RIETTA.COM™ /SECURITY

We believe you should be confident that your web app infrastructure protects customer data! It starts with software development practices.

Before **#DevSecOps** was trendy, we were making sure that security was part of the development process for web applications. We simplify development practices so your business can distinguish itself as a company that customers trust with their data!

Let us help write your recipe for success! Contact us today.

https://rietta.com

hello@rietta.com

(888) 250-6435 ⋄ (770) 623-2059

5805 State Bridge Road. Suite G 158. Johns Creek, GA 30097

Recipe for a Strong Application Security Program

1. All developers should be familiar with the OWASP Top 10 and OWASP Proactive Security Controls to understand common ways applications are compromised in production.
2. Add security to user stories & acceptance test criteria:
    1. Document security constraints in each story.
    2. Write abuser stories from the point of view of a malicious adversary for things the system shall not allow!
    3. Your developers can practice threat modeling methods, such as referring to attack libraries and STRIDE as needed.
3. Practice test driven development where developers write a failing automated test for each user story acceptance criteria.
4. Configure your automated test suite to run static analysis tools and treat a high confidence failure as a failed test.
5. Practice peer code review and require automated tests be written that address the requirements and security constraints.
6. Implement continuous integration/continuous deployment (CI/CD) merge gates that do not allow code to be merged into production when automated tests fail.
7. Configure your CI/CD to run security scans on the existing code base at least nightly, and raise high priority alerts for the DevSecOps team anytime a previously passing test now fails.
8. Require that all deployments to production must go through the CI/CD pipeline at all times.
9. When an external penetration test finds any vulnerability, have developers write failing automated tests before writing a fix.
10. Encourage your team to always be learning about the latest security threats and to participate in OWASP and security community events.