allows the clients to reconstruct the actual token.