Kubernetes: Трансформация к SecDevSecOpsSec

Дмитрий Евдокимов

DevOps Conf 2021

Whoami

- Технический директор, исследователь ИБ
- Организатор конференций ZeroNights, DEFCON Russia (#7812)
- В прошлом редактор рубрик в журнале “ХАКЕР”
- Мейнтейнер проекта “Python Arsenal for Reverse Engineering”
- Автор Telegram-канала “k8s (in)security”
- Автор тренинга “Безопасность облачных приложений в Kubernetes”
- Спикер: BlackHat, HITB, ZeroNights, HackInParis, Confidence, SAS, PHDays и др.

Disclaimer

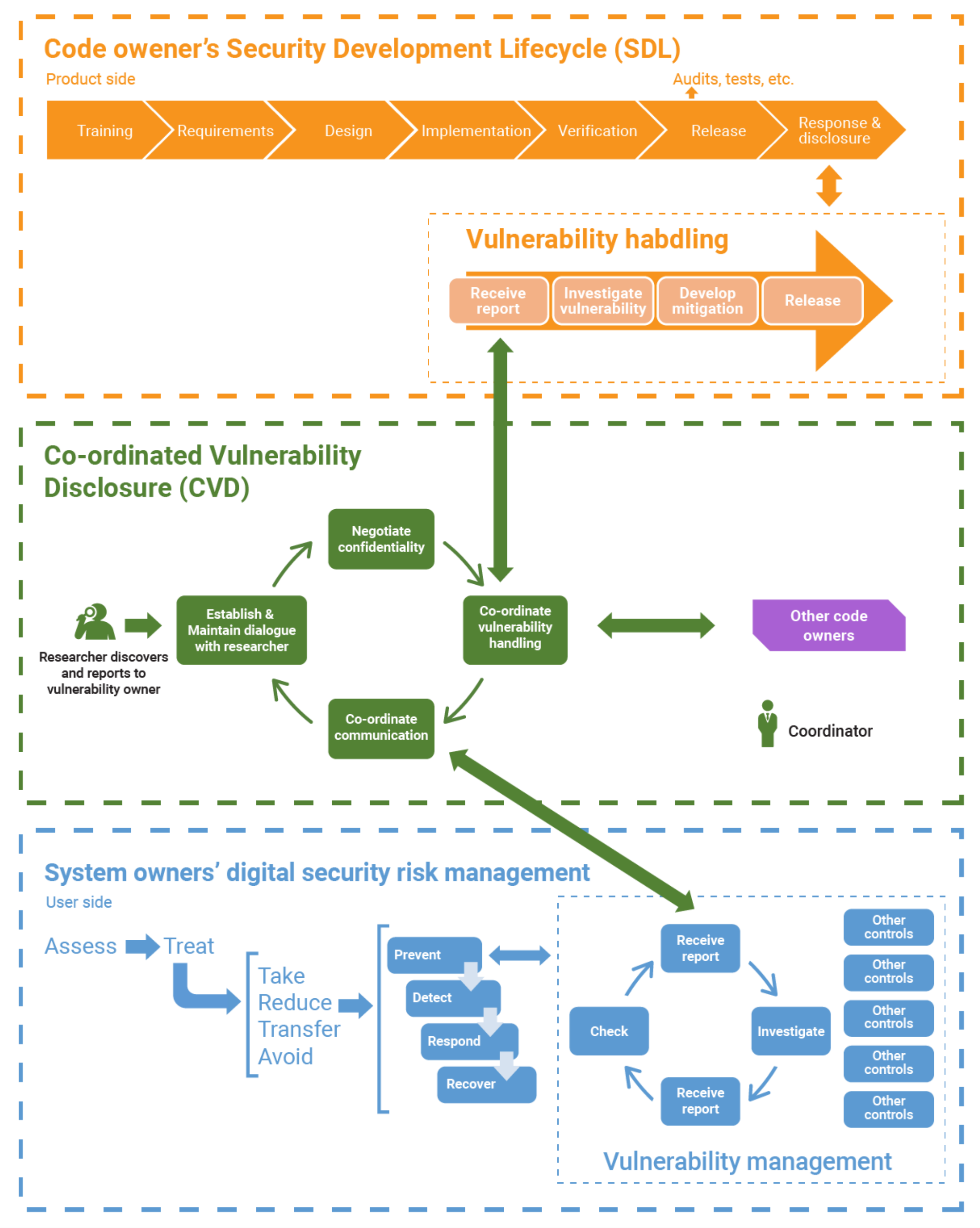
Данный доклад – это взгляд специалиста по информационной безопасности, который большую часть своей карьеры занимался наступательной безопасностью, о том, как Kubernetes способен повысить уровень безопасности, упростить управление безопасностью и ускорить внедрение механизмов безопасности.

Agenda

- DevSecOps, SecDevOps, DevOpsSec, SSDL, SecDevSecOpsSec – WTF?!
- Пару слов о Kubernetes
- Управление рисками и угрозами в Kubernetes
- Заключение

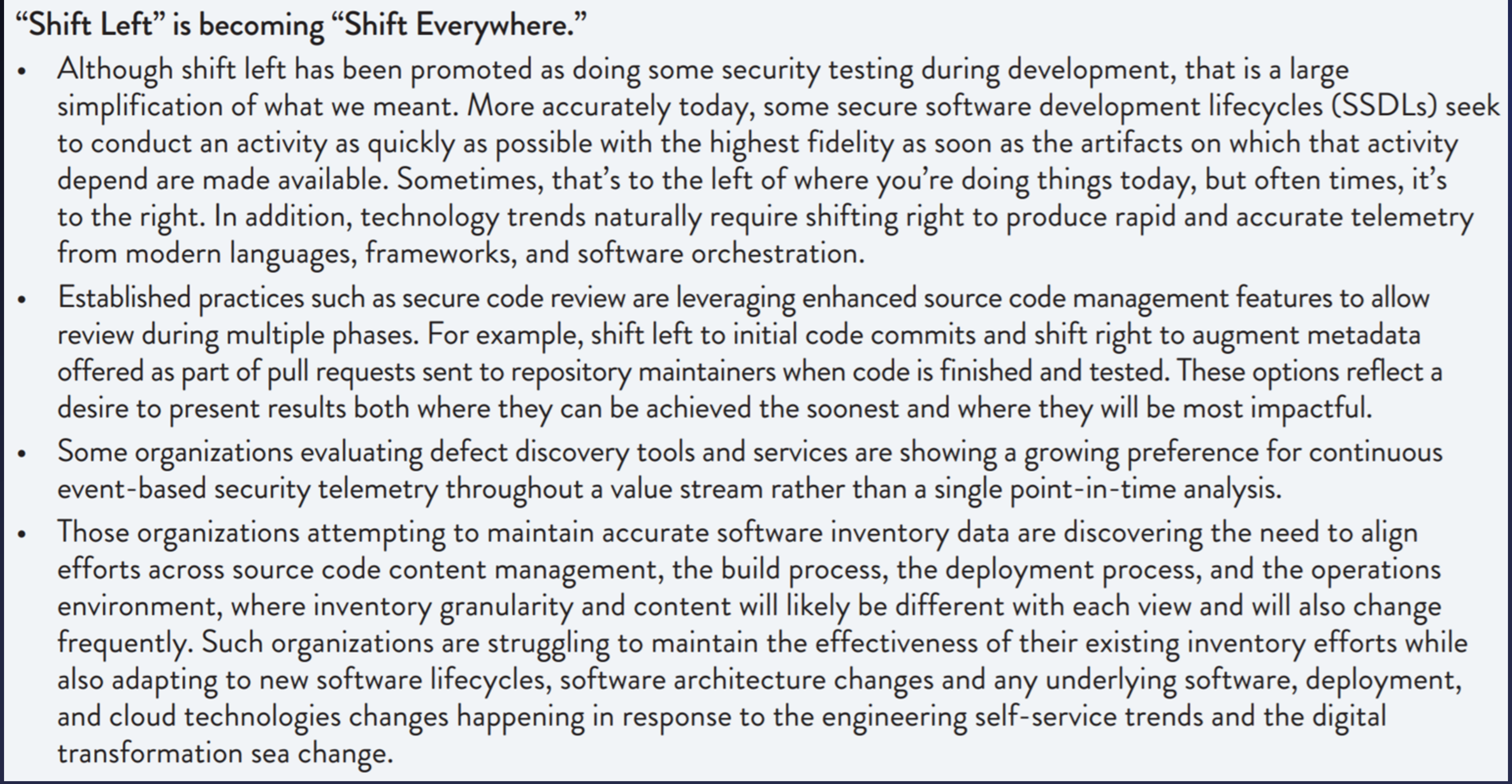
DevSecOps, SecDevOps, DevOpsSec, SSDL, SecDevSecOpsSec – WTF?!

Security > SSDL > DevSecOps



Source: OECD

Shift ~~Left~~ Everywhere Security



"Shift Left" is becoming "Shift Everywhere."

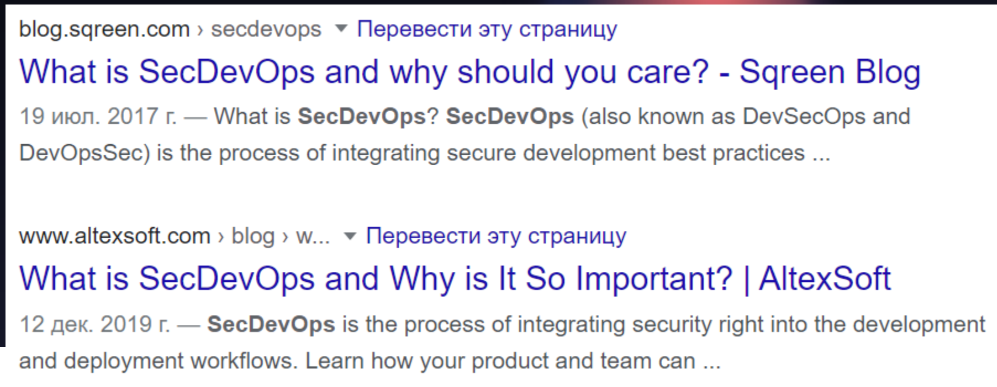
- Although shift left has been promoted as doing some security testing during development, that is a large simplification of what we meant. More accurately today, some secure software development lifecycles (SSDLs) seek to conduct an activity as quickly as possible with the highest fidelity as soon as the artifacts on which that activity depend are made available. Sometimes, that's to the left of where you're doing things today, but often times, it's to the right. In addition, technology trends naturally require shifting right to produce rapid and accurate telemetry from modern languages, frameworks, and software orchestration.
- Established practices such as secure code review are leveraging enhanced source code management features to allow review during multiple phases. For example, shift left to initial code commits and shift right to augment metadata offered as part of pull requests sent to repository maintainers when code is finished and tested. These options reflect a desire to present results both where they can be achieved the soonest and where they will be most impactful.
- Some organizations evaluating defect discovery tools and services are showing a growing preference for continuous event-based security telemetry throughout a value stream rather than a single point-in-time analysis.
- Those organizations attempting to maintain accurate software inventory data are discovering the need to align efforts across source code content management, the build process, the deployment process, and the operations environment, where inventory granularity and content will likely be different with each view and will also change frequently. Such organizations are struggling to maintain the effectiveness of their existing inventory efforts while also adapting to new software lifecycles, software architecture changes and any underlying software, deployment, and cloud technologies changes happening in response to the engineering self-service trends and the digital transformation sea change.

Building Security In Maturity Model (BSIMM) 11

NEW: SecDevSecOpsSec

Маркетинг:

- SecDevOps
- DevSecOps
- DevOpsSec
- ...



blog.sqreen.com › secdevops ▾ Перевести эту страницу

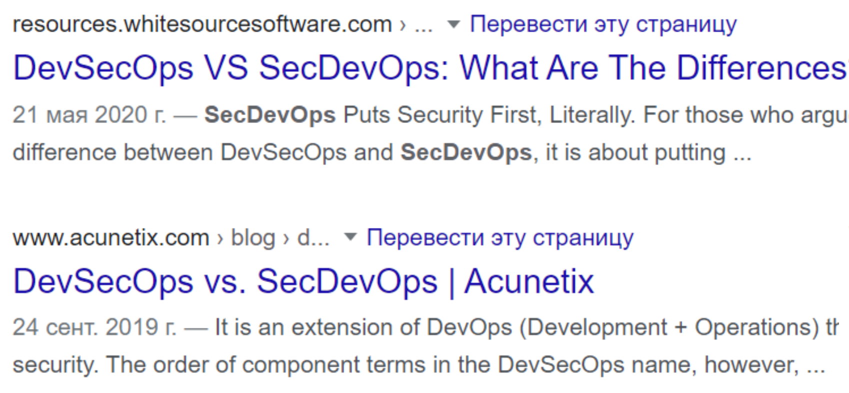
What is SecDevOps and why should you care? - Sqreen Blog

19 июл. 2017 г. — What is **SecDevOps**? **SecDevOps** (also known as DevSecOps and DevOpsSec) is the process of integrating secure development best practices ...

www.altexsoft.com › blog › w... ▾ Перевести эту страницу

What is SecDevOps and Why is It So Important? | AltexSoft

12 дек. 2019 г. — **SecDevOps** is the process of integrating security right into the development and deployment workflows. Learn how your product and team can ...



resources.whitesourcesoftware.com › ... ▾ Перевести эту страницу

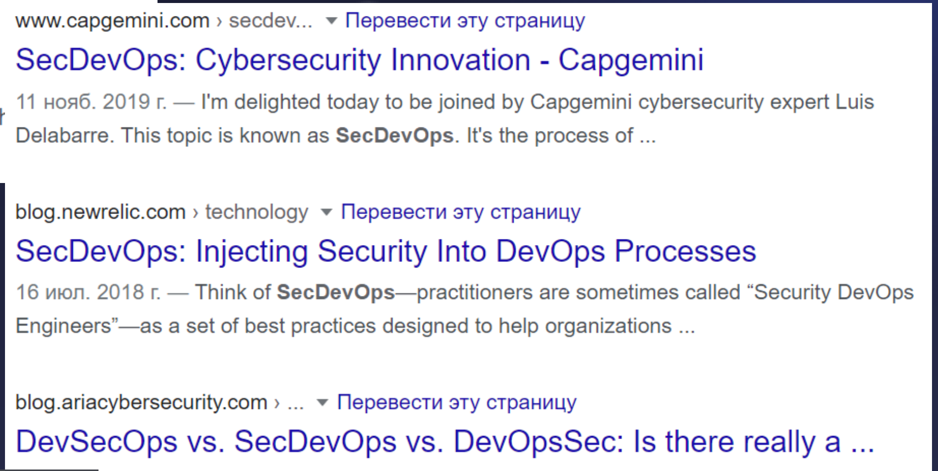
DevSecOps VS SecDevOps: What Are The Differences?

21 мая 2020 г. — **SecDevOps** Puts Security First, Literally. For those who argue there is a difference between DevSecOps and **SecDevOps**, it is about putting ...

www.acunetix.com › blog › d... ▾ Перевести эту страницу

DevSecOps vs. SecDevOps | Acunetix

24 сент. 2019 г. — It is an extension of DevOps (Development + Operations) th security. The order of component terms in the DevSecOps name, however, ...



www.capgemini.com › secdev... ▾ Перевести эту страницу

SecDevOps: Cybersecurity Innovation - Capgemini

11 нояб. 2019 г. — I'm delighted today to be joined by Capgemini cybersecurity expert Luis Delabarre. This topic is known as **SecDevOps**. It's the process of ...

blog.newrelic.com › technology ▾ Перевести эту страницу

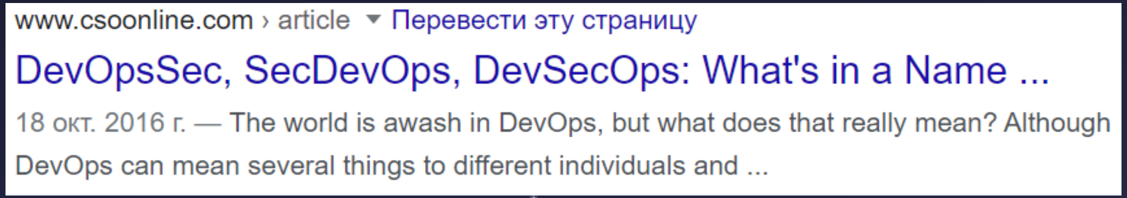
SecDevOps: Injecting Security Into DevOps Processes

16 июл. 2018 г. — Think of **SecDevOps**—practitioners are sometimes called "Security DevOps Engineers"—as a set of best practices designed to help organizations ...

blog.ariacybersecurity.com › ... ▾ Перевести эту страницу

DevSecOps vs. SecDevOps vs. DevOpsSec: Is there really a ...

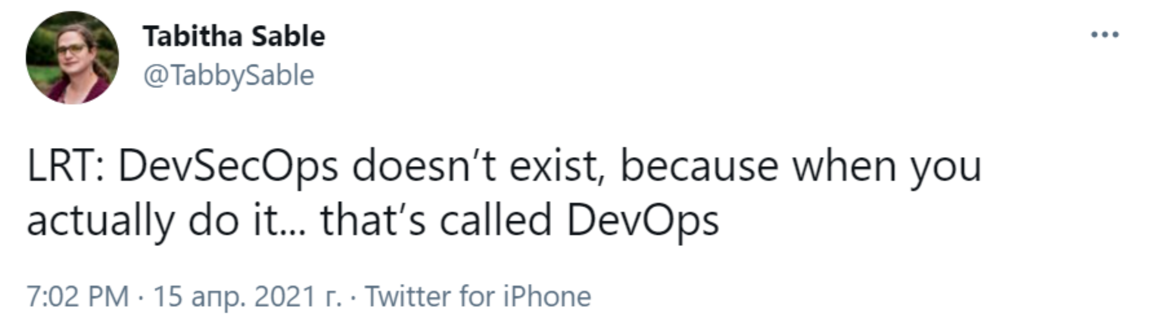
2018 г. — **SecDevOps**: To borrow from "Goldilocks and the Three Bears," this approach ht! According to this insightful article on CSO.com, ...



www.csoonline.com › article ▾ Перевести эту страницу

DevOpsSec, SecDevOps, DevSecOps: What's in a Name ...

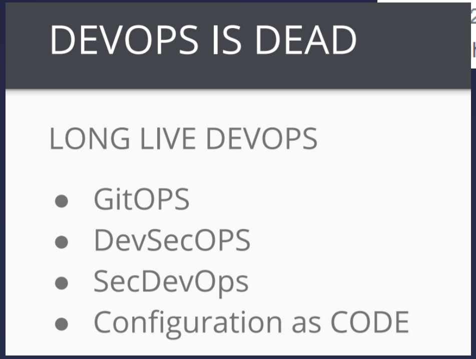
18 окт. 2016 г. — The world is awash in DevOps, but what does that really mean? Although DevOps can mean several things to different individuals and ...



Tabitha Sable
@TabbySable

LRT: DevSecOps doesn't exist, because when you actually do it... that's called DevOps

7:02 PM · 15 апр. 2021 г. · Twitter for iPhone



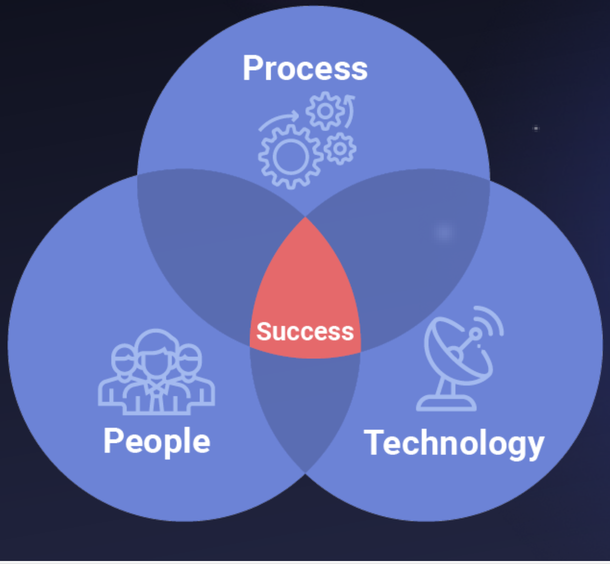
DEVOPS IS DEAD

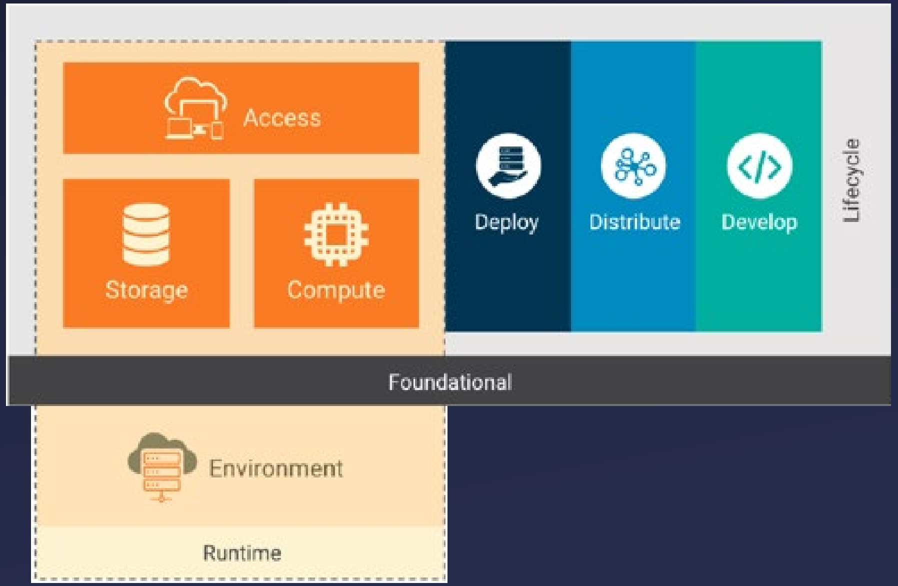
LONG LIVE DEVOPS

- GitOPS
- DevSecOPS
- SecDevOps
- Configuration as CODE

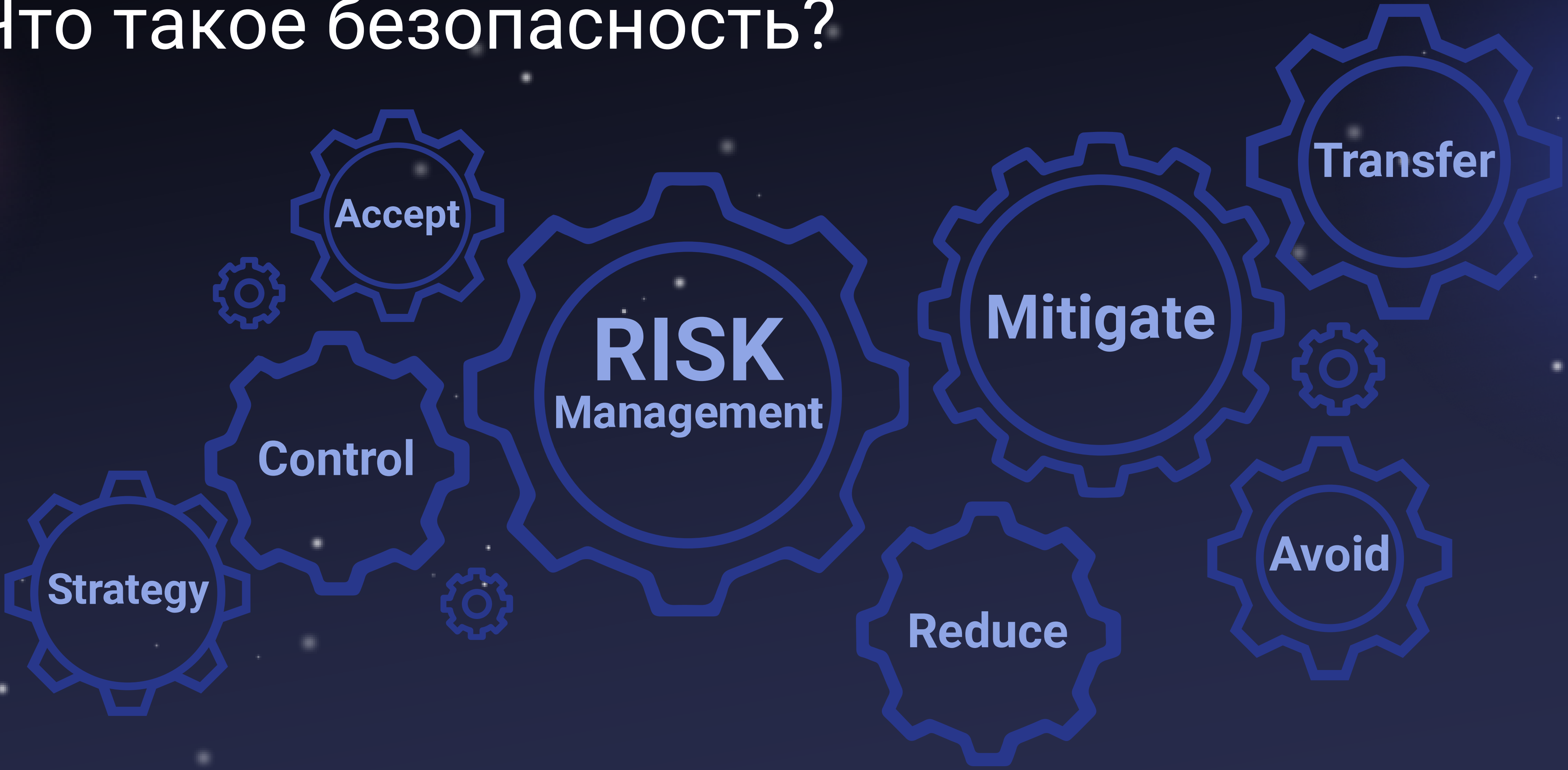
Cloud Native security

- People: Совместная работа всех департаментов
- Process: Задействование всего lifecycle приложений
- Technology: Высокий уровень безопасности без вреда для скорость доставки нового value





Что такое безопасность?



Пару слов о Kubernetes

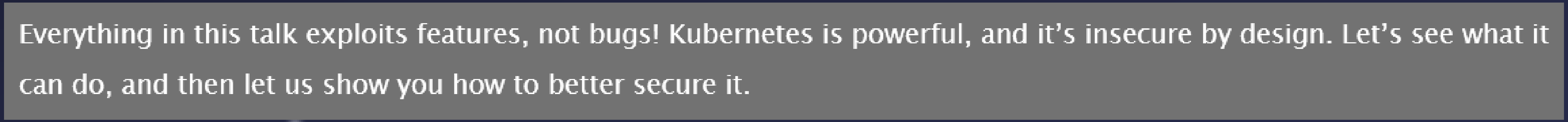
K8s is insecure by default

Kubernetes – это фреймворк

- Каждая инсталляция уникальна
- Реализация части механизмов безопасности лежит на других компонентах

Все настройки по умолчанию призваны ускорить ваш старт в нем

- Почти все механизмы и функции безопасности деактивированы



Everything in this talk exploits features, not bugs! Kubernetes is powerful, and it’s insecure by design. Let’s see what it can do, and then let us show you how to better secure it.

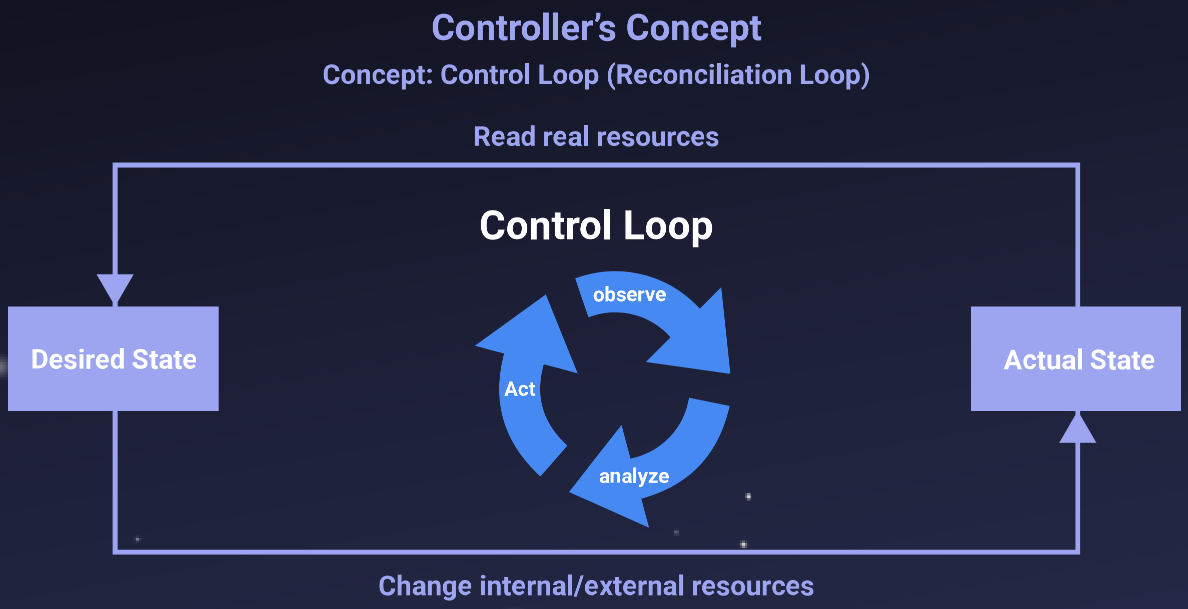
“The Path Less Traveled: Abusing Kubernetes Defaults”, Ian Coldwater, Duffie Cooley, BlackHat USA 2019

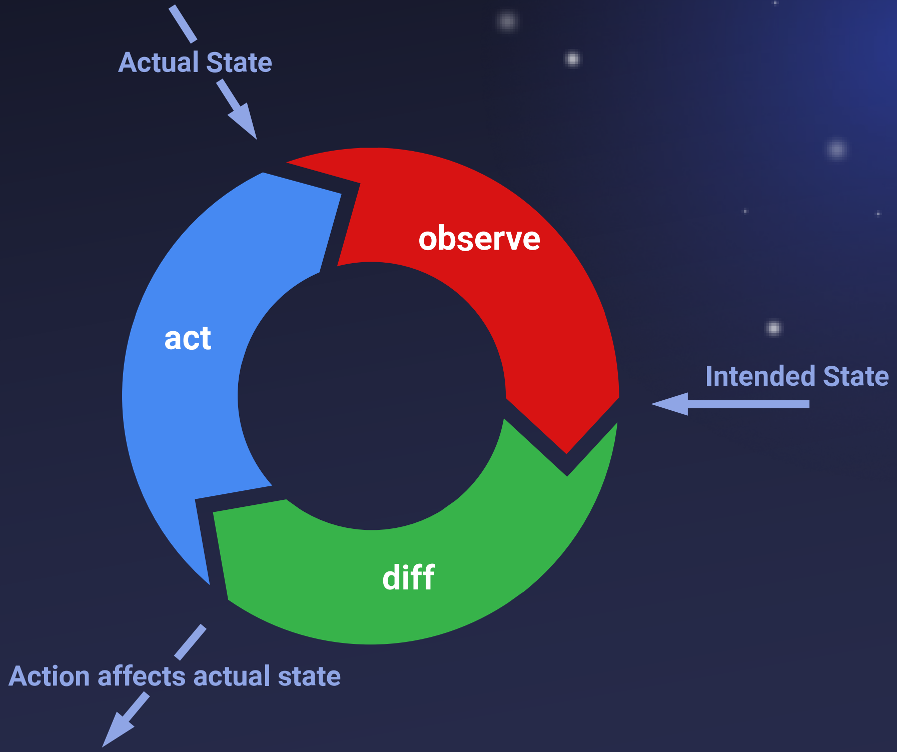
Декларативная природа K8s

Все есть YAML

- Self-documenting
- Концентрация на результате, а не на процессе
- Предсказуемость результата
- Удобно хранить, менять и отслеживать изменения
- Something as Code
  - Infrastructure as Code
  - Security as Code
  - …
- Может обрабатывать машина
  - Меньше вероятность ошибки
  - Можно автоматизировано проверить статическим анализом

Control loop

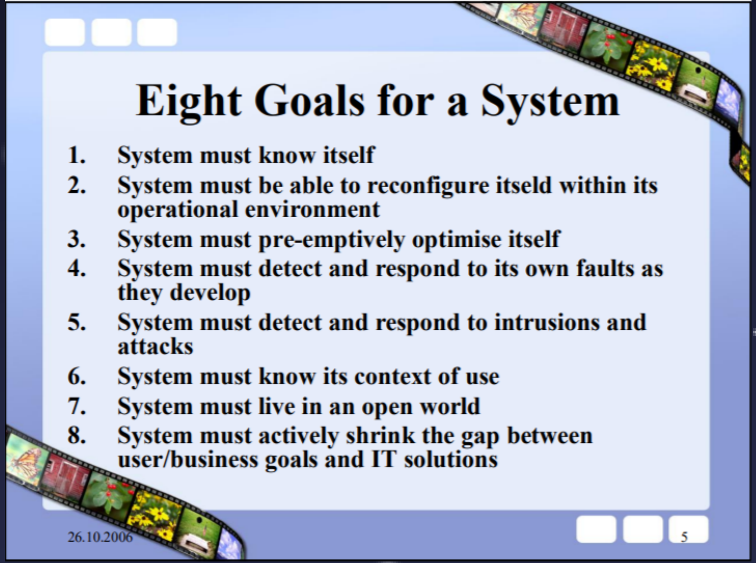




Self-control system

"Self-healing systems: what are they?", Tiina Niklander, AMICT'2006

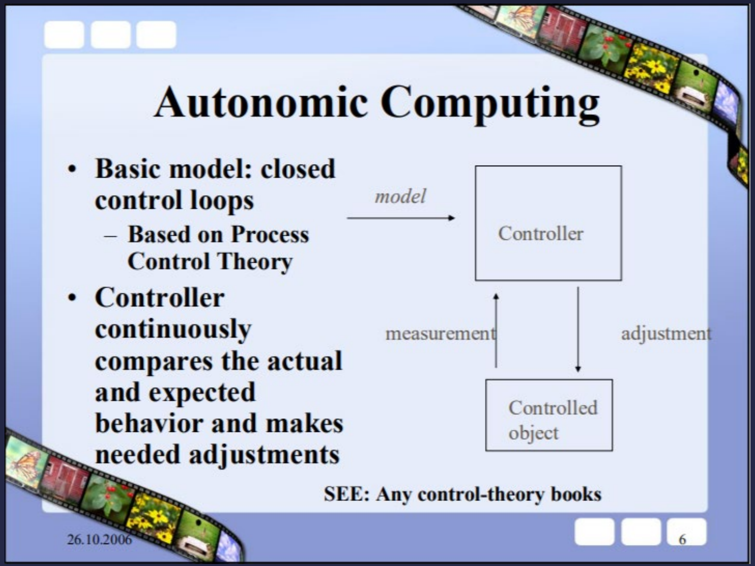
“Large-scale cluster management at Google with Borg”, Google, EuroSys 2015



Eight Goals for a System

1. System must know itself
2. System must be able to reconfigure itseld within its operational environment
3. System must pre-emptively optimise itself
4. System must detect and respond to its own faults as they develop
5. System must detect and respond to intrusions and attacks
6. System must know its context of use
7. System must live in an open world
8. System must actively shrink the gap between user/business goals and IT solutions

26.10.2006 5



Autonomic Computing

- Basic model: closed control loops
  - Based on Process Control Theory
- Controller continuously compares the actual and expected behavior and makes needed adjustments

model

Controller

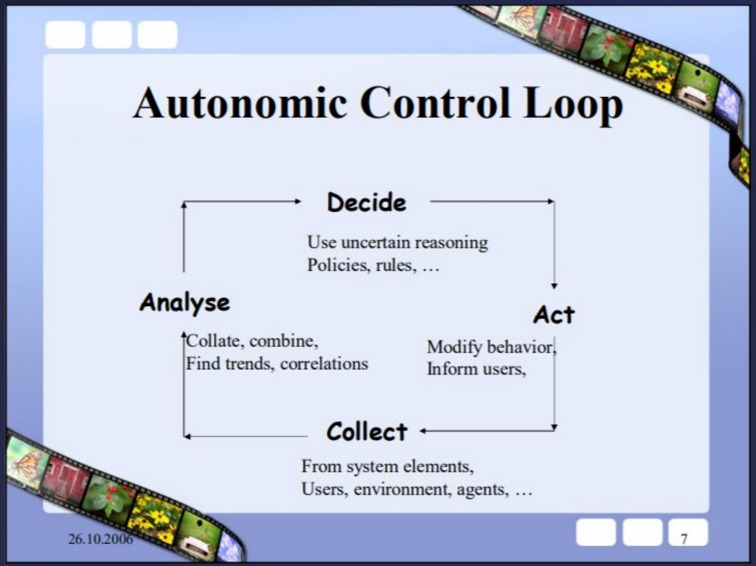
measurement

adjustment

Controlled object

SEE: Any control-theory books

26.10.2006 6



Autonomic Control Loop

Decide

Use uncertain reasoning
Policies, rules, …

Analyse

Collate, combine,
Find trends, correlations

Act

Modify behavior,
Inform users,

Collect

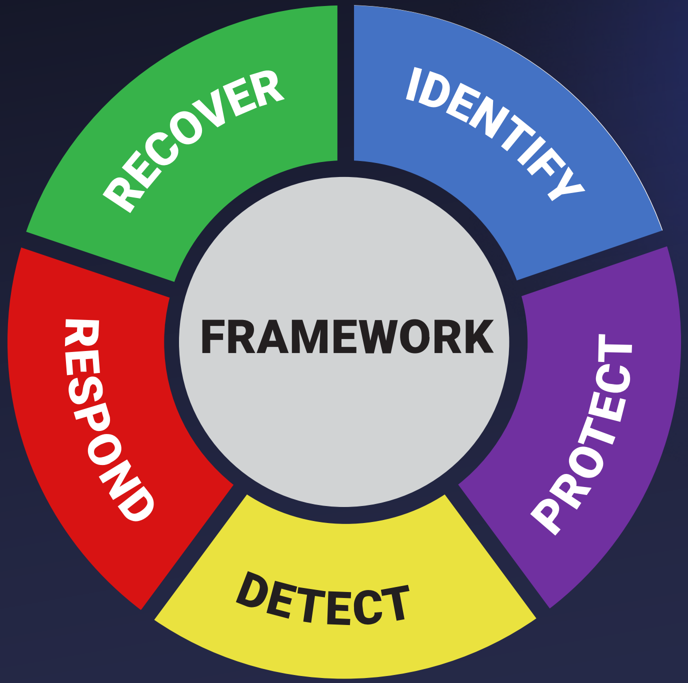
From system elements,
Users, environment, agents, …

26.10.2006 7

Управление рисками и угрозами в Kubernetes

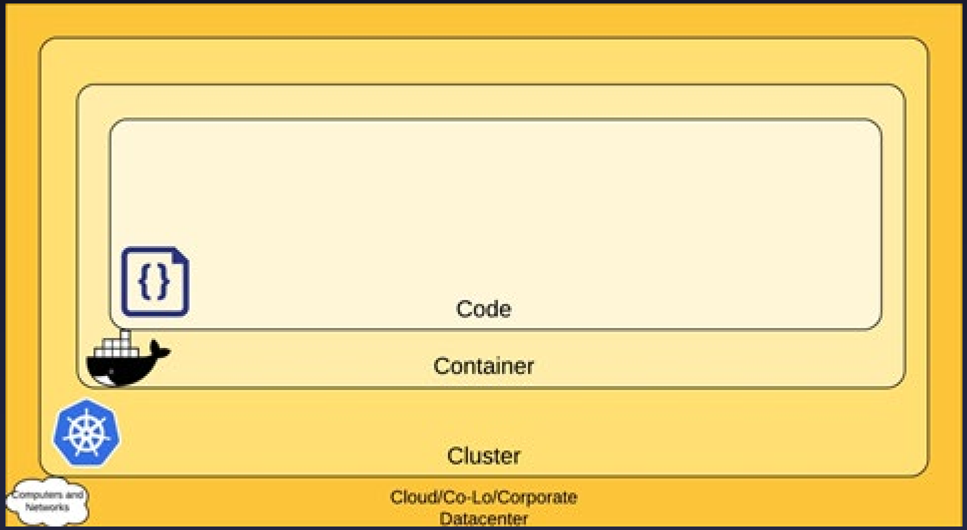
Управление угрозами в K8s

- Уязвимости были, есть и будут
- Нужно строить безопасность с мыслью, что они существуют и могут быть использованы злоумышленниками в любой момент



NIST CyberSecurity Framework

Инвентаризация (Identify)



- Identify = Observability (images, k8s resources, process, ...)
- Нельзя защищать, контролировать то, что не видишь
  - “A system must be observable to be controllable”

Identify

Asset Management

Business Environment

Governance

Risk Assessment

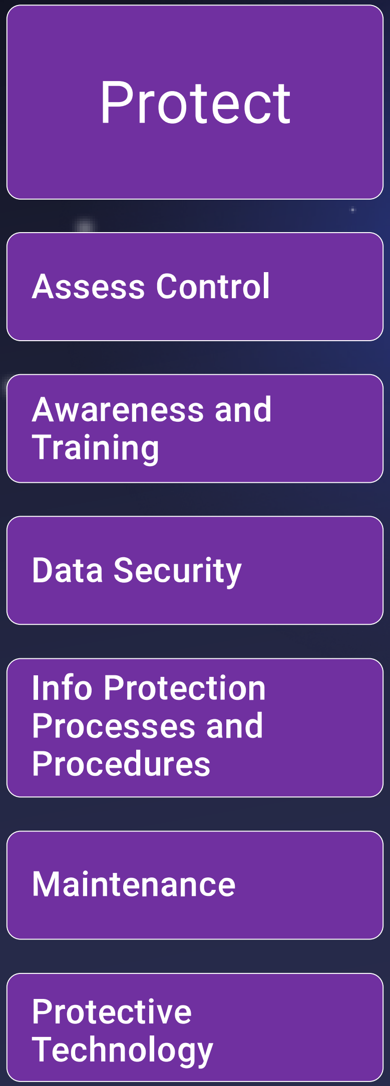
Risk Management Strategy

История 1

Инцидент SolidWinds

Защита (Protect)

- seccomp,
- AppArmor,
- SeLinux,
- PodSecurityPolicy,
- securityContext,
- NetworkPolicy,
- Admission controllers,
  - Policy engines
- …



Protect

Assess Control

Awareness and Training

Data Security

Info Protection Processes and Procedures

Maintenance

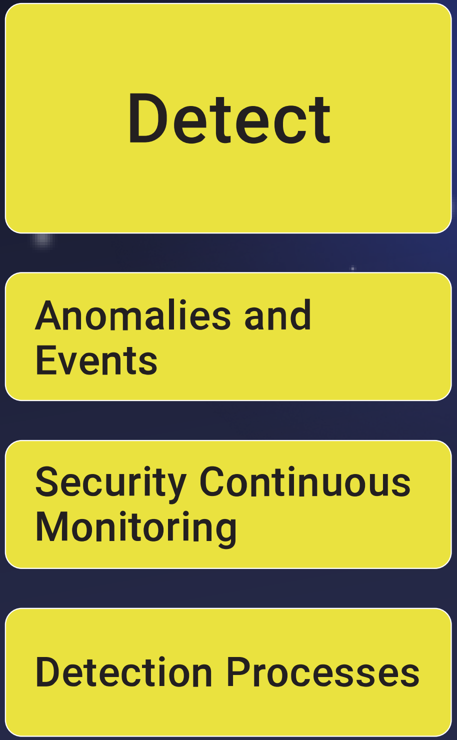
Protective Technology

История 2

“YX.XXX уязвимостей в образах – не проблема. Мы закрываем уязвимости с высоким уровнем и спим спокойно.”

Обнаружение (Detect)

- Detect ~= Observability,
- Audit policy,
- Admission controllers,
- возможности eBPF



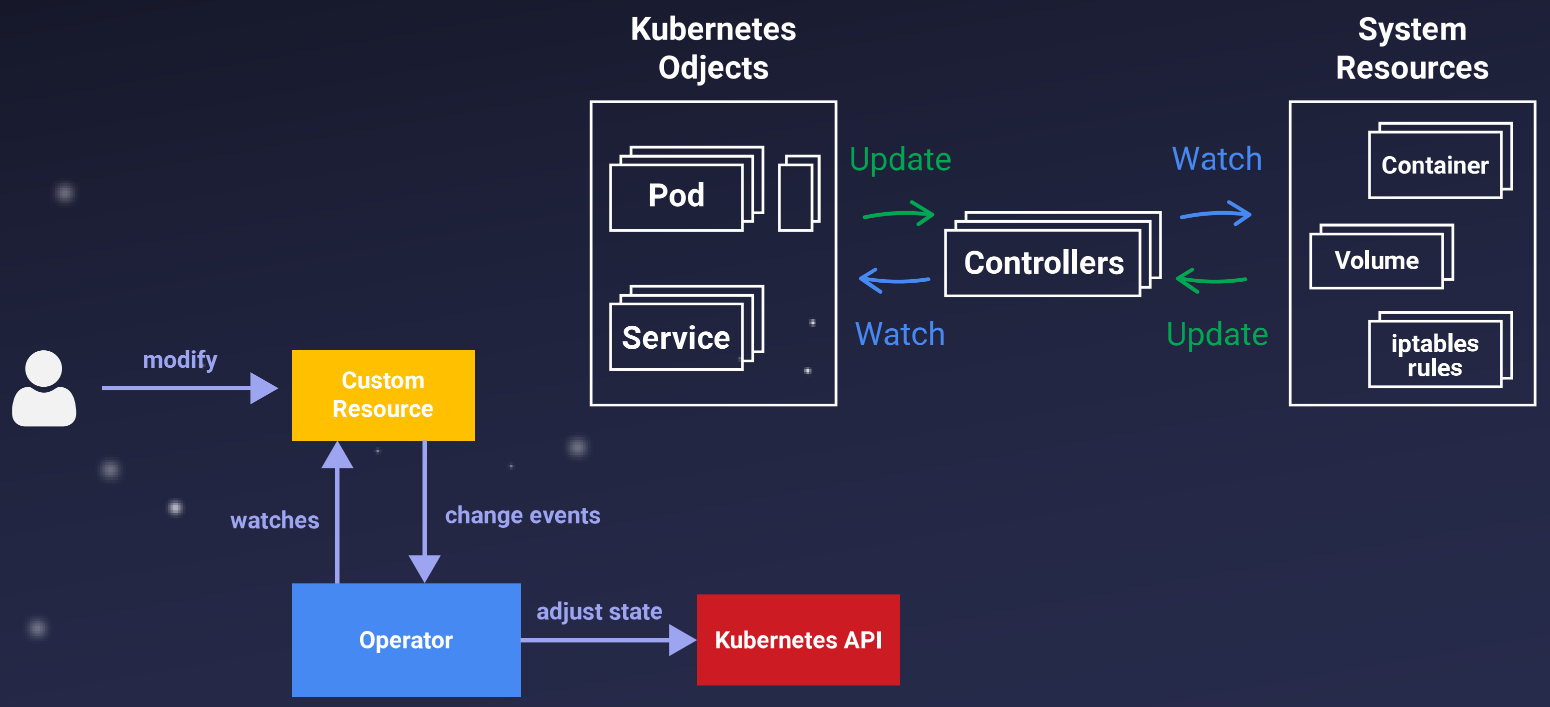
История 3

“BugBounty: а ты точно ничего не делал дальше?” или “О результатах пентестов мы узнаем из отчетов”

Реагирование и восстановление (Respond, Recover)

Respond/Recover = CRD + свой Operator

- Свойство конвергенции



Respond	Recover
Response Planning	Recovery Planning
Communications	Improvements
Analysis	Communications
Mitigation	
Improvements	

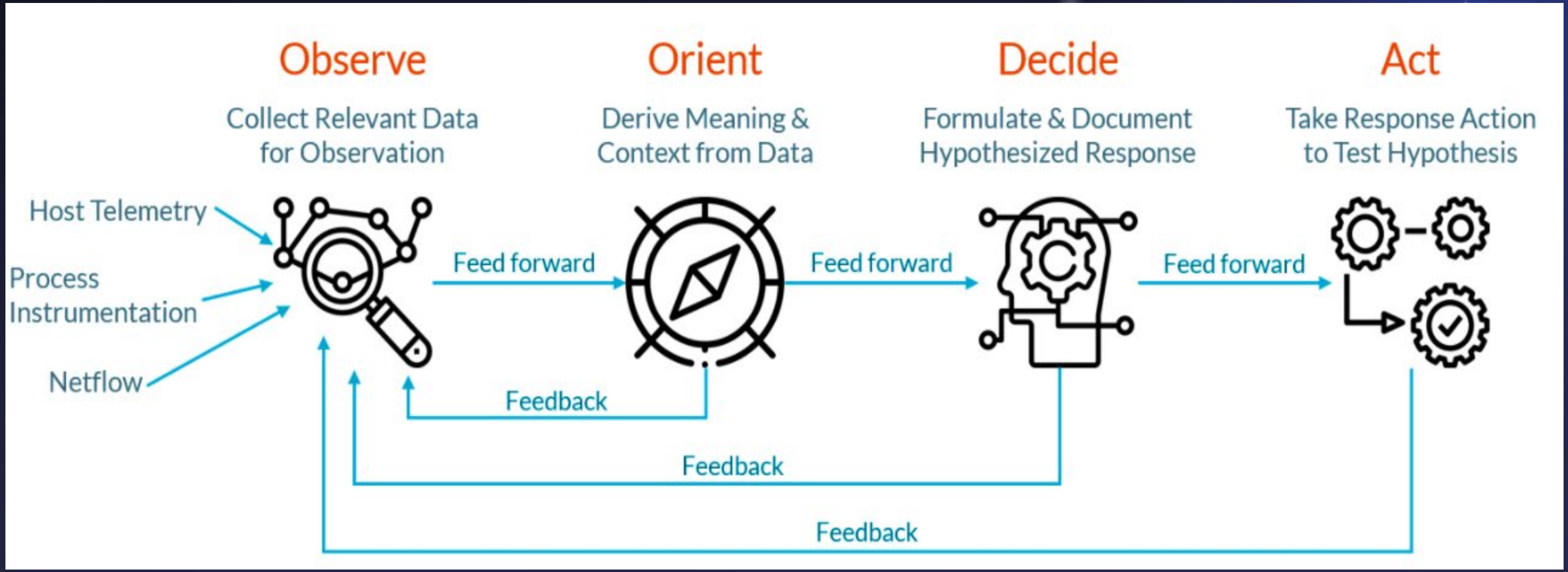
История 4

“Шел Шредингера”

OODA loop

OODA:

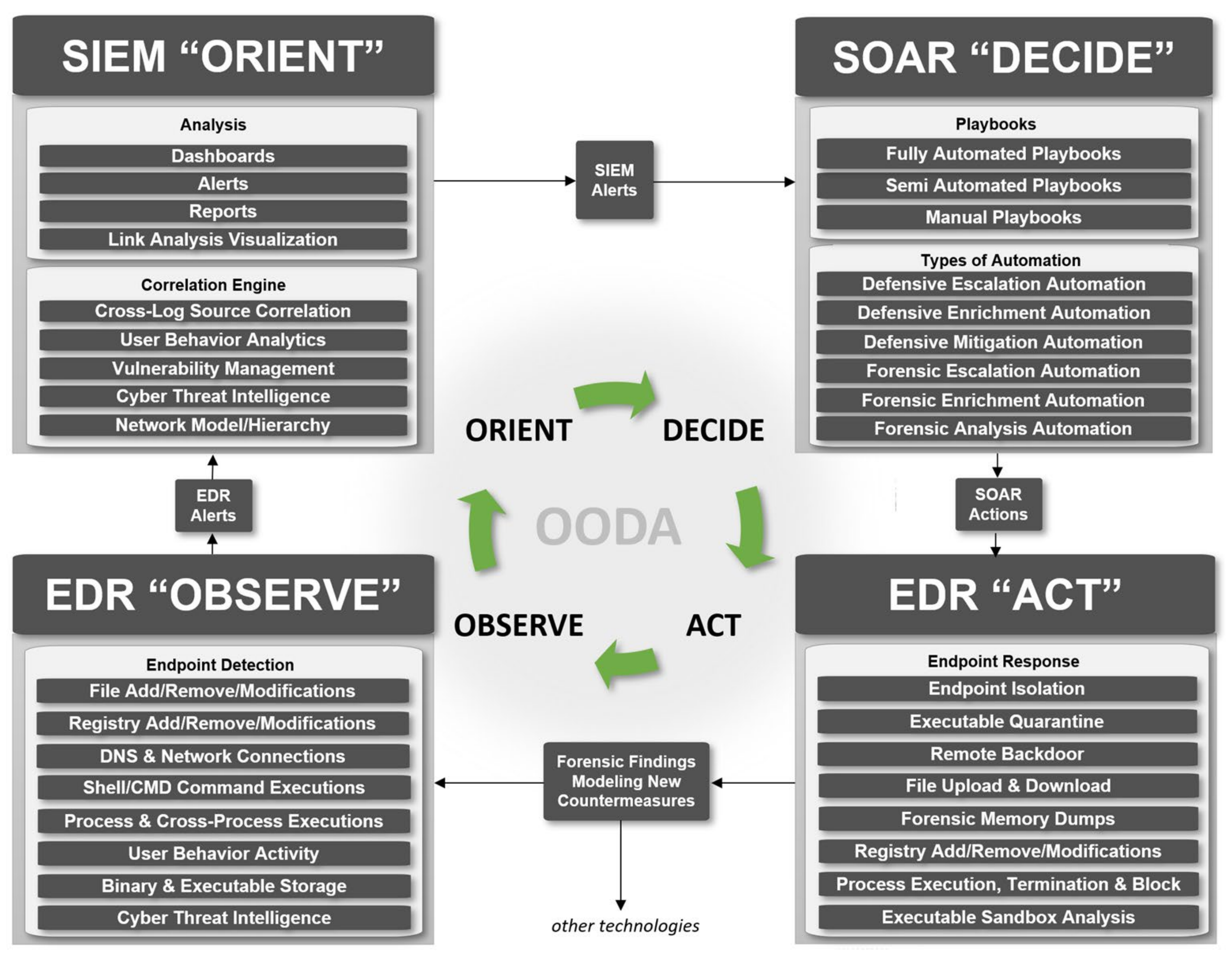
- Observe – наблюдение
- Orient – ориентирование
- Decide – решение
- Act – действие


Observe
Collect Relevant Data for Observation
Orient
Derive Meaning & Context from Data
Decide
Formulate & Document Hypothesized Response
Act
Take Response Action to Test Hypothesis
Host Telemetry
Process Instrumentation
Netflow
Feed forward
Feed forward
Feed forward
Feedback
Feedback
Feedback

Image

Безопасность – это процесс, а не состояние!

OODA loop in enterprise



SIEM "ORIENT"

Analysis
- Dashboards
- Alerts
- Reports
- Link Analysis Visualization

Correlation Engine
- Cross-Log Source Correlation
- User Behavior Analytics
- Vulnerability Management
- Cyber Threat Intelligence
- Network Model/Hierarchy

SIEM Alerts

SOAR "DECIDE"

Playbooks
- Fully Automated Playbooks
- Semi Automated Playbooks
- Manual Playbooks

Types of Automation
- Defensive Escalation Automation
- Defensive Enrichment Automation
- Defensive Mitigation Automation
- Forensic Escalation Automation
- Forensic Enrichment Automation
- Forensic Analysis Automation

SOAR Actions

EDR Alerts

EDR "OBSERVE"

Endpoint Detection
- File Add/Remove/Modifications
- Registry Add/Remove/Modifications
- DNS & Network Connections
- Shell/CMD Command Executions
- Process & Cross-Process Executions
- User Behavior Activity
- Binary & Executable Storage
- Cyber Threat Intelligence

Forensic Findings Modeling New Countermeasures

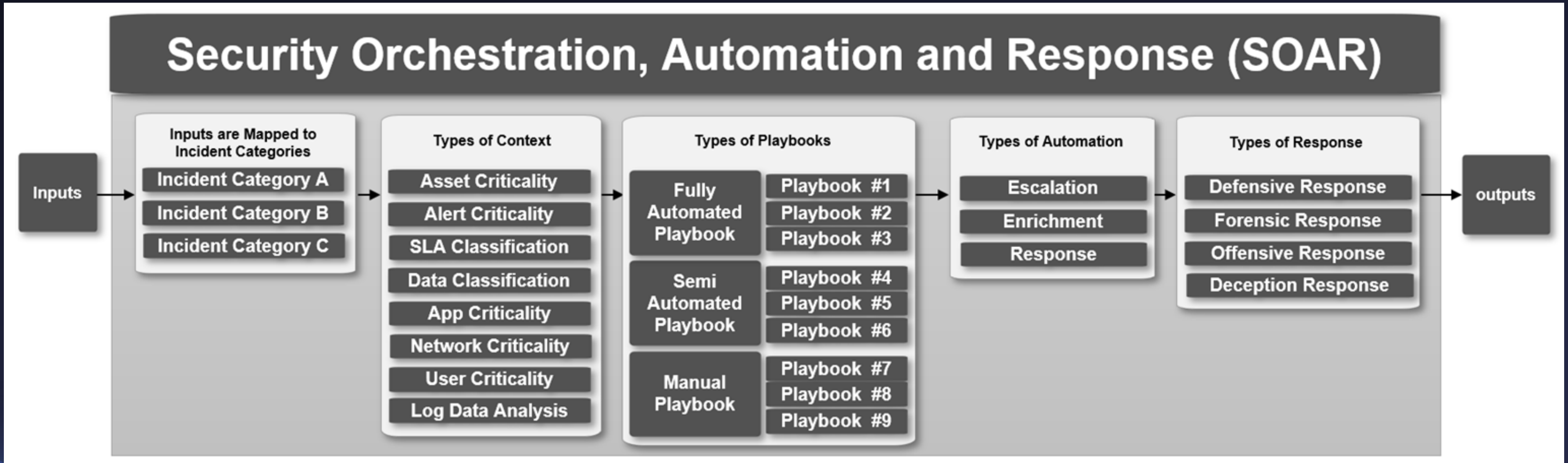
other technologies

EDR "ACT"

Endpoint Response
- Endpoint Isolation
- Executable Quarantine
- Remote Backdoor
- File Upload & Download
- Forensic Memory Dumps
- Registry Add/Remove/Modifications
- Process Execution, Termination & Block
- Executable Sandbox Analysis

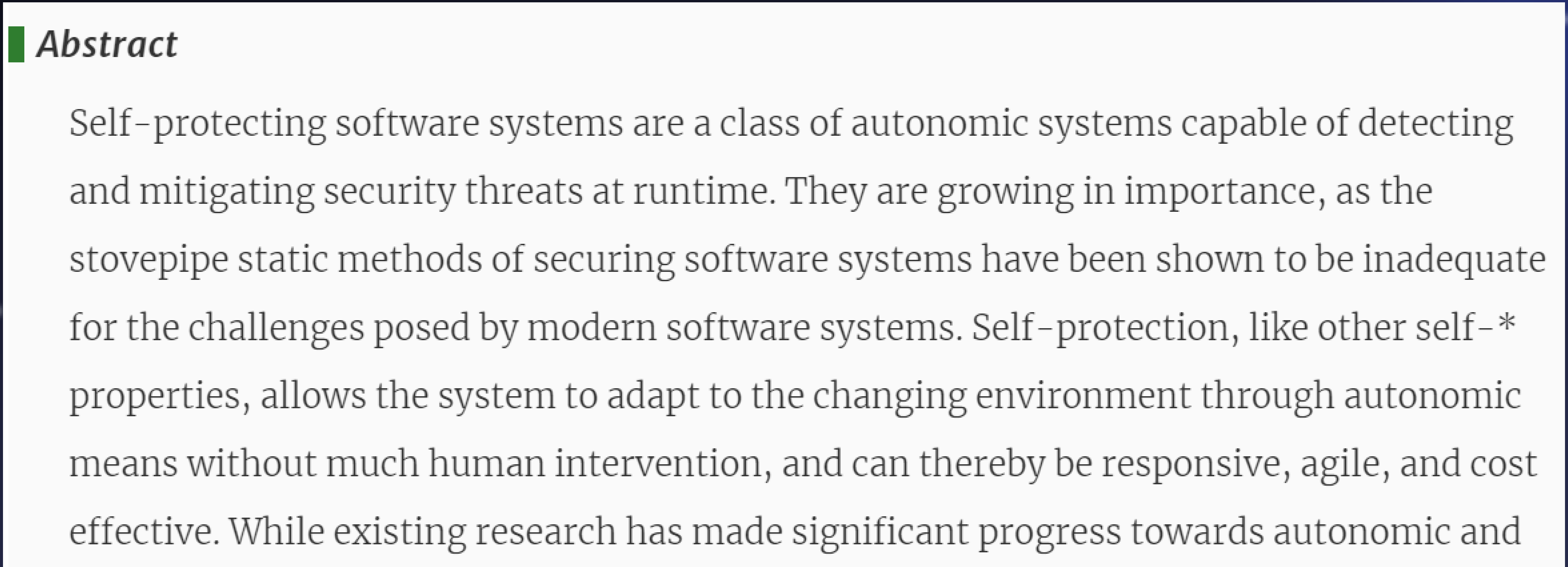
Image

SOAR


Security Orchestration, Automation and Response (SOAR)
Inputs
Inputs are Mapped to Incident Categories
Incident Category A
Incident Category B
Incident Category C
Types of Context
Asset Criticality
Alert Criticality
SLA Classification
Data Classification
App Criticality
Network Criticality
User Criticality
Log Data Analysis
Types of Playbooks
Fully Automated Playbook
Playbook #1
Playbook #2
Playbook #3
Semi Automated Playbook
Playbook #4
Playbook #5
Playbook #6
Manual Playbook
Playbook #7
Playbook #8
Playbook #9
Types of Automation
Escalation
Enrichment
Response
Types of Response
Defensive Response
Forensic Response
Offensive Response
Deception Response
outputs

Image

Zero Trust и Self-protecting



Abstract

Self-protecting software systems are a class of autonomic systems capable of detecting and mitigating security threats at runtime. They are growing in importance, as the stovepipe static methods of securing software systems have been shown to be inadequate for the challenges posed by modern software systems. Self-protection, like other self-* properties, allows the system to adapt to the changing environment through autonomic means without much human intervention, and can thereby be responsive, agile, and cost effective. While existing research has made significant progress towards autonomic and

[“A Systematic Survey of Self-Protecting Software Systems”]

LUNTRY

Точки внедрения и расширения

- Kubectl plugins
- Authentication Webhook
- Authorization Webhook
- Admission controllers:
  - Image Policy webhook
  - MutatingAdmissionWebhook
  - ValidatingAdmissionWebhook
- Dynamic Admission Control
  - SDK!
- Container Lifecycle Hooks
- Audit Log webhook backend

> You know yourself, your team and your product. Build around your requirement

LUNTRY

Эшелонированная оборона

Threat modeling

Code	Images	k8s resources	Authentication Webhook	Authorization Webhook	Admission controllers	Audit Log Webhook	Container/Sandbox/VM	Observability
SAST	Immutable	Labels, annotations	RBAC	RBAC	LimitRanger		Isolation	Asset management
DAST		IaC	IAM		ResourceQuota		Rootless containers, Capabilities	Security monitoring
IAST		Security as Code			PodSecurityPolicy		seccomp, AppArmor, Selinux, distroless images	Application monitoring
RASP		Compliance as Code			ImagePolicyWebhook		Limiting the blast radius	Anomaly detection
SCA		Configuration check			NetworkPolicy		Segregation of duties (Secrets, ServiceAccounts token)	Event resource
…					PodSecurityPolicy			
					MutatingAdmissionWebhook			
					Init containers + sidecars containers injection			
					ValidatingAdmissionWebhook			
					Custom Resource + operator (policy engines)			

+ Multi-tenancy*

Заключение

Заключение

- Не видитесь на маркетинг и стройте безопасность исходя из собственных рисков и угроз
- K8s – это фреймворк, который дает много возможностей, способов и точек встраивания для своего расширения
- K8s позволяет прозрачно встроить и использовать практики ИБ без вреда удобству разработки
- Высокий потенциал по observability в сочетании с control loop дает возможность реализовывать средства защиты, которые раньше сделать было сложно или невозможно

Q&A

Спасибо за внимание!

Email: de@luntry.ru

Twitter: @evdokimovds

Telegram: @Qu3b3c