De Código a Release

Seguridad en Cada Etapa del SDLC

Carlos Oliva // Security Architect @ ZeroFox // carlos.oliva@gmail.com

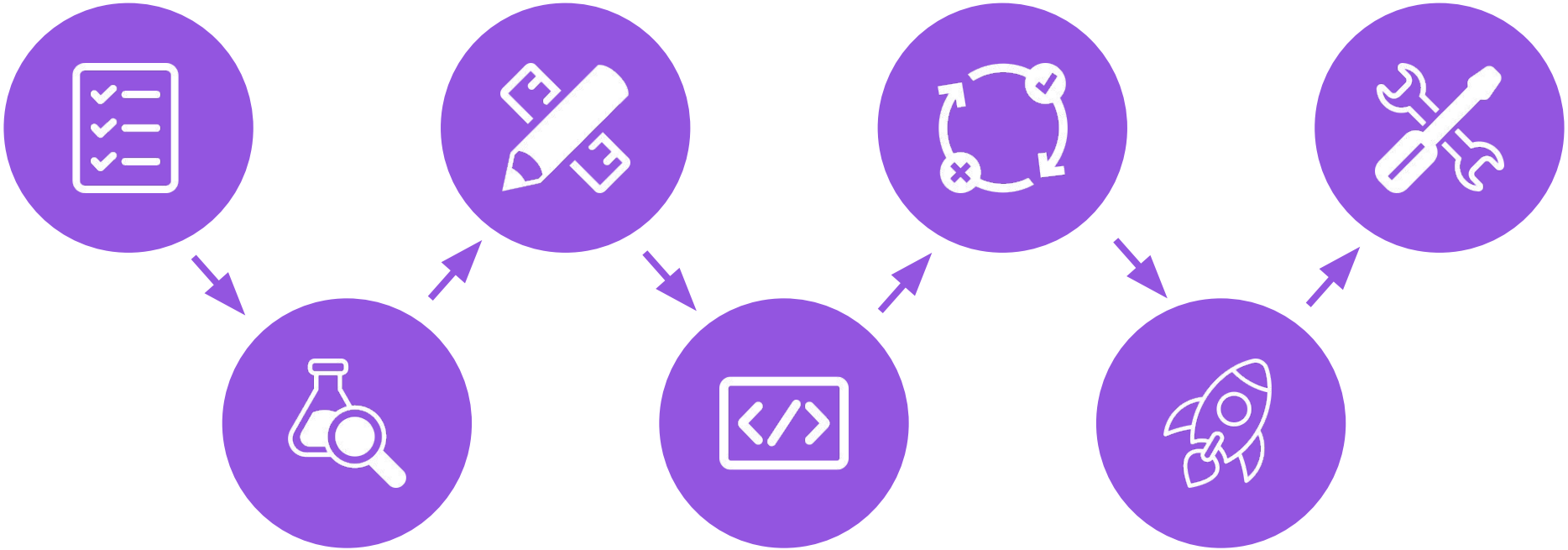
Agenda

- Intro: Hacia un SDLC Seguro
- Vulnerabilidades comunes en el SDLC
- Gobernanza sólida de seguridad en el SDLC
- Shift Left Security y DevSecOps
- Modelado de Amenazas
- Cómo lograr un SDLC Seguro

Agenda

- SLS: Planificación, Análisis y Diseño
- DevSecOps: Codificación, Testing y Despliegue
- DevSecOps: Mantenimiento
- Protocolo de Respuesta a Incidentes
- Metodologías populares para SSDLC
- Referencias y Recursos

Hacia un SDLC Seguro



Vulnerabilidades comunes en el SDLC

- Herramientas DevOps e Infraestructura
  - Configuraciones por defecto priorizan eficiencia y velocidad, no seguridad
- Manipulación de Código
  - Alteración no autorizada del código e inyección de código malicioso
- Prácticas inseguras de codificación
  - Exposición de credenciales, falta de validación de entradas, OWASP
- Filtraciones de código y datos sensibles
  - Publicación accidental o maliciosa de código privado, divulgación de datos de usuarios

Gobernanza de Seguridad en el SDLC

- Visibilidad
  - Seguridad debe conocer y tener acceso a herramientas, personal y procesos del SDLC
- Políticas de Privilegios Mínimos
  - Desarrollo y Operaciones deben contar con el acceso mínimo necesario para sus funciones
- Autenticación
  - MFA / SSO
- Protección de ramas y reglas de compilación
  - Revisión de código, firmado de commits, prohibición de forzar pushes, prohibición de usar repositorios personales, búsqueda de credenciales y secretos codificados, pruebas unitarias, etc.
- Supervisar cambios a controles establecidos
  - Solicitud de cambios sujetos a aprobación de Seguridad
  - Alertas ante cambios en el ambiente que no han sido autorizados

Shift Left Security y DevSecOps

- Integrar la seguridad en cada etapa del SDLC
- Alcances distintos y complementarios
- Shift Left Security:
  - Detectar y corregir problemas de seguridad en las etapas tempranas del SDLC
  - Reducir el costo y esfuerzo para solucionarlos más adelante
- DevSecOps:
  - Hacer de la seguridad una responsabilidad compartida durante todo el SDLC
  - Involucra desarrolladores, operaciones y seguridad

Modelado de Amenazas

- ¿Qué estamos construyendo?
- ¿Qué puede salir mal?
- ¿Qué haremos al respecto?
- ¿Lo hicimos bien?
- Metodología STRIDE para identificar violaciones de seguridad:
    - Spoofing, Tampering, Repudiation, Information Disclosure, Denial of Service, Elevation of Privilege

Cómo lograr un SDLC Seguro

- ¿Qué problemas resuelve un SDLC Seguro?
- Adoptar una mentalidad Shift Left Security y DevSecOps
- Involucrar a expertos en seguridad en las fases tempranas del SDLC
- Sacar provecho del modelado de amenazas
- Capacitar a desarrolladores en buenas prácticas de codificación segura

Cómo lograr un SDLC Seguro

- Implementar revisión de código
- Monitorear el software y detectar vulnerabilidades una vez desplegado
- Actualizar el SDLC en la medida de que surjan nuevas amenazas
- Utilizar herramientas y tecnologías para identificar y mitigar riesgos
- Contar con un protocolo de respuesta a incidentes

SLS: Planificación, Análisis y Diseño

- Planificación
  - Involucrar al equipo de seguridad durante las primeras etapas del proceso
  - Presentar especificaciones y lineamientos de seguridad al equipo de desarrollo
- Análisis de Requerimientos
  - Involucrar al equipo de desarrollo durante la priorización de tareas
  - Evaluar el impacto de seguridad de requerimientos y componentes
- Diseño
  - Contar con el equipo de seguridad en el diseño de la arquitectura de software
  - Identificar sistemas involucrados, servicios a crear, interacción de usuarios

DevSecOps: Codificación, Testing y Despliegue

- Codificación
  - Implementación de SAST en el pipeline (IDE, git, etc.)
  - Revisión (y aprobación) de código
  - SCA: Seguridad en OSS
- Testing y Despliegue
  - Implementación de DAST
  - Búsqueda y análisis de vulnerabilidades en contenedores
  - Ambientes diferenciados para desarrollo, pruebas, staging y producción

DevSecOps: Mantenimiento

- Monitoreo
  - Herramientas de telemetría
  - WAF
- Reportería
  - Análisis de logs en tiempo real
- Alertas
  - Generación de alertas relevantes y oportunas
- Bug Bounty
  - Externaliza la búsqueda de vulnerabilidades
  - Divulgación responsable

Protocolo de Respuesta a Incidentes

- Equipos de respuesta especializados
  - Sistema de guardia (on-call)
  - Rotaciones y alternativas
- Invocar a todos los actores involucrados
  - Respuesta a incidentes debe ser un esfuerzo colaborativo entre equipos
- Identificar severidad del incidente
  - Superficie de ataque y exposición
  - Establecer prioridad (P1, P2, etc)
  - CVSS (https://www.first.org/cvss/)

Protocolo de Respuesta a Incidentes

- Identificar causa y soluciones
  - Solución inmediata y a mediano/largo plazo
- Documentar y comunicar
  - Template de documento de incidente
  - Comprometer tareas y asignar responsabilidades
  - Comunicar responsablemente a partes afectadas
- Agendar reuniones de equipos y revisión posterior
- Post Mortem
  - Qué salió bien, qué salió mal, en qué tuvimos suerte
  - Aprendizajes

Metodologías populares para SSDLC

- Microsoft Security Development Lifecycle (SDL)
  - https://www.microsoft.com/en-us/securityengineering/sdl
- OWASP Software Assurance Maturity Model (SAMM)
  - https://owasp.org/www-project-samm/
- NIST Cybersecurity Framework (NIST CSF)
  - https://www.nist.gov/cyberframework
- Google Supply-chain Levels for Software Artifacts (SLSA)
  - https://cloud.google.com/blog/products/application-development/google-introduces-slsa-framework
- ISO/IEC 27001
  - https://www.iso.org/standard/27001

Referencias y Recursos

- OWASP Top 10
  - https://owasp.org/www-project-top-ten/
- Calculadora CVSS
  - https://www.first.org/cvss/calculator/4.0
- Secure Developer Podcast
  - https://www.devseccon.com/the-secure-developer-podcast/
- LiveOverflow en YouTube
  - https://www.youtube.com/@LiveOverflow

De Código a Release

Seguridad en Cada Etapa del SDLC

Carlos Oliva // Security Architect @ ZeroFox // carlos.oliva@gmail.com