Microsoft

SH#FT Happens!! How to deploy DevSecOps principles in the development lifecycle.

Joylynn Kirui – Senior Cloud Security Advocate, Microsoft

https://twitter.com/joylynn_kirui

52% of companies

sacrifice cybersecurity for speed

57% of ops teams

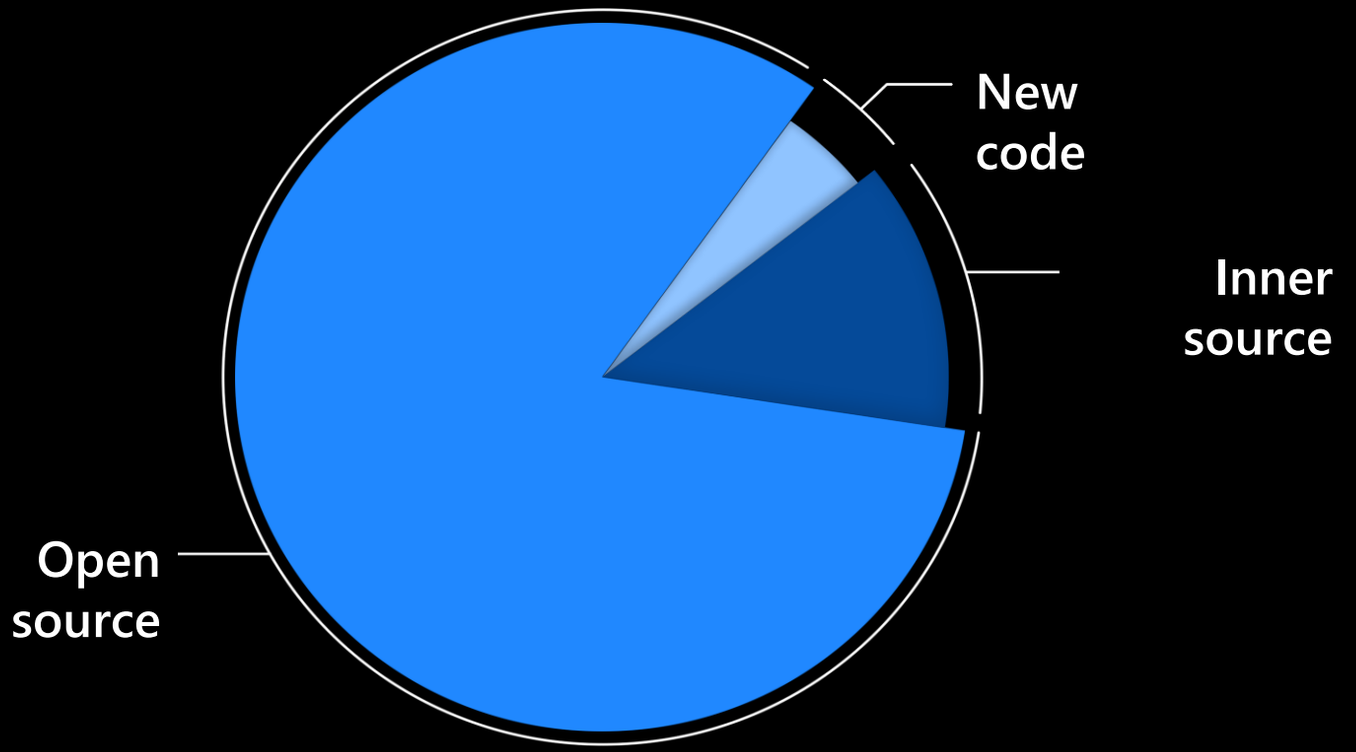
push back on security best practices

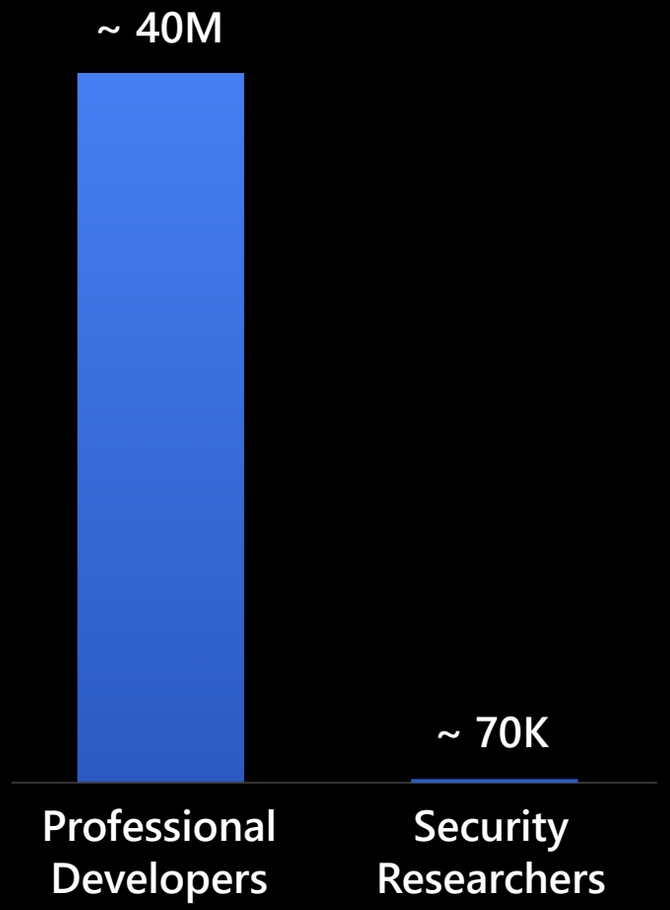
44% of developers

are not trained to code securely

Source: Threat Stack, 2018

80-90% of the code in new applications comes from open source.

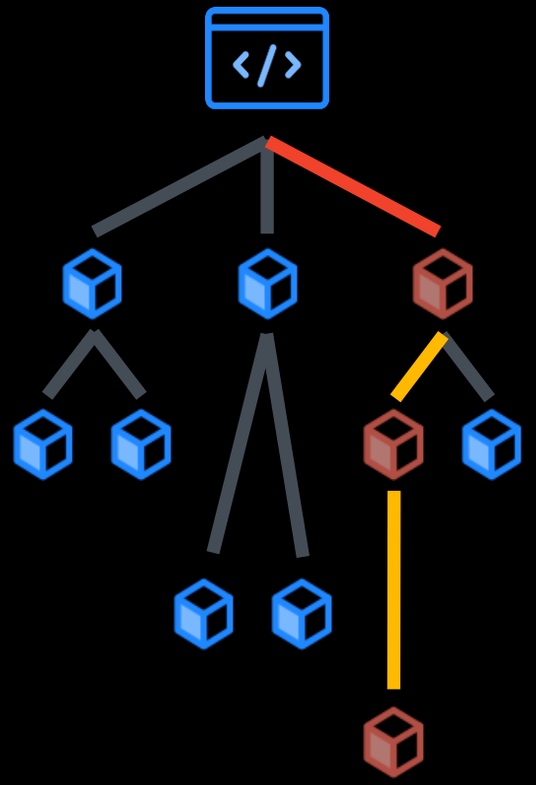


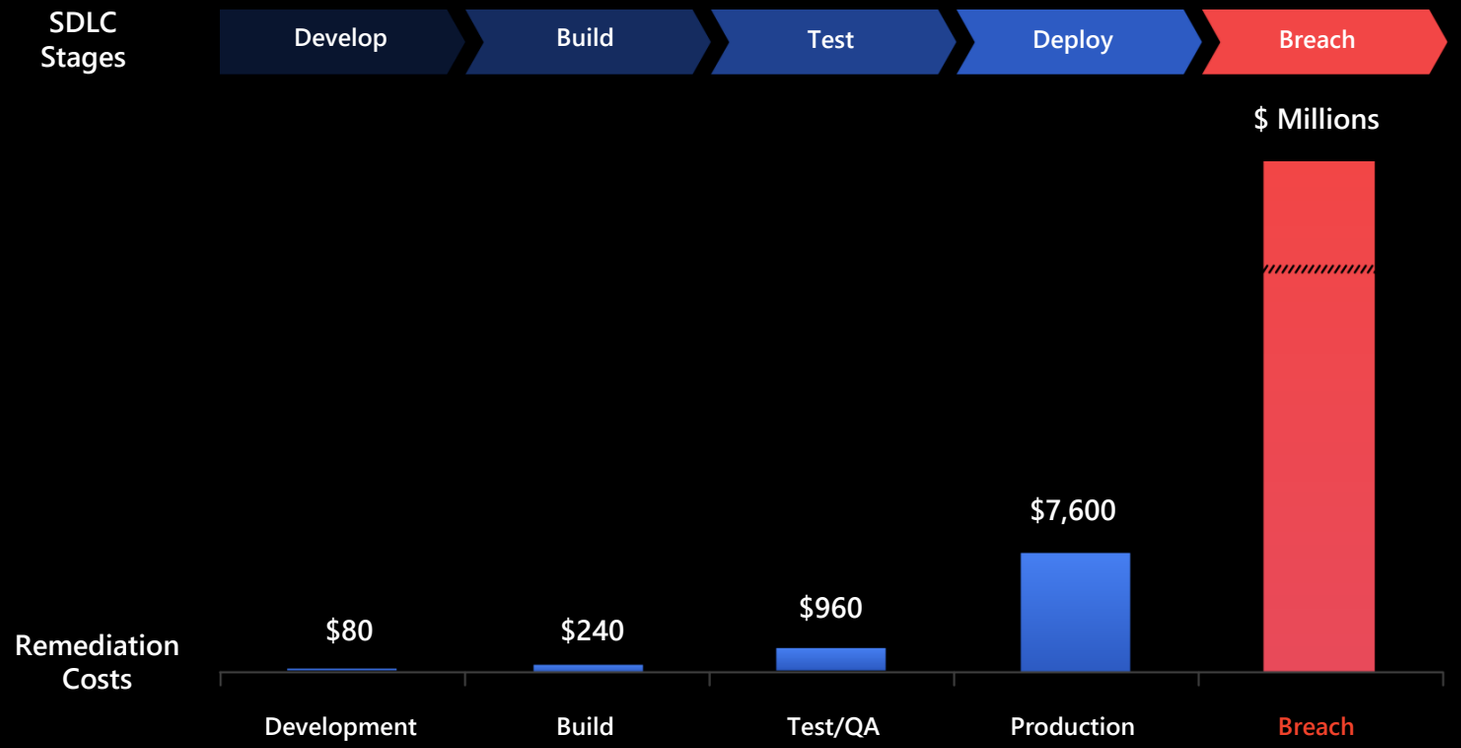


There 570x more developers than security researchers

Other sources of vulnerabilities

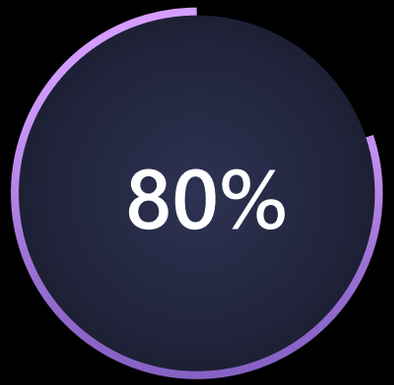
- Unchecked dependencies (80-90% of your code)
- Employee error (exposed access tokens, unsafe code patterns)
- 570x more developers than security researchers
- Damage is exponentially greater if it reaches production



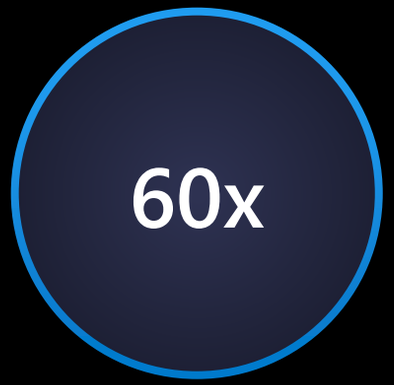


Sources: NIST, Polemon Institute

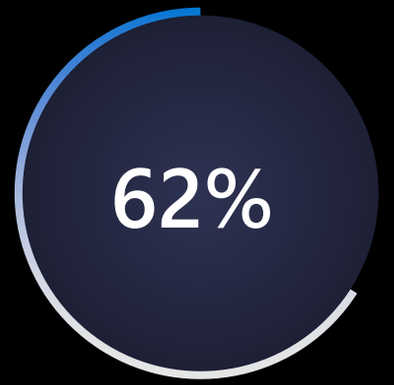
Importance of shifting security left



reduction in security incidents by extending security to development[2]



Security cost to fix a security defect in production versus in development[1]



of enterprises do not integrate security in the development phase[3]

[1]National Institute of Standards and Technology
[2]https://www.gartner.com/smarterwithgartner/is-the-cloud-secure/
[3]Sources: McKinsey Developer Velocity, Microsoft Enterprise DevOps Report, GitHub Octoverse Report 2020

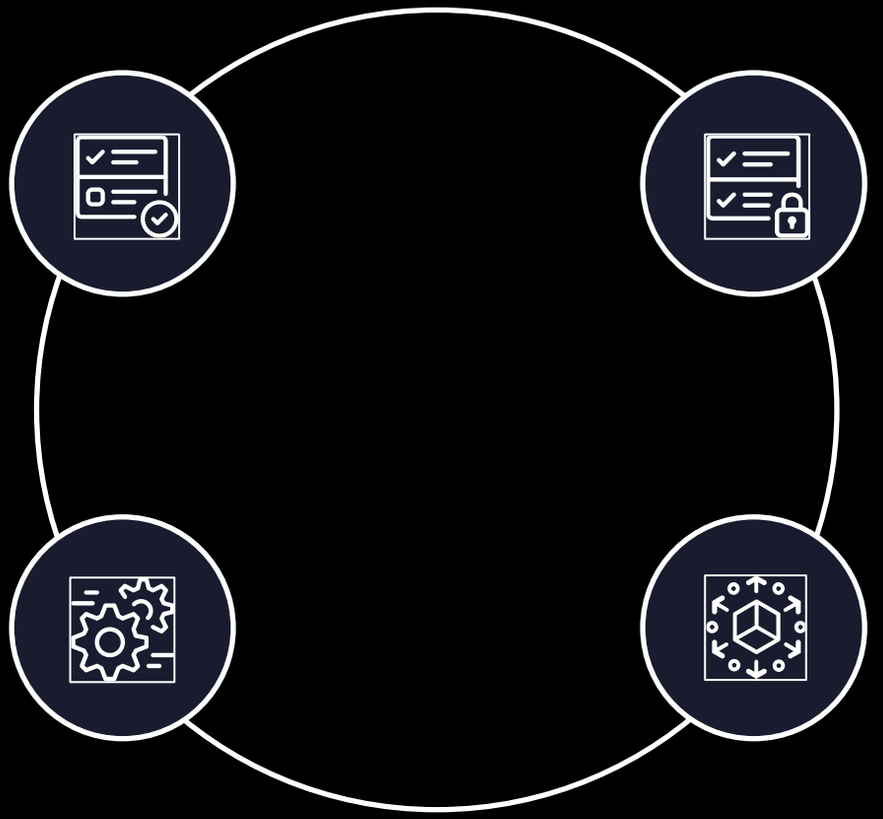
How security fits in the development lifecycle

PRE-COMMIT

- Threat modeling
- IDE security plug-in
- Pre-commit hooks
- Secure coding standards
- Peer review

OPERATE & MONITOR

- Continuous monitoring
- Threat intelligence
- Blameless post-mortems



COMMIT (CI)

- Static Application Security Testing (SAST)
- Security unit tests
- Dependency management / Software Composition Analysis (SCA)
- Credential scanning

DEPLOY (CD)

- Infra as code (IaC)
- Dynamic security scanning
- Cloud configuration checks
- Security acceptance tests

Run static & dynamic analysis

AUTOMATED SECURITY REVIEW AND TESTING THROUGHOUT THE DEVOPS LIFECYCLE

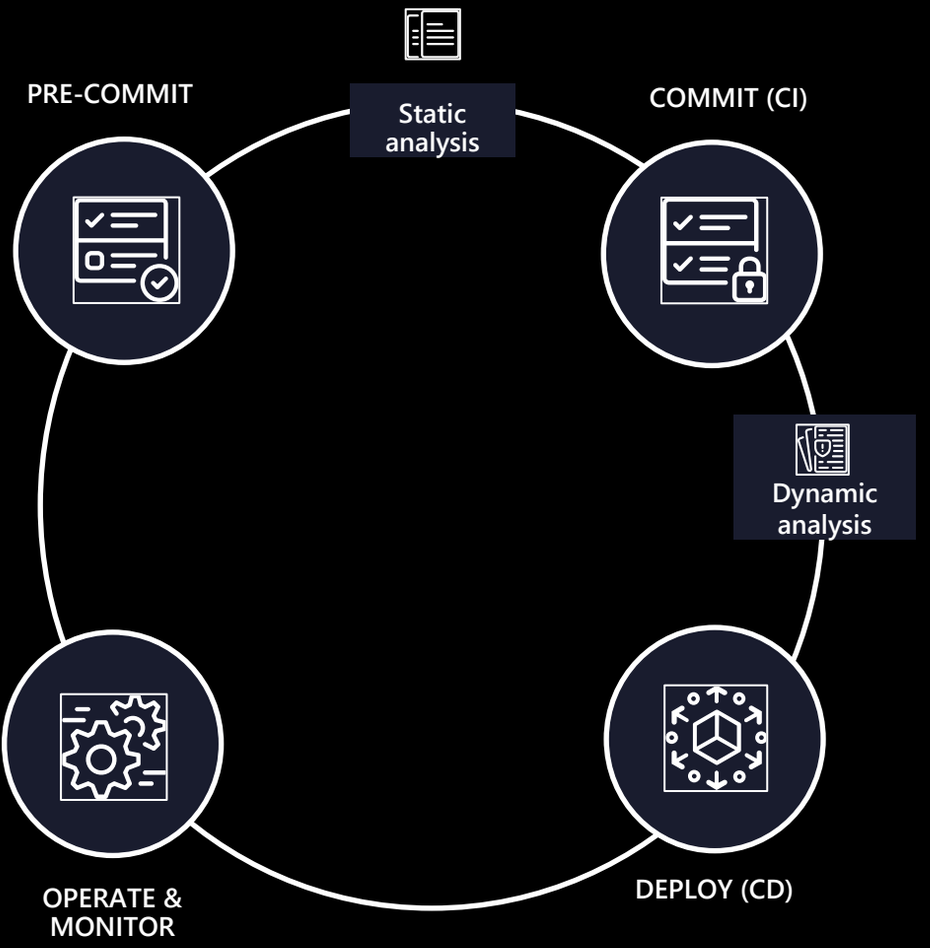
Automations:

Automated security review of code

Automated simulated attacks targeting running application

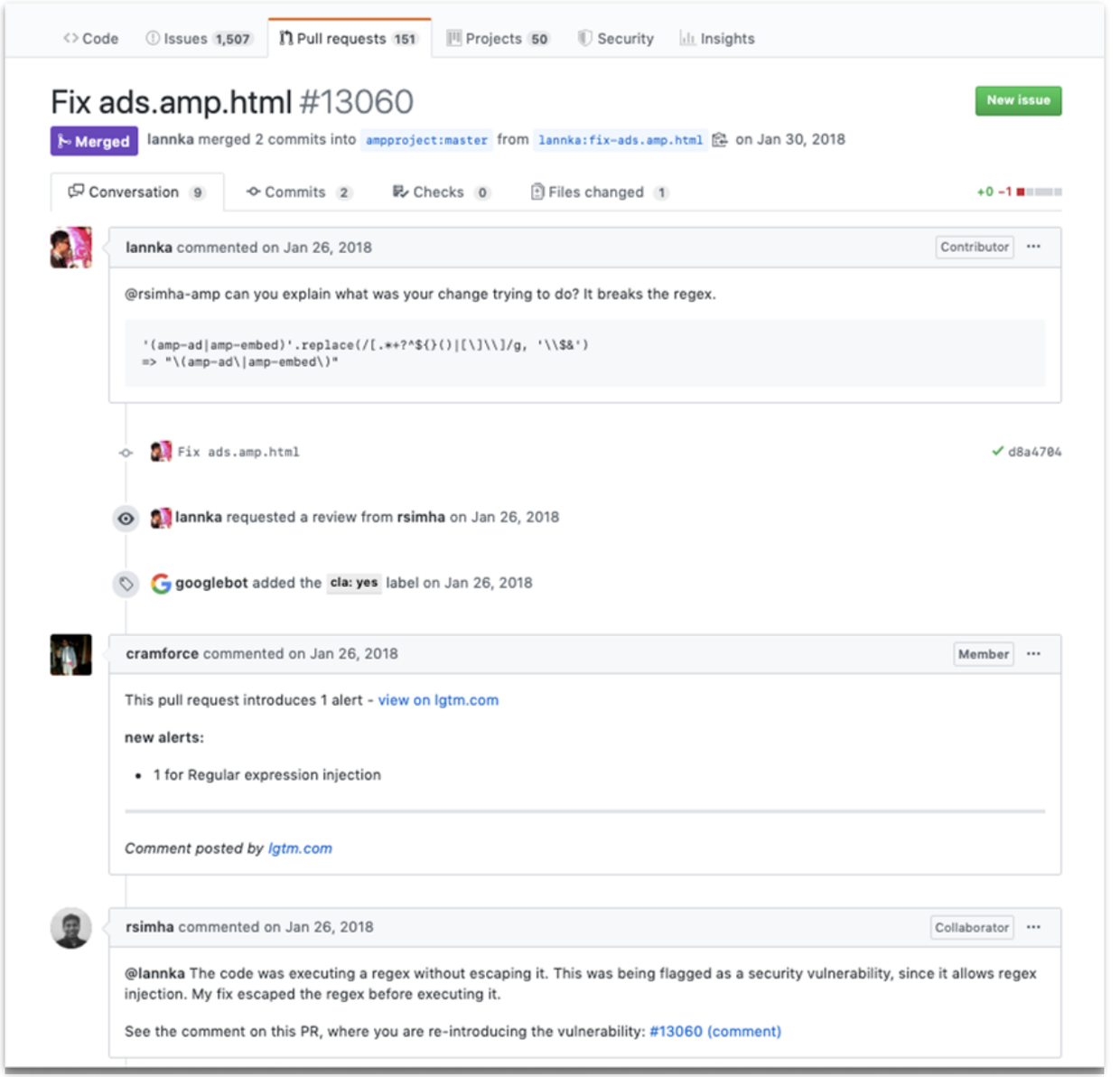
PREVENT THESE TYPES OF ATTACKS:

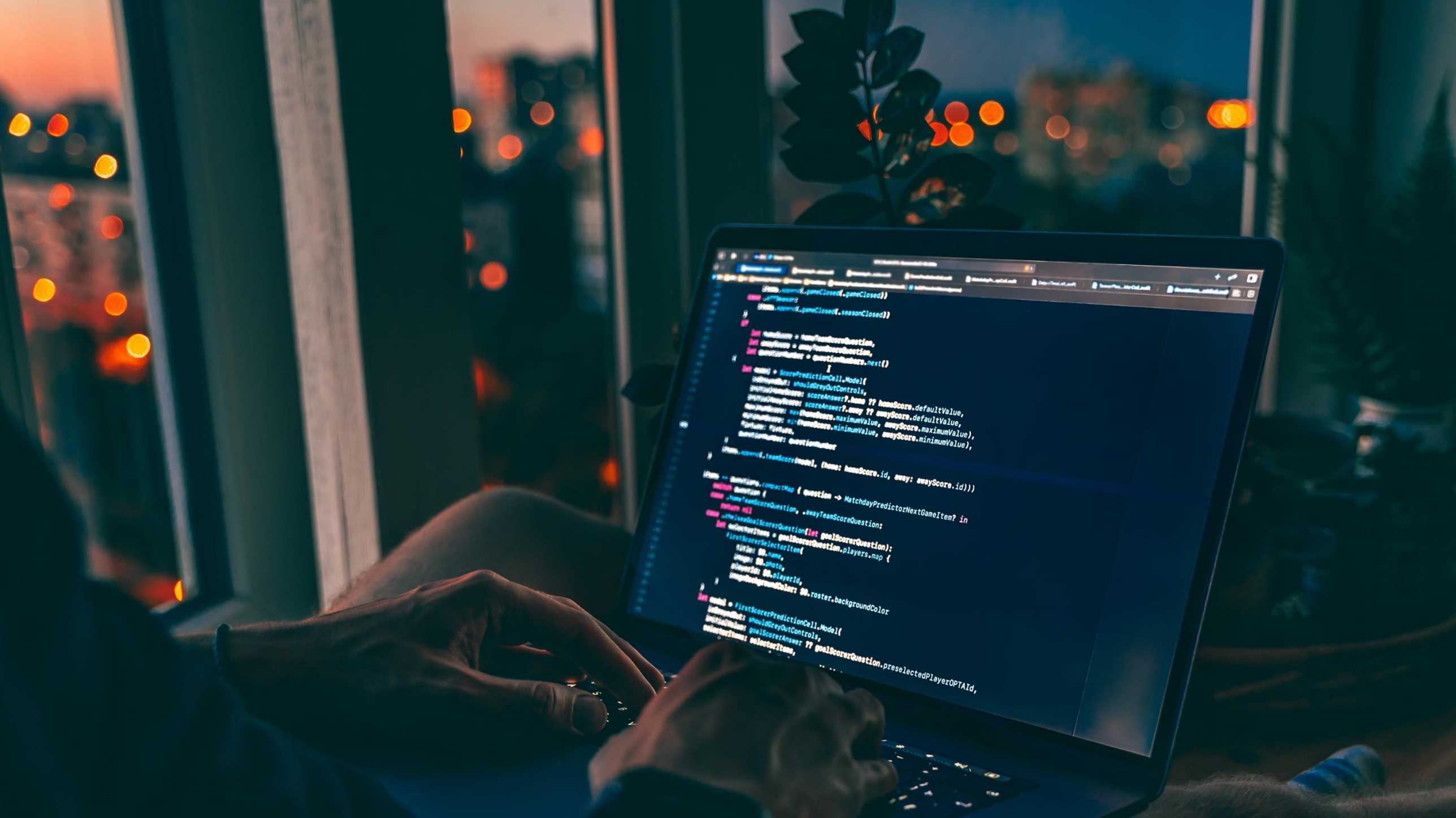
- Common technical application security attacks



Code Scanning

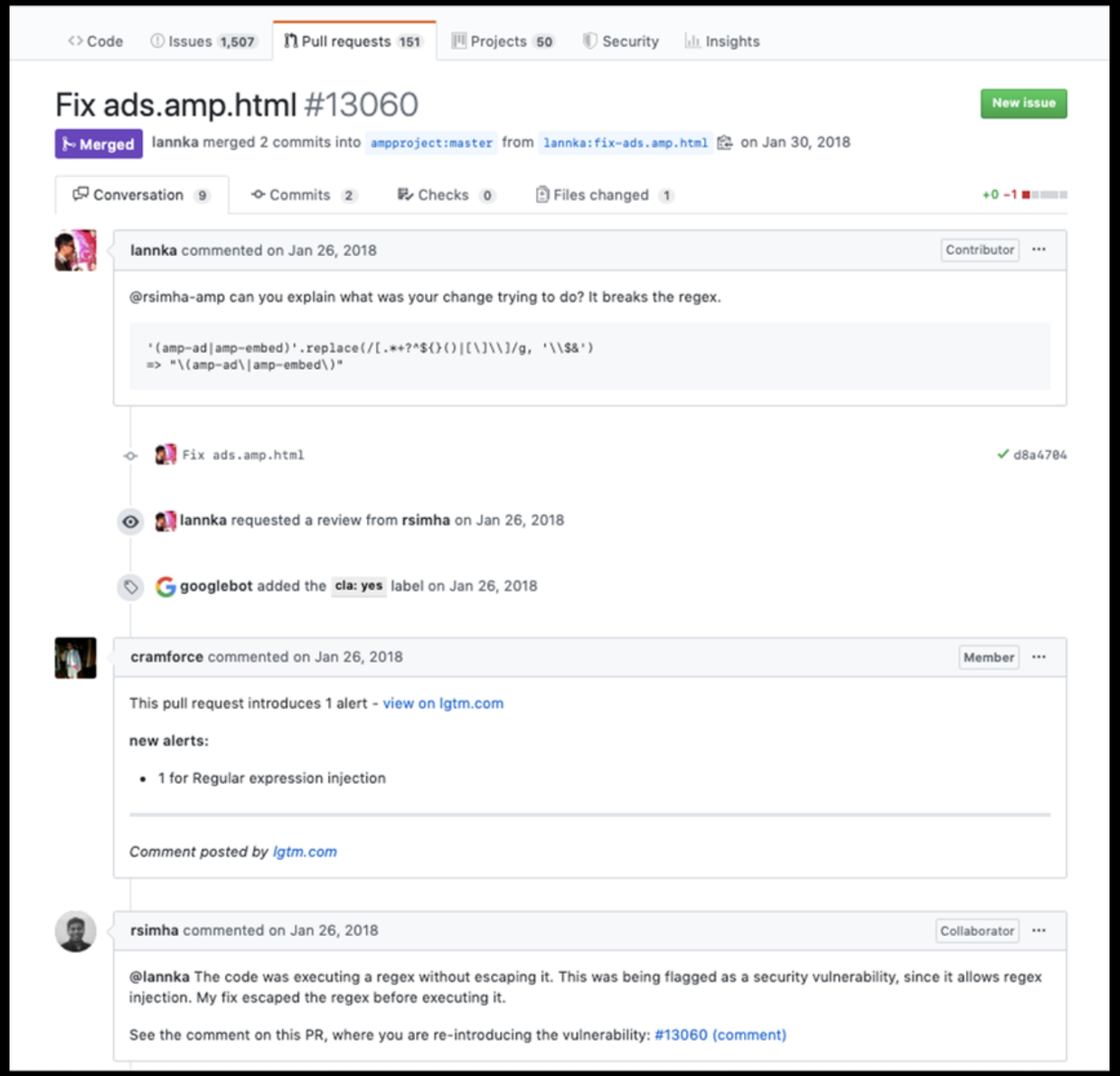
- CodeQL: The world's most advanced semantic code engine
- Community-driven query set brings top experts to your team
- Customize & build new queries to adapt to your specific threat topology and to find variants
- Extensible, with support for DAST and other SAST tools





aka.ms/DevSecOpsSolution

https://codeql.github.com/docs/codeql-for-visual-studio-code/



Resources

1. Configure Microsoft Security DevOps GitHub Actions – https://learn.microsoft.com/en-us/azure/defender-for-cloud/github-action

2. Connect your GitHub repositories to Microsoft Defender for Cloud - https://learn.microsoft.com/en-us/azure/defender-for-cloud/quickstart-onboard-github

3. DevOps Security Workbook - https://techcommunity.microsoft.com/t5/microsoft-defender-for-cloud/devops-security-workbook/ba-p/3637662

Thank you

https://twitter.com/joylynn_kirui