ALLIANCE AUTOMOTIVE GROUP

A GPC COMPANY

GPC®

How to DevSecOps

26-October-2022
Christian Kluenter

GPC Cybersecurity

Speaker

Current roles:

- Regional Cyber Defense Manager Europe
    - Responsible for all technical security areas in the region
    - Leading of Engineers & Analysts
    - Responsible for Incident Response in the region
    - Specialized of Application Security
    - Created & Designed the global AppSec program with his partners
- Former:
    - Data Protection Officer (GDPR)
    - Penetration Tester
    - Information Security Lead / Officer

Hobbies & Private:

- Married, 1 kid, 2 Dogs
- Collecting books, Studying new stuff, personal growth, walking in the woods.

Agenda

1. DevSecOps and Application Security
2. The Journey
    1. Survey and overview
    2. Understanding Obligations, goals and expectations
    3. Setting common goals
    4. Embedding and becoming a Partner
    5. Grow together
3. There is no end of this journey

DevSecOps & AppSec

- What are your most important aspects of Team-Culture?
- Why are you trying to implement DevSecOps?
- What do you wish from DevSecOps culture?

DevSecOps & AppSec

DevSecOps & AppSec

- Streamline workflows
- Continuous improvement
- Holistic perspective on the product lifecycle
- Addressing & Scoping the right requirements and obligations
- Build secure software to secure the Business and your customers
- Higher values for the Business and customers

DevSecOps & AppSec

Paradigm:

- Collaboration and sharing of knowledge and expertise between **Dev**elopers, **Sec**urity and **Op**eration**s**

The Idea:

- interdisciplinary teams can deliver better, more secure and faster
- Security built in before deployment

← Security is **shifting left,** to the start

DevSecOps & AppSec

Promises:

- Knowledge Transfer in all directions, even management
- Better insights and views on risks
- Holistic view on IT
- Repeatable results
- Streamlined workflows with reduced lead times
- Fewer outages and issues
- Shorter problem solving times
- Continuous improvements

DevSecOps & AppSec

Application Security is a **full security discipline**, including:

- Secure Coding **practices**
- Secure Coding environments
- Design **principles**
- Architecture **reviews**
- Supply Chain Security
- Security **Testing** (e.g. Pentests, SAST, DAST, Fuzzing)
- Licenses and **Governance**
- CI/CD **Pipelines**

- **Build - Container Security** / Kubernetes (K8s)
- Bug **Handling** (e.g. Security Issues)
- Risk **Management**
- Data Classification / Data Management
- **Lifecycle Management**
- Contracts / SLAs
- Documentation
- Process Definition

The Journey

Survey and Overview

- Understand the Business (goals)
  - Products
  - Strategy
  - Processes
  - Key Stakeholder

- Understand where you are
  - Do we have trained Staff?
  - Do we already have a need / appetite to implement DevSecOps?
  - How are we currently providing value?

Obligations, goals and expectations

- Are we currently fulfilling our Obligations?
  - Laws, Executive Orders, legal regulations?
  - Internal & external regulations / requirements
  - Internal & external policies and standards?

- Are we achieving our goals?
  - Are we delivering in time?
  - Are we matching the expected costs / budgets?
  - Are we Implementing / deploying what we wanted to deploy?

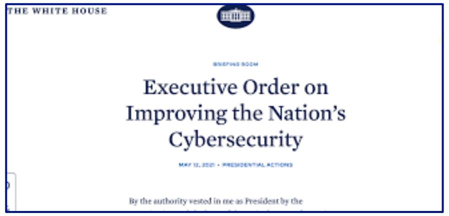
- Are we matching the expectations (internal / external)?
  - Are we delivering what we were expected to deliver?
  - Are we in control of what we delivered?
  - Is it matching the expectations of our customers?

PCI DSS COMPLIANT

TÜV NORD
TÜV NORD CERT GmbH
IEC 62443
Industrial Security

EU Cyber Resilience Act
For safer & more secure digital products
#DigitalEU #CyberSecEU

GDPR


THE WHITE HOUSE

BRIEFING ROOM

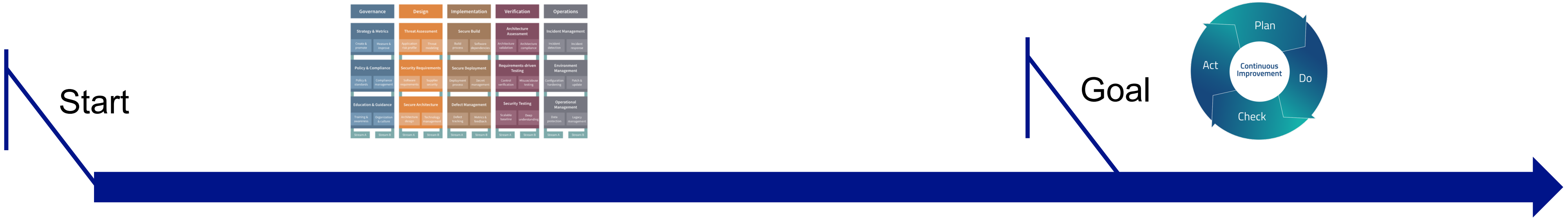
Executive Order on Improving the Nation's Cybersecurity

MAY 12, 2021 • PRESIDENTIAL ACTIONS

By the authority vested in me as President by the

Setting common goals

1. Analyze the gaps between your current situation, problems and your goals
2. Find a Partner supporting you & get leadership buy-in
3. Define what is needed to reach the goals, with your partners
   - Automation, budget, resources procedures, processes
4. Define a framework that supports you on reaching your goals
5. Build out a roadmap
   1. Start small (quick wins)
      - e. g. standardized Dev-Environments
      - Automated Ops-Deployments
      - Embedding Security and Operations in the Dev-Teams processes
6. Celebrate achievement and failures
   1. Learn from failures, grow from failures



Governance	Design	Implementation	Verification	Operations
Strategy & Metrics	Threat Assessment	Secure Build	Architecture Assessment	Incident Management
Create & promote Measure & improve	Application risk profile Threat modeling	Build process Software dependencies	Architecture validation Architecture compliance	Incident detection Incident response
Policy & Compliance	Security Requirements	Secure Deployment	Requirements-driven Testing	Environment Management
Policy & standards Compliance management	Software requirements Supplier security	Deployment process Secret management	Control verification Misuse/abuse testing	Configuration hardening Patch & update
Education & Guidance	Secure Architecture	Defect Management	Security Testing	Operational Management
Training & awareness Organization & culture	Architecture design Technology management	Defect tracking Metrics & feedback	Scalable baseline Deep understanding	Data protection Legacy management
Stream A Stream B	Stream A Stream B	Stream A Stream B	Stream A Stream B	Stream A Stream B

Embedding and Becoming a Partner

Get into the Teams!

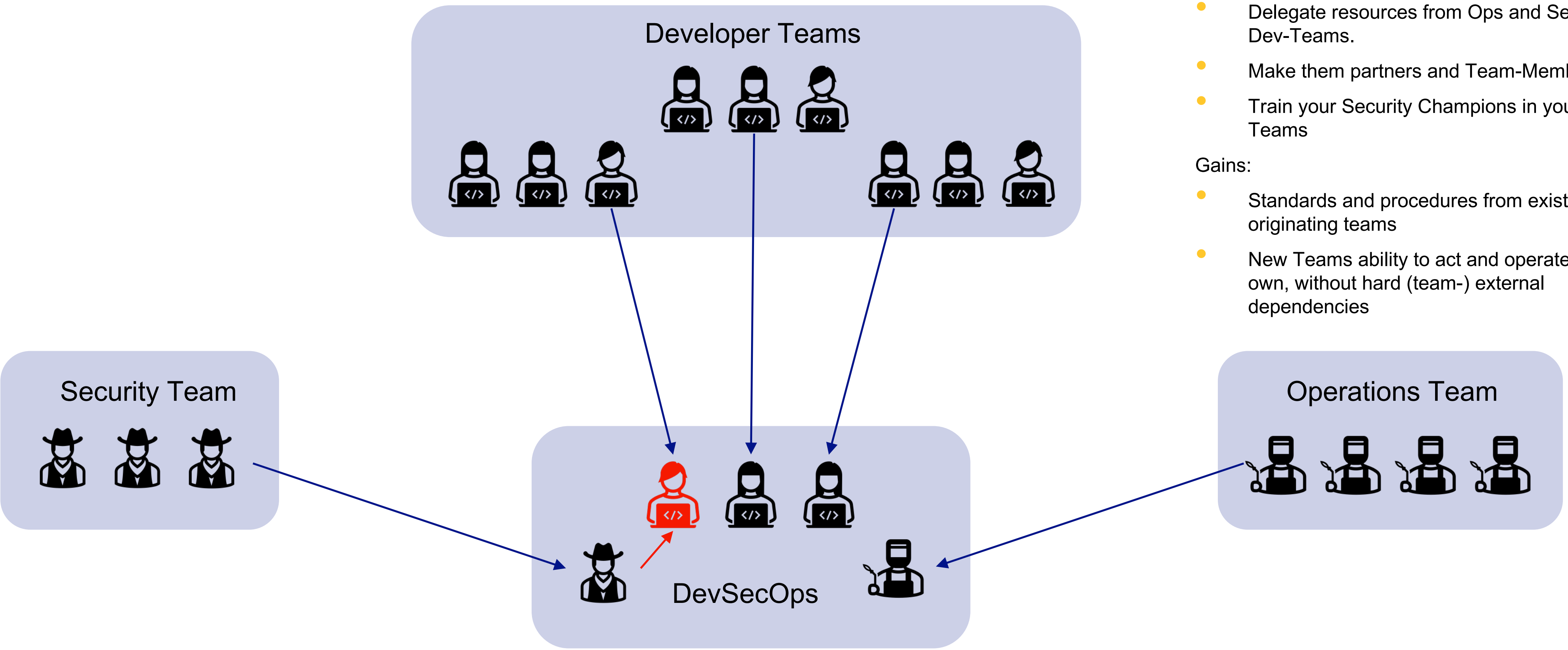
- Embed/leverage Operations and Security in your Dev-Dailies, retrospectives and/or planning meetings
- Support each other directly from the beginning
- Be open on what you are working on, what is planned and where you have struggles
- Managers: Be / get transparent on workload, projects and tasks
- Make operational work visible
- Find solutions together

Always start small!

- Start with loosely coupled Teams and Architectures
- Train and learn from each other – Training Programs -> Security Champions Program
- Define what can be supported/operated and secured – e.g. with Standards and guidelines
- Define a layered approach based on applications risks and maturity – Risk Based Approach

Embedding and Becoming a Partner

How could this look like?



Keep:

- Organizational structure
- Team structure
- Leadership

Change:

- Delegate resources from Ops and Sec into Dev-Teams.
- Make them partners and Team-Members
- Train your Security Champions in your Dev-Teams

Gains:

- Standards and procedures from existing / originating teams
- New Teams ability to act and operate on their own, without hard (team-) external dependencies

Grow together

Grow with every improvements and failures

- Learn from mistakes
- Build a culture of openness
- Do not judge or blame
- Use failures for improvements

Grow together as a team

- See your team as a football(Soccer) club (Offense, Defense, Goalkeeper)
- No one can win without the other
- All are better if they understand how to / and support each other
- Celebrate failures and mistakes as an opportunity to improve
- Managers: Bring teams into a position where they can fix their dependencies within the Team (Dev+Sec+Ops = Champions League)

There is no end of the Journey

Have you said end?

This is just a start.

- Preparation is key
- Know where you are & want to go
- Know what is needed from the business
- Understand the maturity of the teams to understand where to start
    - Not everything is a nail!

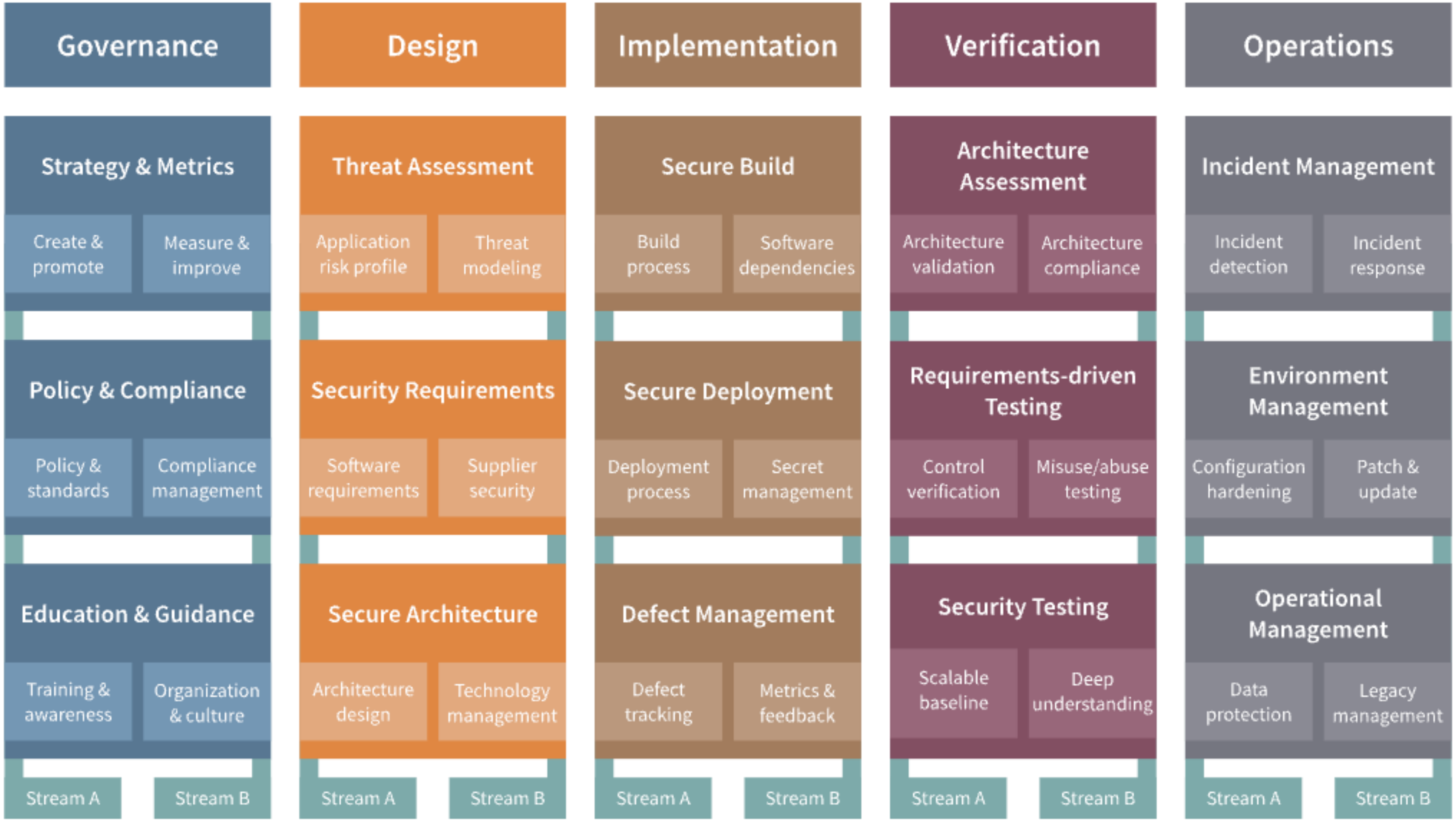
- Training and continuous learning is necessary
- It is not all about automation, it is about achieving goals
- Think always about one of the most important security principles:

Good enough Security

Security Review

Q & A

Appendix: OWASP SAMM V2.0



Governance	Design	Implementation	Verification	Operations
Strategy & Metrics	Threat Assessment	Secure Build	Architecture Assessment	Incident Management
Create & promote	Application risk profile	Build process	Architecture validation	Incident detection
Measure & improve	Threat modeling	Software dependencies	Architecture compliance	Incident response
Policy & Compliance	Security Requirements	Secure Deployment	Requirements-driven Testing	Environment Management
Policy & standards	Software requirements	Deployment process	Control verification	Configuration hardening
Compliance management	Supplier security	Secret management	Misuse/abuse testing	Patch & update
Education & Guidance	Secure Architecture	Defect Management	Security Testing	Operational Management
Training & awareness	Architecture design	Defect tracking	Scalable baseline	Data protection
Organization & culture	Technology management	Metrics & feedback	Deep understanding	Legacy management
Stream A Stream B	Stream A Stream B	Stream A Stream B	Stream A Stream B	Stream A Stream B