URAL DIGITAL WEEKEND

Готовим сервис к безопасной работе в Kubernetes: рекомендации для разработчиков (и не только)

Анатолий Карпенко

Luntry

whoami

- Автоматизатор автоматизации в Luntry
- Любитель митапошных форматов; SPb Reliability Meetup, ITGM, TechTrain, DevOops, DEFCON'ы, SafeCode, БЕКОН, ТБ Форум
- Веду канал «Технологический Болт Генона»
- Рисую несмешные мемы

ДИСКЛЕЙМЕР

ЭТО ТЕКСТ КОТОРЫЙ НИКТО НЕ ЧИТАЕТ
ПОЭТОМУ Я ПРОГОВОРЮ ЧТО Я НЕ СМОГУ
РАССКАЗАТЬ ОБО ВСЁМ ЧЁМ ХОТЕЛОСЬ БЫ
НО ПОСТАРАЮСЬ ОТМЕТИТЬ ЧАСТЬ
ВАЖНЫХ ПУНКТОВ ДЛЯ РАЗРАБОТЧИКОВ
ОСТАЛЬНОЕ МОЖНО БУДЕТ ОБСУДИТЬ ПОСЛЕ
ДОКЛАДА ИЛИ ПОСМОТРЕТЬ ПО ССЫЛКАМ
КОТОРЫЕ Я ОСТАВЛЮ В ПРЕЗЕНТАЦИИ

Разработчики?

Разработчики?

QA?

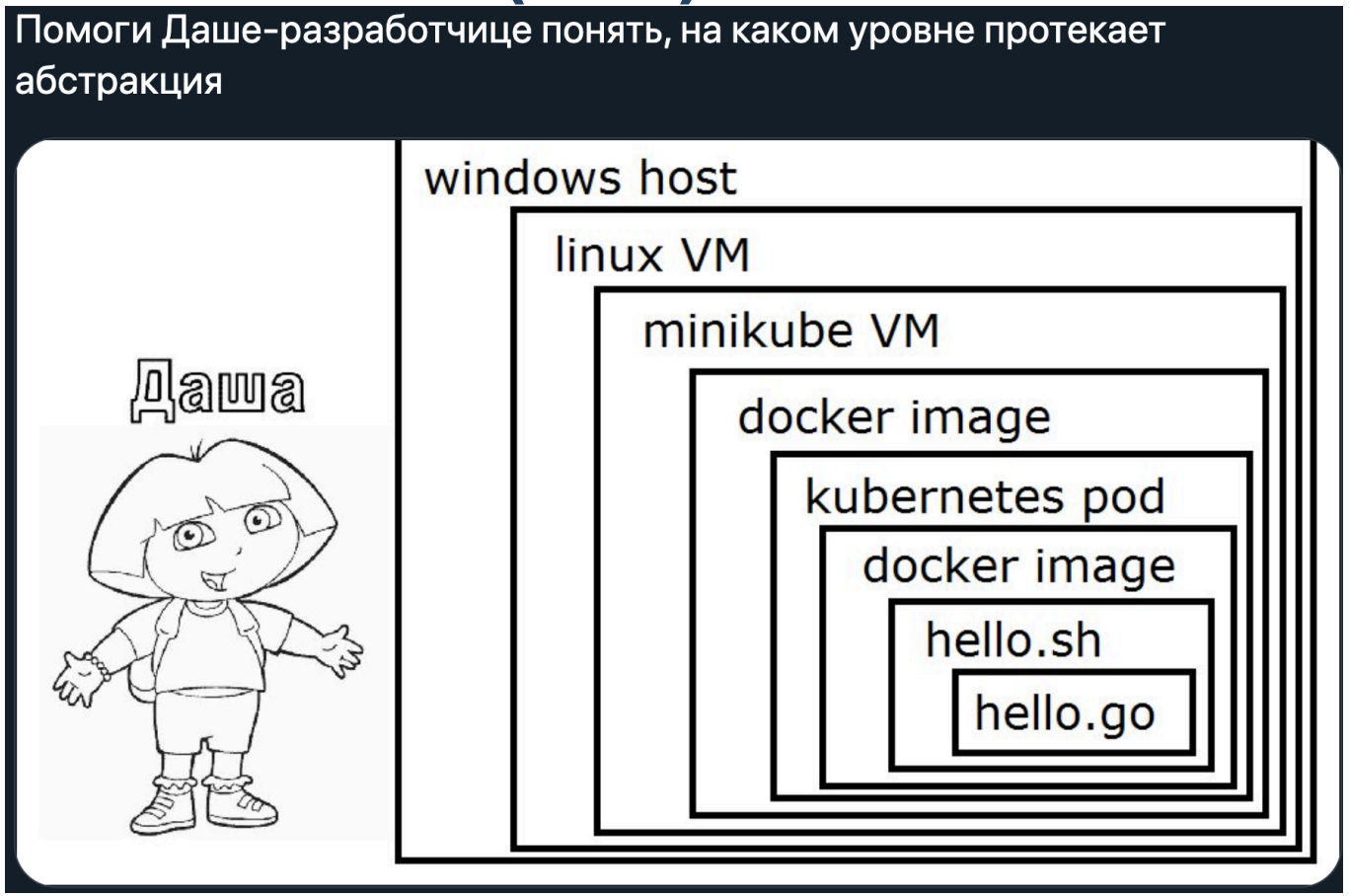
Разработчики?

QA?

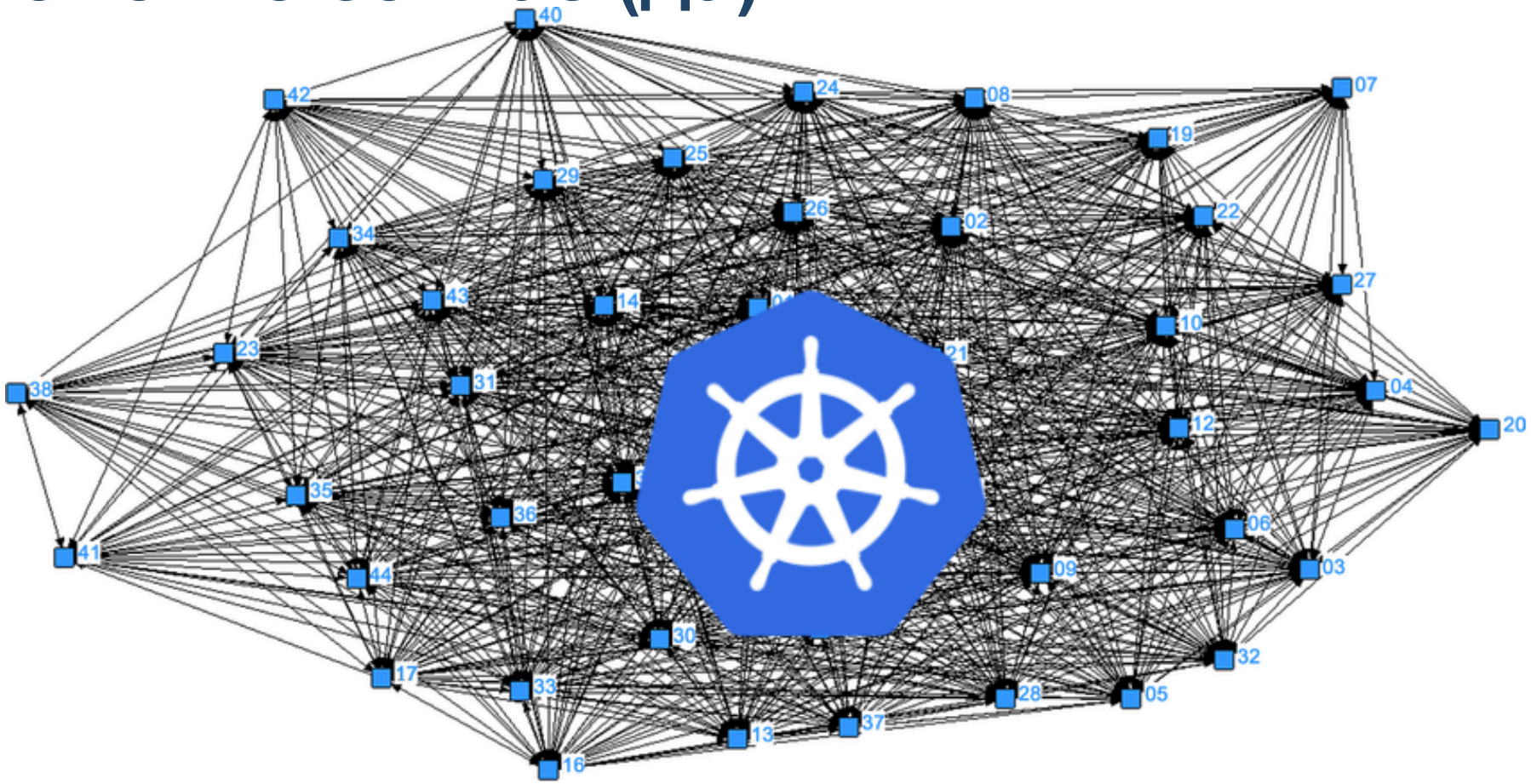
Админы, SRE, Dev[Sec]Ops и прочие Ops'ы?

Приложение сейчас (нет)

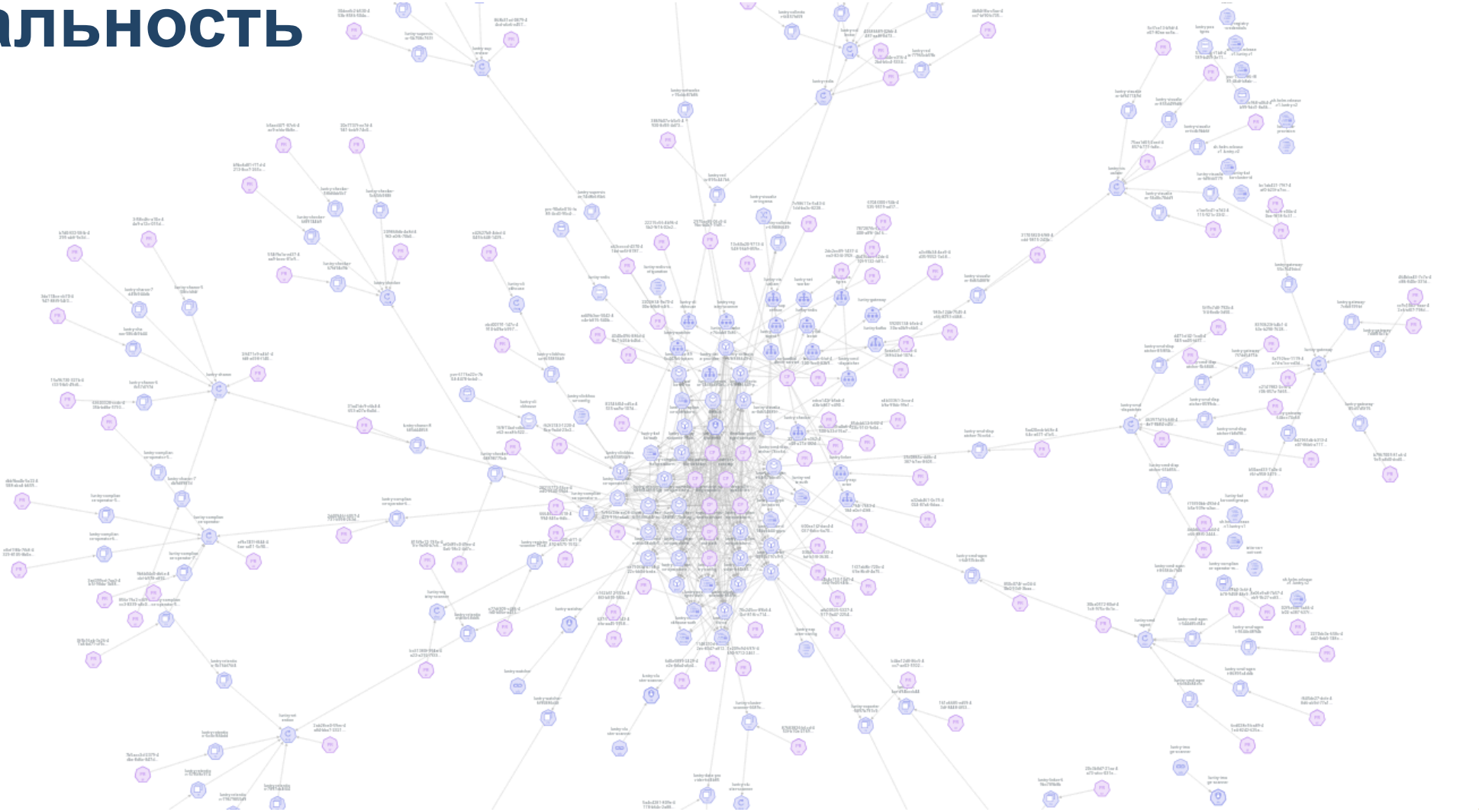
Помоги Даше-разработчице понять, на каком уровне протекает абстракция



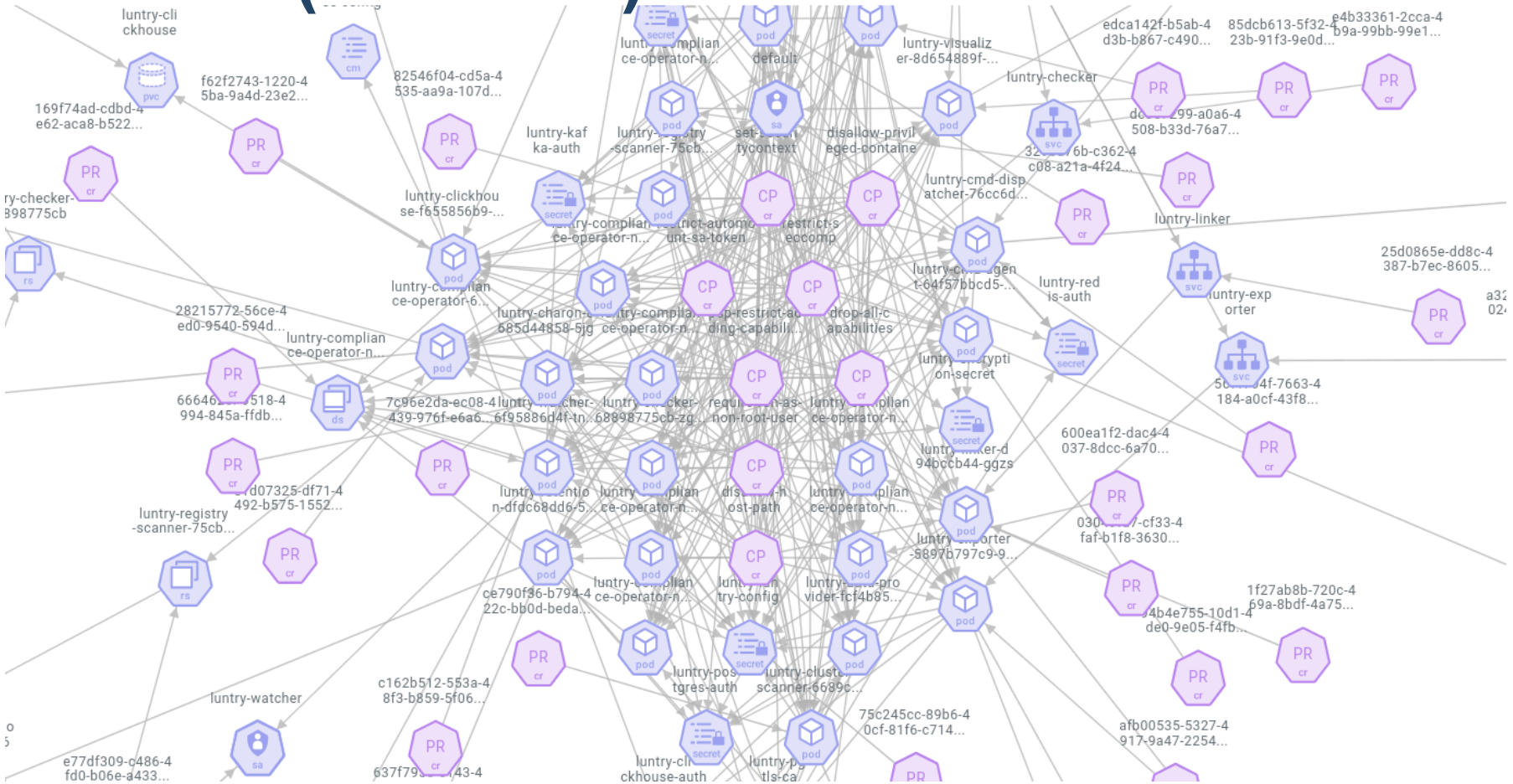
Приложение сейчас (да)



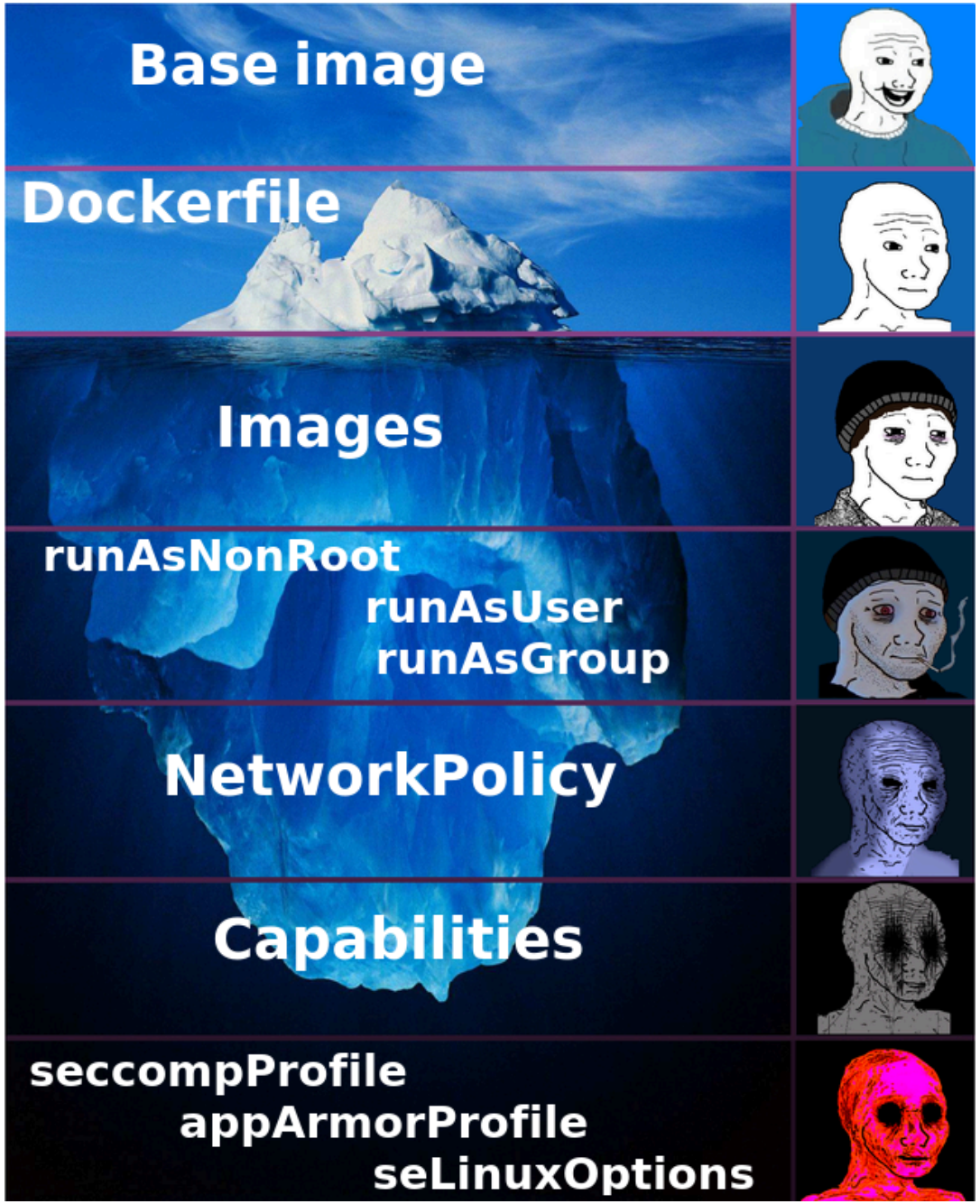
Реальность



Реальность (Zoom in)



Давайте разбираться (погружаться)



Кто пишет Dockerfile?

Кто пишет Dockerfile?

Кто пишет манифесты/чарты под Kubernetes?

На что опереться

- Docker Building best practices
- CIS Docker Benchmarks
- CIS Kubernetes Benchmarks
- NIST 800-190. Application Container Security Guide
- Приказ ФСТЭК №118. Требования по безопасности информации к средствам контейнеризации
- О повышении безопасности средств защиты информации, в состав которых разработчики включают средства контейнеризации или образы контейнеров

Base image

Kafka

Upcoming changes to the Bitnami catalog (effective August 28th, 2025)

Minimal images

- gcr.io (distroless)
- RapidFort
- Chainguard $ (`latest` ✅)
- VulnFree $
- SIGHUP 🫡
- . . .

LotL-атака (Living off the Land)

- Антивирус не поможет
- Вредоносных файлов нет
- Shell легитимен
- Расследование и атрибуция атак сложны

```
/tmp $ TF=$(mktemp)
/tmp $ chmod +x $TF
/tmp $ echo -e '#!/bin/sh\n/bin/sh 1>&0' >$TF
/tmp $ wget --use-askpass=$TF 0
$
```

```
$ TF=$(mktemp)
$ echo 'os.execute("/bin/sh")' > $TF
$ sudo nmap --script=$TF
Starting Nmap 7.80 ( https://nmap.org ) at 2025-07-14
NSE: Warning: Loading '/tmp/tmp.41C2Gov4kT' -- the rec
 extension is '.nse'.
# uid=0(root) gid=0(root) группы=0(root)
#
```

Dockerfile

SAST

- Hadolint
- KICS
- Checkov
- Semgrep
- ...

https://vkvideo.ru/video-211664921_456239070

Пример KICS

```
$ kics scan -p Dockerfile
. . .
Last User Is 'root', Severity: HIGH, Results: 1

Platform: Dockerfile
CWE: 250

        [1]: Dockerfile:46

                045:
                046: USER root
                047:
```

https://docs.kics.io/latest/queries/dockerfile-queries/67fd0c4a-68cf-46d7-8c41-bc9fba7e40ae

Пример KICS

fix(dockerfile): remove user root and add platform #7031

Merged **cx-andre-peixoto** merged 6 commits into `master` from `AST-40561` on Apr 30, 2024

Conversation 1 · Commits 6 · Checks 0 · Files changed 2

cx-rui-araujo commented on Apr 29, 2024

Closes #

Proposed Changes

- Remove user root
- Add platform to command FROM
- Change GIT image tag from 'latest' to 'latest-root'

update(dockerfile): revert KICS user change from 65532 back to root

Merged **cx-rui-araujo** merged 7 commits into `master` from `ruiar/update-kics-image` on Jan 31

Conversation 3 · Commits 7 · Checks 31 · Files changed 6

cx-rui-araujo commented on Jan 31 • edited — Contributor

Reason for Proposed Changes

- KICS always used root, but in the latest release, we changed it to 65532 due to FedRAMP policies. This change is causing issues in AST-CLI. Since we only use the binaries inside the image, there is no relevant reason to use 65532 instead of root. A possible fix is to run the KICS image with the same user as the container running it. However, since this would require users to change their setup, I am reverting this change to avoid that and to prevent other potential issues.

Proposed Changes

- revert KICS user change from 65532 back to root
- update KICS gh action to v2.1.4
- fix vulnerabilities

Reviewers cx-ed cx-art

Assignees cx-rui

Labels dockerfile

Projects

#7031,#7322, The Federal Risk and Authorization Management Program(FedRAMP)

Пример Hadolint

```
1 FROM scratch as base
  <string>:2:18:
  |
  2 | COPY --chmod=777 hadolint /bin/hadolint
  | ^
  invalid flag: --chmod
2 COPY --chmod=777 hadolint /bin/hadolint
3 LABEL org.opencontainers.image.source="https://github.com/hado
4 CMD ["/bin/hadolint", "-"]
5
6 FROM debian:bullseye-slim as debian
7 COPY --chmod=777 hadolint /bin/hadolint
8 LABEL org.opencontainers.image.source="https://github.com/hado
9
```

https://hadolint.github.io/hadolint/

Docker (COPY)

COPY

COPY has two forms:

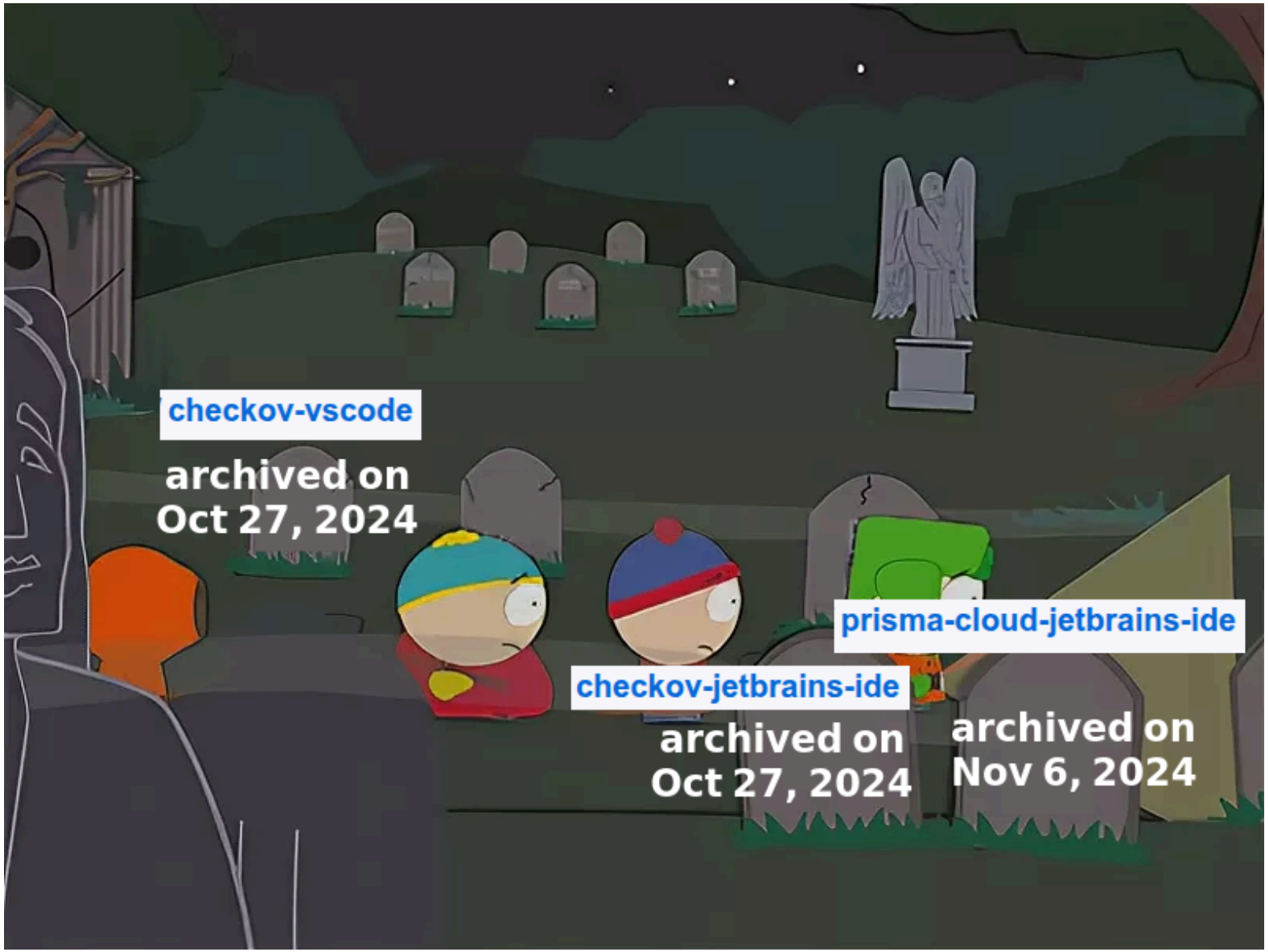
```
COPY [--chown=<user>:<group>] [--chmod=<perms>] <src>... <dest>
COPY [--chown=<user>:<group>] [--chmod=<perms>] ["<src>",... "<dest>"]
```

This latter form is required for paths containing whitespace

https://docs.docker.com/reference/dockerfile/#copy---chown---chmod

Плагины к IDE

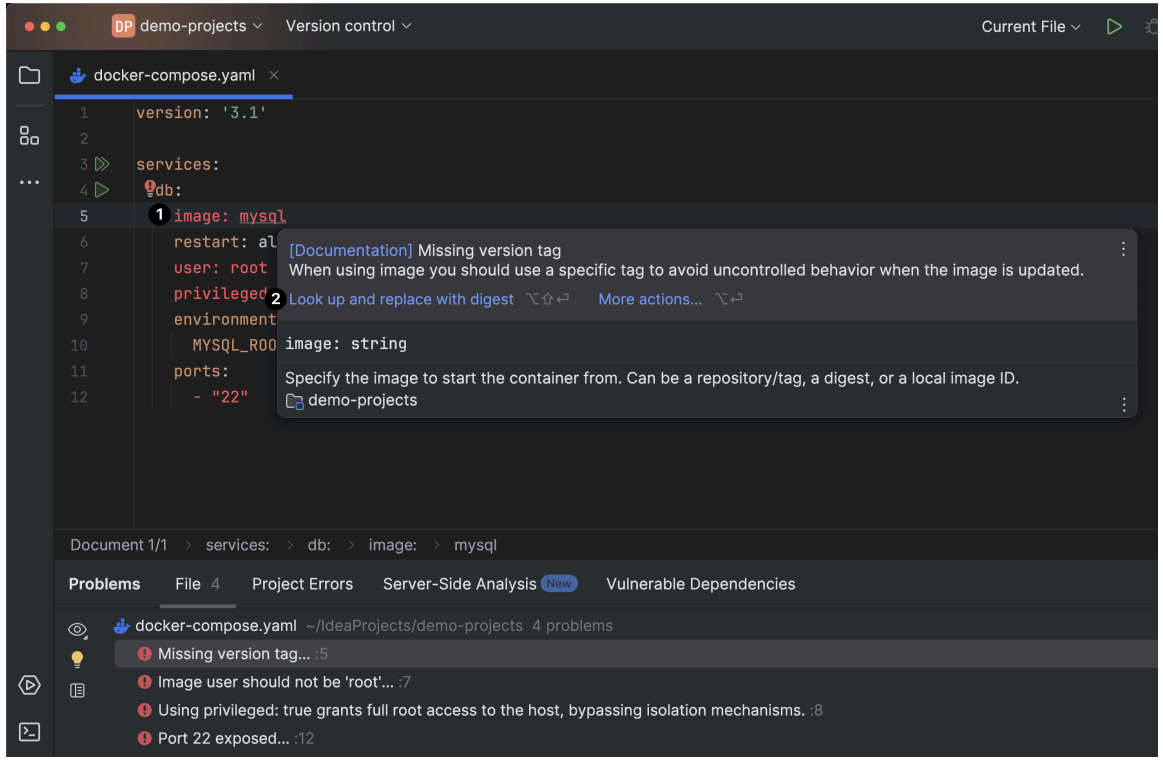
- Hadolint ✅
- Semgrep ✅
- KICS ❌
- Checkov 🫡



Hadolint,Semgrep,KiCS

Плагины к JB IDE's

- `curl` - Do Not Pipe to Shell
- `latest` Version Tags
- Secrets in ENV Keys
- Using RUN with sudo
- Standardise Remote GET Tools
- . . .
- AppArmor profile disabled or overridden
- Setting custom SELinux options
- Using non-root containers
- Using privileged containers
- . . .

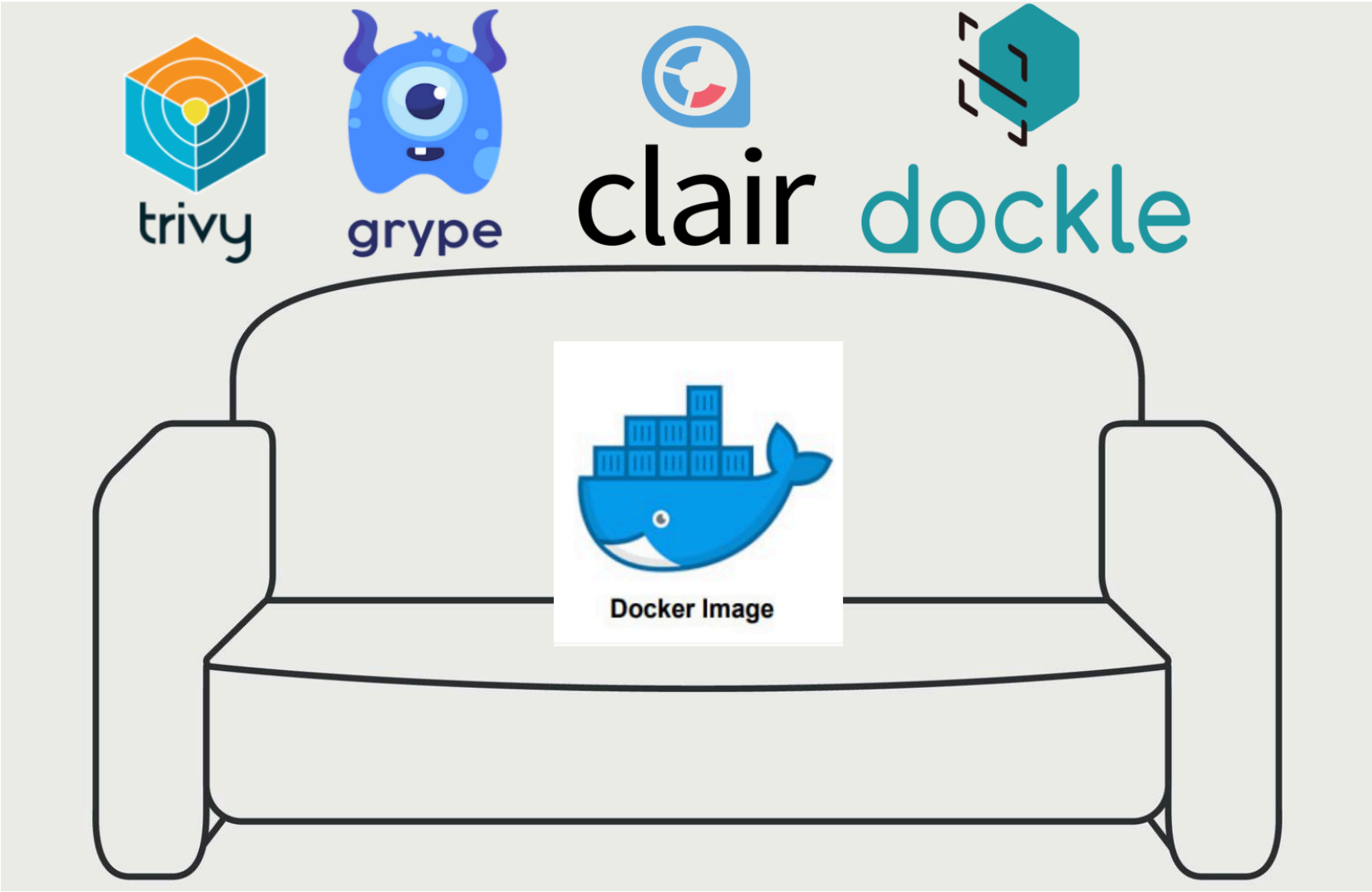


JetBrains Marketplace,GitHub,k8s (in)security

Images

Images

- Trivy
- Grype
- Clair
- Dockle
- ...



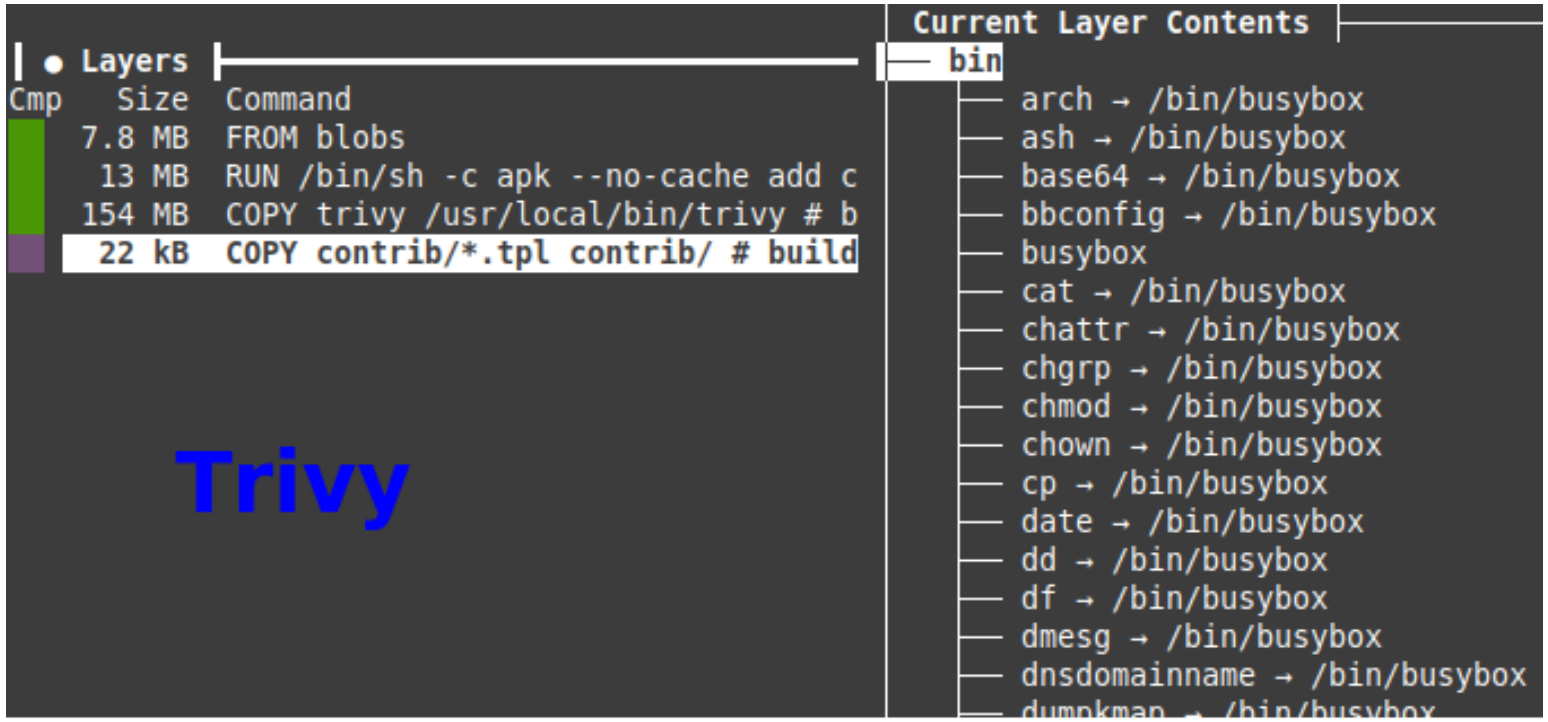
Trivy <-> Grype

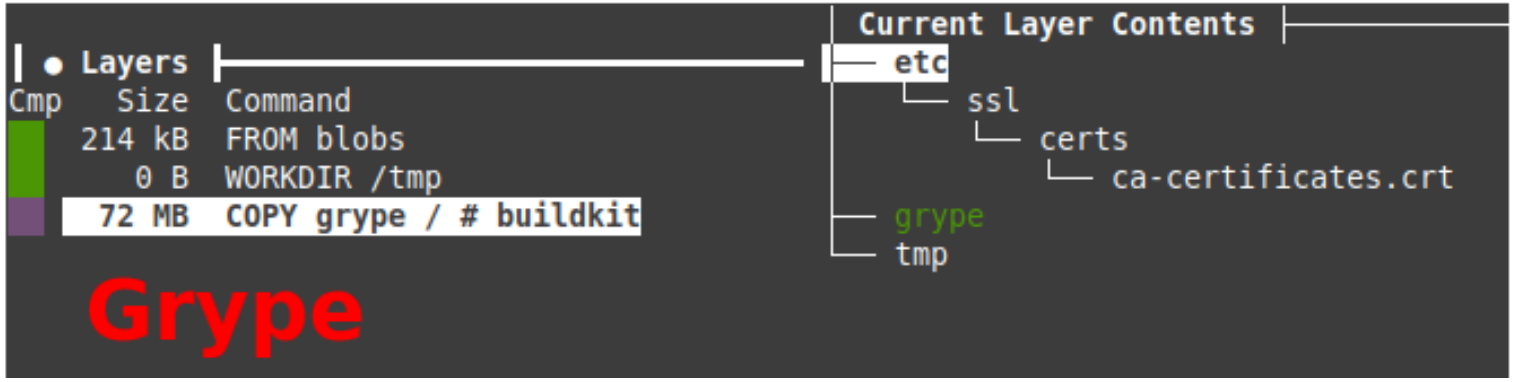
```
$ grype aquasec/trivy:latest --sort-by severity
 ✔ Loaded image
NAME                INSTALLED   FIXED-IN   TYPE       VULNERABILITY        SEVERITY  EPSS%  RISK
git                 2.47.2-r0   2.47.3-r0  apk        CVE-2025-48385       High      28.94  < 0.1
git-init-template   2.47.2-r0   2.47.3-r0  apk        CVE-2025-48385       High      28.94  < 0.1
git                 2.47.2-r0   2.47.3-r0  apk        CVE-2025-46835       High       3.72  < 0.1
git-init-template   2.47.2-r0   2.47.3-r0  apk        CVE-2025-46835       High       3.72  < 0.1
git                 2.47.2-r0   2.47.3-r0  apk        CVE-2025-27614       High       3.62  < 0.1
git-init-template   2.47.2-r0   2.47.3-r0  apk        CVE-2025-27614       High       3.62  < 0.1
git                 2.47.2-r0   2.47.3-r0  apk        CVE-2025-46334       High       3.31  < 0.1
git-init-template   2.47.2-r0   2.47.3-r0  apk        CVE-2025-46334       High       3.31  < 0.1
git                 2.47.2-r0   2.47.3-r0  apk        CVE-2025-48384       High       1.70  < 0.1
git-init-template   2.47.2-r0   2.47.3-r0  apk        CVE-2025-48384       High       1.70  < 0.1
helm.sh/helm/v3     v3.18.3     3.18.4     go-module  GHSA-557j-xg8c-q2mm  High       0.38  < 0.1
git                 2.47.2-r0   2.47.3-r0  apk        CVE-2025-48386       Medium     1.50  < 0.1
```

```
$ trivy image --severity HIGH anchore/grype:latest

Report Summary

┌────────┬──────────┬─────────────────┬─────────┐
│ Target │   Type   │ Vulnerabilities │ Secrets │
├────────┼──────────┼─────────────────┼─────────┤
│ grype  │ gobinary │        0        │    -    │
└────────┴──────────┴─────────────────┴─────────┘

Legend:
- '-': Not scanned
- '0': Clean (no security findings detected)
```

Grype -> Trivy

Trivy -> Grype

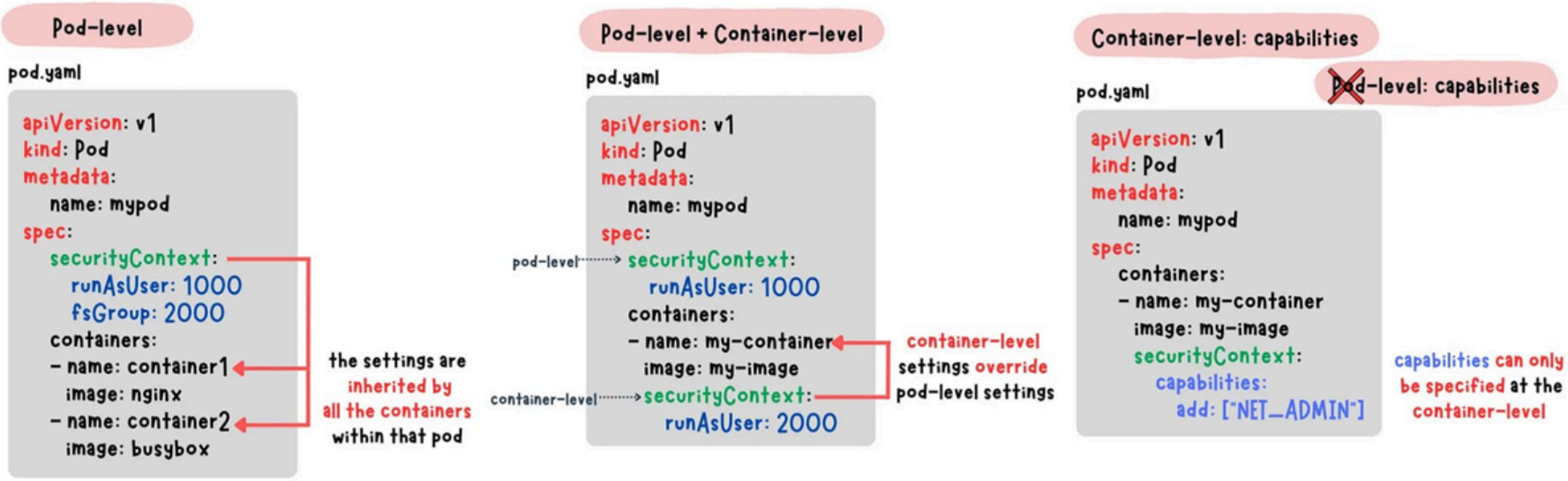

● Layers
Cmp Size Command
7.8 MB FROM blobs
13 MB RUN /bin/sh -c apk --no-cache add c
154 MB COPY trivy /usr/local/bin/trivy # b
22 kB COPY contrib/*.tpl contrib/ # build
Trivy
Current Layer Contents
bin
arch → /bin/busybox
ash → /bin/busybox
base64 → /bin/busybox
bbconfig → /bin/busybox
busybox
cat → /bin/busybox
chattr → /bin/busybox
chgrp → /bin/busybox
chmod → /bin/busybox
chown → /bin/busybox
cp → /bin/busybox
date → /bin/busybox
dd → /bin/busybox
df → /bin/busybox
dmesg → /bin/busybox
dnsdomainname → /bin/busybox


● Layers
Cmp Size Command
214 kB FROM blobs
0 B WORKDIR /tmp
72 MB COPY grype / # buildkit
Grype
Current Layer Contents
etc
ssl
certs
ca-certificates.crt
grype
tmp

Продолжаем погружаться (в Kubernetes)

> SecurityContext holds security configuration that will be applied to a container

> PodSecurityContext holds pod-level security attributes and common container settings.



k8s v1.33 API, Kubernetes: Security Contexts

Users

runAsNonRoot/runAsUser/runAsGroup

> The UID to run the entrypoint of the container process
> Defaults to user specified in image metadata if unspecified

> The GID to run the entrypoint of the container process
> Uses runtime default if unset

```
spec:
  securityContext:
    runAsNonRoot: true
    runAsUser: 1000
    runAsGroup: 1000
```

```
Exception in thread "main" java.lang.IllegalStateException:
  Unable to create the directory [/tomcat.8888] to use as the base directory
        at org.apache.catalina.startup.Tomcat.initBaseDir(Tomcat.java:847)
        at org.apache.catalina.startup.Tomcat.getServer(Tomcat.java:613)
        at org.apache.catalina.startup.Tomcat.getEngine(Tomcat.java:586)
        at org.apache.catalina.startup.Tomcat.getHost(Tomcat.java:570)
        at launch.Main.main(Main.java:20)
```

NetworkPolicy

NetworkPolicy

> allow you to specify how a pod is allowed to communicate with various network "entities"

```
spec:
  podSelector:
    matchLabels:
      app: pod-two
  policyTypes:
    - Ingress
    - Egress
  ingress:
    - from:
      - podSelector:
          matchLabels:
            app: pod-one
  egress:
    - to:
      - podSelector:
          matchLabels:
            app: pod-one
```

Ural Digital Weekend 2023

Spectr TAGLINE

Сергей Канибор
Luntry, Security Researcher

Создание Network Policy в Kubernetes:

ключевые аспекты и рекомендации для разработчиков

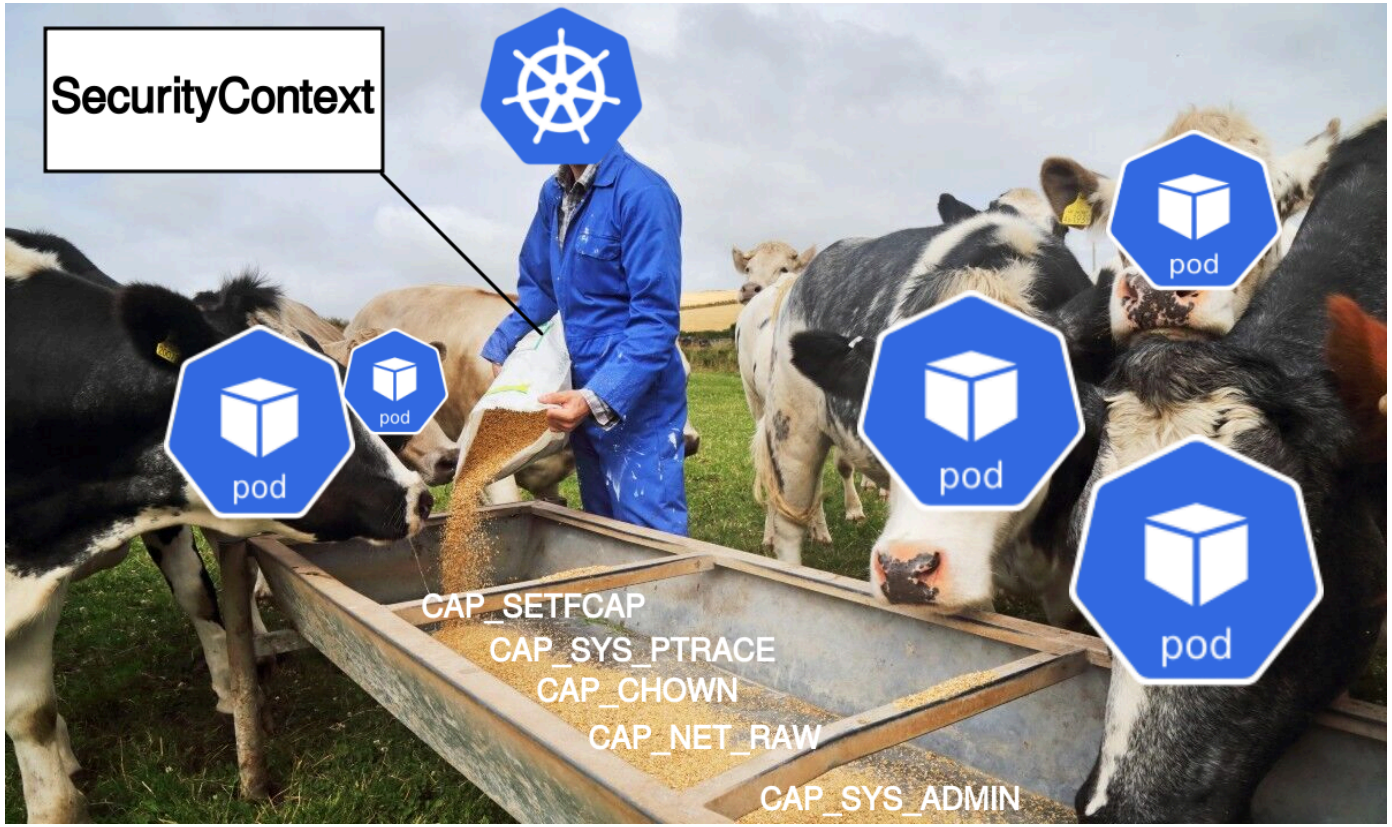
https://www.youtube.com/watch?v=4PdRXVhL4X8

capabilities

capabilities

> Adds and removes POSIX capabilities from running containers

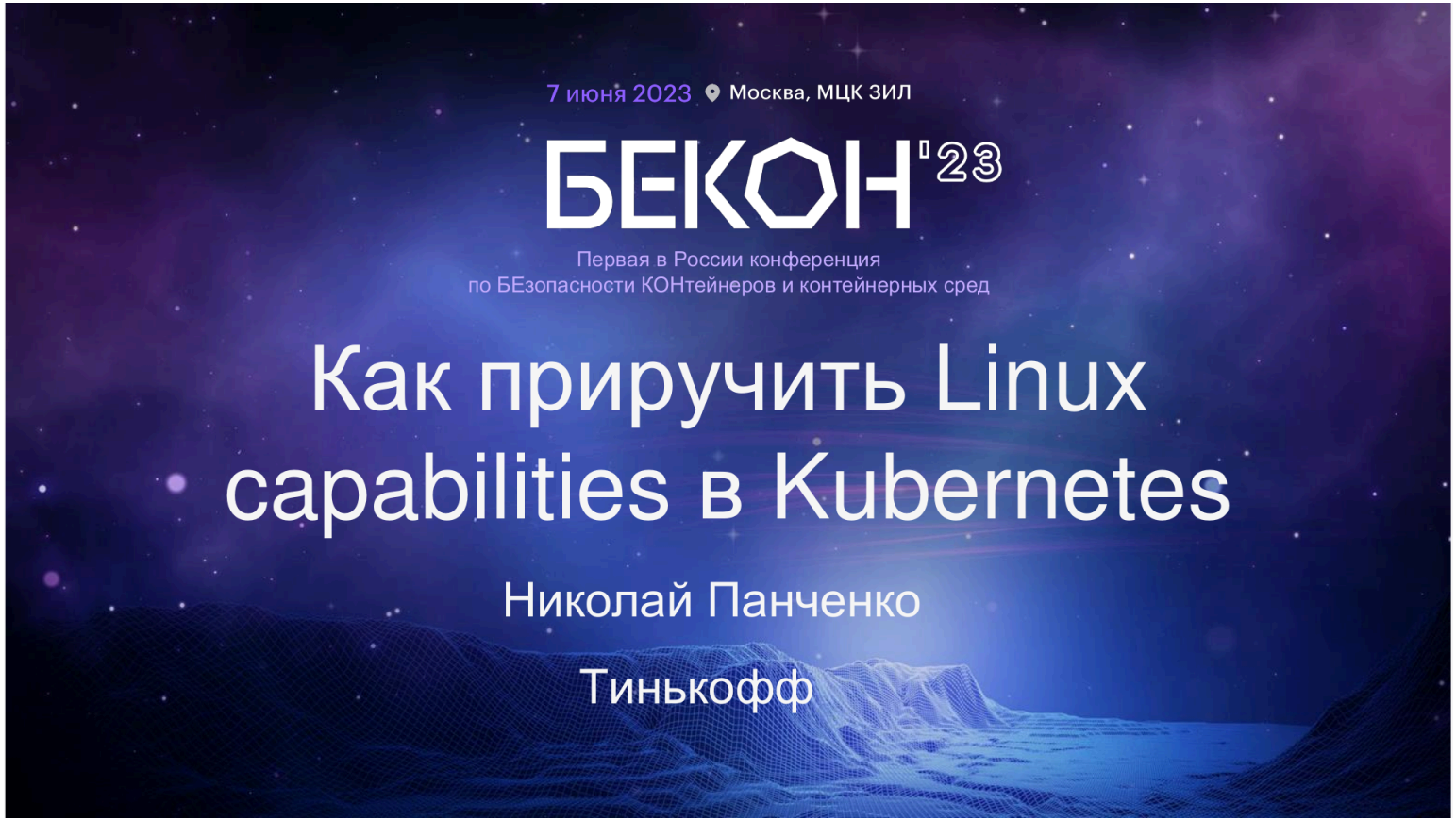
```
- CAP_CHOWN
- CAP_DAC_OVERRIDE
- CAP_DAC_READ_SEARCH
- CAP_SETUID
- CAP_SETGID
- CAP_NET_RAW
- CAP_SYS_ADMIN
- CAP_SYS_PTRACE
- CAP_SYS_MODULE
- CAP_FOWNER
- CAP_SETFCAP
- . . .
```



capsh

```
$ docker run -it --rm r0binak/mtkpi:v1.5 /bin/bash
. . .
# capsh --print | grep 'Bounding set'
Bounding set =cap_chown,cap_dac_override,cap_fowner,cap_fsetid,cap_kill
cap_setgid,cap_setuid,cap_setpcap,cap_net_bind_service,cap_net_raw,
cap_sys_chroot,cap_mknod,cap_audit_write,cap_setfcap

$ docker run -it --rm --privileged r0binak/mtkpi:v1.5 /bin/bash
. . .
# capsh --print | grep 'Bounding set'
Bounding set =cap_chown,cap_dac_override,cap_dac_read_search,cap_fowner,
cap_fsetid,cap_kill,cap_setgid,cap_setuid,cap_setpcap,cap_linux_immutable,
cap_net_bind_service,cap_net_broadcast,cap_net_admin,cap_net_raw,
cap_ipc_lock,cap_ipc_owner,cap_sys_module,cap_sys_rawio,cap_sys_chroot,
cap_sys_ptrace,cap_sys_pacct,cap_sys_admin,cap_sys_boot,cap_sys_nice,
cap_sys_resource,cap_sys_time,cap_sys_tty_config,cap_mknod,cap_lease,
cap_audit_write,cap_audit_control,cap_setfcap,cap_mac_override,
cap_mac_admin,cap_syslog,cap_wake_alarm,cap_block_suspend,cap_audit_read,38,39,40
```

Linux capabilities и Kubernetes



https://vkvideo.ru/video-224467080_456239034

Linux kernel security constraints

seccompProfile

> SeccompProfile defines a pod/container's seccomp profile settings

```yaml
apiVersion: v1
kind: Pod
. . .
spec:
  securityContext:
    seccompProfile:
      type: Localhost
      localhostProfile: file.json
  containers:
  - name: my-best-app
    image: my-best-container:latest
    securityContext:
      allowPrivilegeEscalation: false
```

```json
{
    "defaultAction": "SCMP_ACT_ERRNO",
    "syscalls": [
        {
            "names": [
                . . .
                "read",
                "write",
                "exit",
                . . .
            ],
            "action": "SCMP_ACT_ALLOW"
        }
    ]
}
```

https://kubernetes.io/docs/tutorials/security/seccomp/

```
$ diff default.json default_ww.json
380d379
<                                 "write",
```

```
$ docker run -it --rm \
        --security-opt seccomp='default.json' \
                              ubuntu:22.04 /bin/bash
root@5d612022de21:/#
```

```
$ docker run -it --rm \
        --security-opt seccomp='default_ww.json' \
                              ubuntu:22.04 /bin/bash

docker: Error response from daemon: OCI runtime start failed
```

appArmorProfile / seLinuxOptions

Запрет записи на ФС через AppArmor

```
#include <tunables/global>

profile k8s-apparmor-example-deny-write
            flags=(attach_disconnected) {
  #include <abstractions/base>

  file,

  # Deny all file writes.
  deny /** w,
}
```

Пример запуска контейнера с конкретной меткой SELinux

```
...
securityContext:
  seLinuxOptions:
    user: system_u
    role: system_r
    type: container_t
    level: s0:c829,c861
```

Предотвращение Runtime угроз в контейнерах и Kubernetes

Технология	seccomp	Linux Security Modules (LSM)	
Реализация	seccomp профиль	AppArmor профиль	SeLinux политика
Краткое описание	Контроль на уровне системных вызовов	Контроль на уровне процессов. Mandatory Access Control (MAC)	Контроль на уровне процессов. Mandatory Access Control (MAC)
Основная цель	Усложнить побег из контейнера на хост через уязвимости ядра	Ограничить возможности контейнера для развития атаки и побега на хост	Усложнить побег из контейнера на хост через уязвимости ядра и некорректные конфигурации
Контроль над	Любым системным вызовом	Доступами к файлам, каталогам, сетевым портам, ресурсам системы, capabilities и действиями процессов	Доступами к файлам, каталогам, сетевым портам, ресурсам системы и действиями процессов
Версия ядра	2.6.12	2.6.36	2.6.0
Версия Kubernetes для GA	1.19	1.30	1.12
Предоставляется	Средствами ОС	Средствами ОС	Средствами ОС
Интеграция с Kubernetes	Раздел SecurityContext.SeccompProfile	Раздел SecurityContext.AppArmorProfile	Раздел SecurityContext.SELinuxOptions
Применение	Файл на Node	Файл на Node	Файл на Node
Режимы работы	Whitelist, Blacklist	Whitelist, Blacklist	Whitelist, Blacklist

https://luntry.ru/page72417913.html, https://vkvideo.ru/video-211664921_456239080?t=16m2s

Резюмируя сказанное

"Статика"

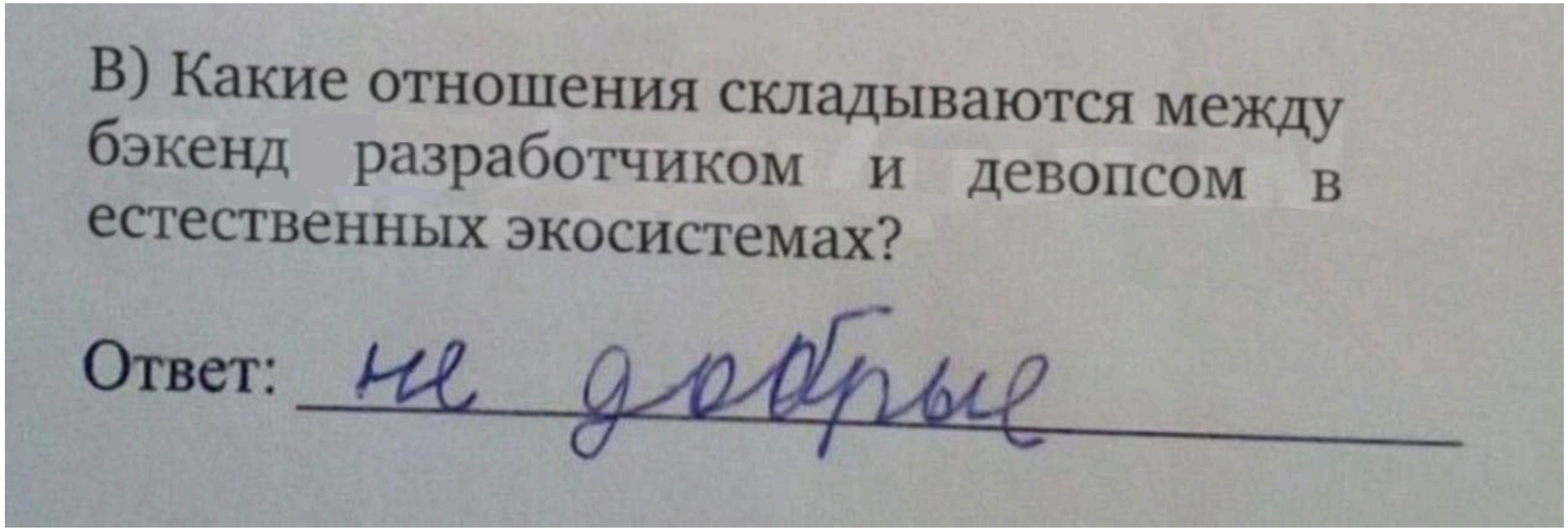
- Правильно выбираем базовый образ
  - Актуальные версии
  - Минималистичность
  - dev/prod-образы
- Проверяем Dockerfile
  - rootless
  - Не используем `latest`
- Проверяем образы
  - CVE
  - Актульные версии библиотек

"Динамика"

- Ограниваем запуск под `root`
  - Выставляем пользователя
- Описываем сетевые политики
- Оптимизируем привилегии
- Механизмы ядра Linux
  - LSM (AppArmor, SELinux)
  - seccomp

Вместо заключения

- Правильный Dev[Sec]Ops
- Делитесь знаниями о приложениии
- Помогайте в эксплуатации
- Делайте безопасность на ранних этапах (shift-left)



Полезные ссылки

- Латаем огрехи в образах приложений до рантайма, во время и после
- Руководство Бравого Докер Секурити Мастера :latest
- 10 самых распространенных проблем при линтинге Dockerfile'ов
- 10 Kubernetes Security Context, которые необходимо понимать
- Механизмы обеспечения повышенной безопасности контейнеров в Linux

Спасибо за внимание!

site: luntry.ru

tg: @rusdacent

channel: @tech_b0lt_Genona

Вопросы?