CodeIQ Security Program & Software

Security practices & standards to protect client data and project integrity.

Anti-virus / Malware Policy

Connected hardware is routinely checked and maintained for anti-virus and malware protection. Anti-virus software is deployed across all workstations and devices, providing a secure environment for all team members.

Sensitive Information

Client data is securely stored on production servers with access restricted only to designated employees. Backups are managed in secure, limited-access environments as determined by the platform requirements. Data encryption is available upon request to protect sensitive information. Client data and sensitive information, when stored on paper, is shredded before disposal to ensure confidentiality.

Data Access Control

Only team members directly involved in a project have access to its production environment. Access is promptly revoked upon the termination of employment or project completion to uphold data security.

Technology

Software & Frameworks

Only team members directly involved in a project have access to its production environment. Access is promptly revoked upon the termination of employment or project completion to uphold data security.

- Laravel: Built-in features for encryption, authentication, and data protection.
- Shopify: for e-commerce solutions and webshops.

Security updates are applied when a security contract is agreed upon with the client. Otherwise, updates may be requested or done on an hourly basis.

Hosting Provider

Our virtual servers are hosted by 4AllBusiness Telecom BV and managed via Ploi.io. All servers are accessible only via SSH keys, with no password-based access. We do not offer shared hosting; all servers are privately managed and secured with two-factor authentication (2FA).

Security Components for Laravel Projects

1. **Secure Communication – HTTPS**
   We request SSL certificates for all projects to ensure secure connections between the server and the client.
2. **SSH Access - Private Keys**
   SSH access to production servers is managed through private keys, which are secured by passphrases and only provided to a limited group of developers.
3. **Password Hashing - Bcrypt with Unique Salt**
   User passwords are hashed one-way with unique salts, ensuring they cannot be reversed.
4. **Sensitive Data Encryption - AES Encryption**
   Sensitive data can be encrypted using AES encryption, with an encryption key stored securely outside the web environment and version control.
5. **SQL Injection Prevention - Parameterized Queries**
   Parameterized queries are used through Laravel's ORM to prevent SQL injection attacks.
6. **XSS Protection - Safe Output by Default**
   Secure templating engines like Twig or Blade are used to prevent XSS attacks, with variable escaping enabled by default.
7. **CSRF Protection**
   A CSRF token is generated for every session and required for all POST/PUT/DELETE requests to prevent cross-site request forgery.

8. **Authentication and Authorization**
   Laravel's built-in authentication features protect user credentials and restrict access to specific routes based on login status and user roles.
9. **Validation**
   Laravel's validation tools securely process user input, ensuring required file types and unique rows.
10. **Configuration Variables**
    Sensitive configuration details are stored in a .env file, inaccessible from version control and web environments, available only on the production server.

Version Control and Backups

Application code is maintained in a Git repository without sensitive information or database passwords. Backups of critical data are kept within secured, restricted environments.