Empowering Security Champions with Essential Tooling

Material used throughout the pipeline

- Security Champions leverage various security tools to identify and mitigate vulnerabilities across different stages of the **Software Development Life Cycle (SDLC)**. These tools assist in enhancing application security by detecting **code-level risks**, **third-party vulnerabilities**, and **secrets exposure**.
- **Types of Security Analysis Tools**
- **Static Application Security Testing (SAST):** Analyzes self-written code for security vulnerabilities before deployment.
- **Software Composition Analysis (SCA):** Detects vulnerabilities in third-party libraries and dependencies.
- **Infrastructure-as-Code (IaC) Scanning:** Ensures secure configurations in cloud deployments.
- **Secrets Detection:** Identifies hardcoded secrets, such as credentials and API keys.
- **Container Scanning:** Examines container images for vulnerabilities to secure deployment environments.

Commonly Used Security Tools

- Security Champions are trained to utilize industry-standard security tools, including:
  - **OWASP ZAP** (Dynamic Application Security Testing - DAST)
  - **Burp Suite** (Web Application Security Testing)
  - **API Security Testing** using automated solutions such as OWASP APISec
  - **CI/CD-integrated Security Tools** to ensure continuous security validation
- These tools enable Security Champions to detect and remediate vulnerabilities **early in the pipeline** and validate the effectiveness of security fixes before deployment.

Benefits, incentives

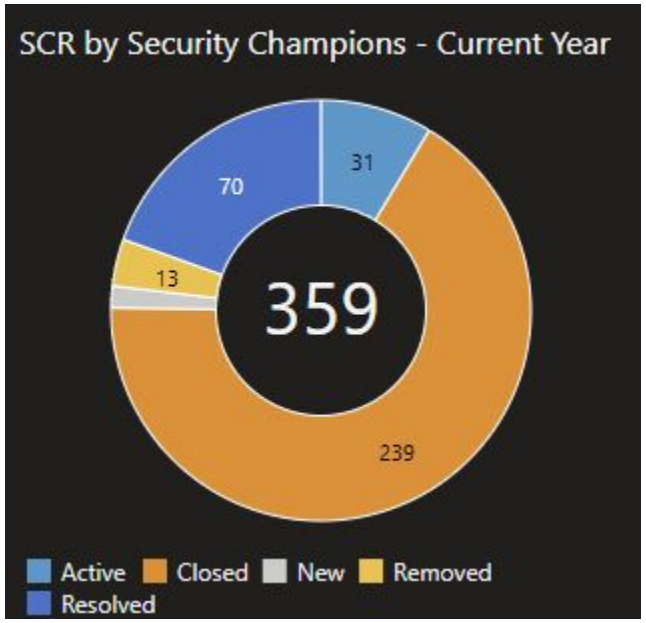
- By integrating multiple security tools across the **SDLC stages**, Security Champions can:
  - Enhance **defense-in-depth** by starting security at the **code level** and validating it in the **final release stage**.
  - Reduce **dependency on external consultants**, allowing DevOps teams to take ownership of security.
  - Minimize last-minute security fixes and prevent costly rework by **embedding security early**.
  - Improve **application security posture** without disrupting development timelines.

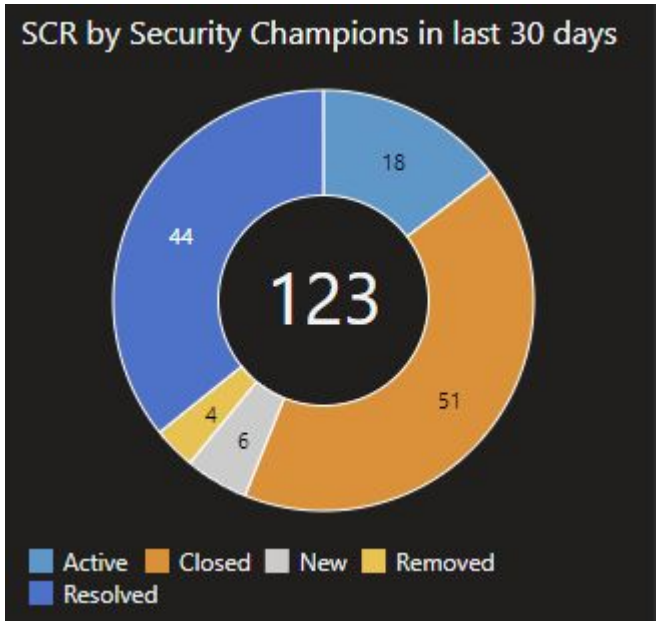
Gaining Knowledge

- Security Champions need access to the right resources to grow their expertise. Key learning paths include:
  - **Hands-on Labs & Training Platforms:** Platforms like **OWASP Security Knowledge Framework (SKF)** and **SecureFlag** provide interactive training modules.
  - **Knowledge Sharing:** Security Engineers and Senior Champions host **webinars, guild meetings, and discussions**to ensure Champions can ask questions and refine their approach.
  - **Peer Collaboration:** A **mentorship model** where Champions can seek guidance from experienced security professionals within the organization.

Way to monitor and WoW

- Different methods can be used to track reviews of SAST or SCA or other things that are being done. Most simple way of tracking: every review needs to be mentioned in azure ticket on a Security Champion Azure board. Statistics can be tracked like this:





Workflow (WoW)

- Tracking security reviews ensures **accountability and continuous improvement**. A structured workflow includes:

1. **Request Submission:** Developers raise a security review request via a **ticketing system** (e.g., Azure Boards, Jira, or equivalent).
2. **Analysis & Review:** A Security Champion evaluates the request, conducts a scan, and **documents findings**.
3. **Approval & Next Steps:** Security Champions provide feedback, remediation guidance, or approval for security-related changes.
4. **Audit Readiness:** All security review steps are **logged in the ticketing system**, creating an **audit-friendly** record without relying on scattered email threads.

- This structured approach **streamlines security testing**, reduces overhead, and ensures compliance with **security best practices**.