



OWASP

Open Web Application
Security Project

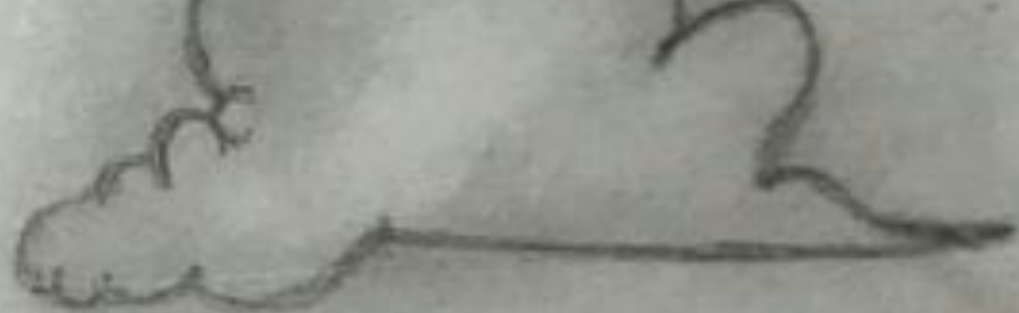
App security in current era

Ajit Dhumale

Director of Engineering, Qualys

OWASP Pune Chapter Meet - 8 June 2019

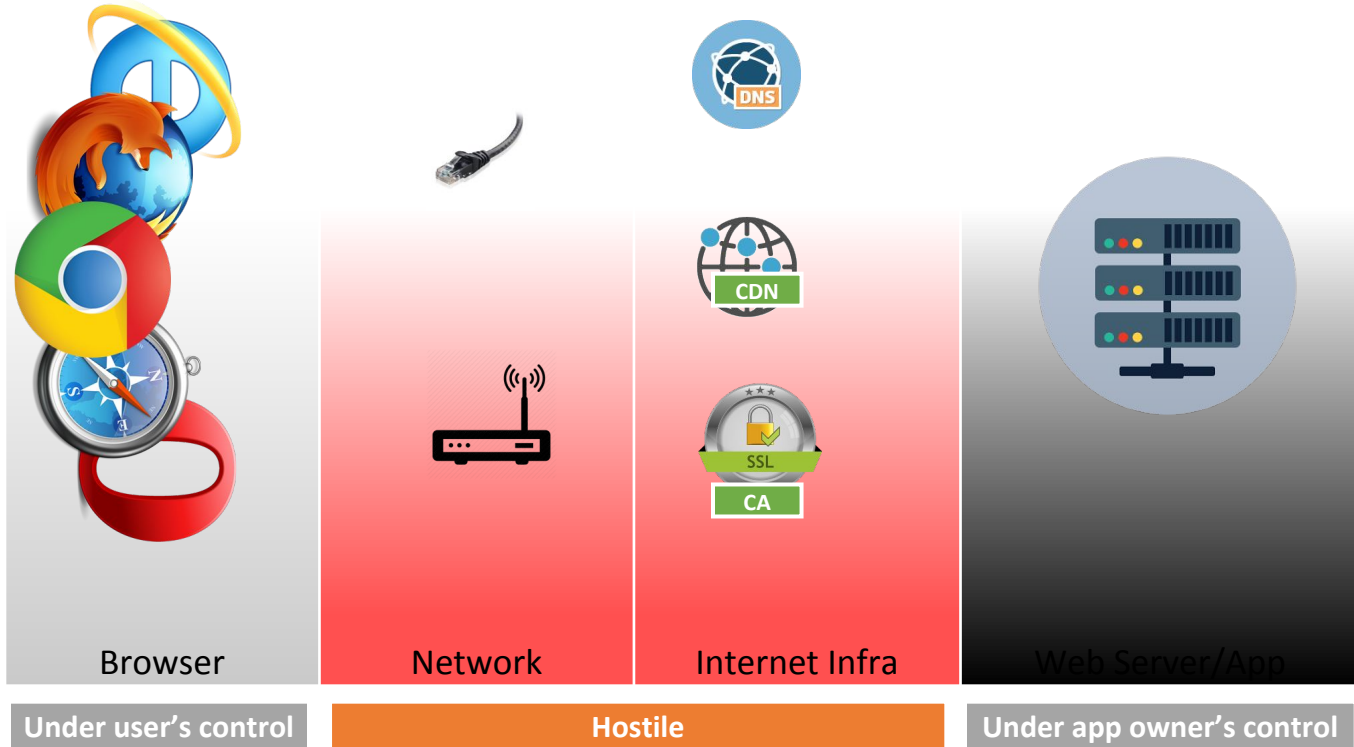
DADDY, WHAT ARE
CLOUDS MADE OF?



LINUX SERVERS,
MOSTLY

What are clouds made of?

Web App Access Eco System



Changes over time : technology



$$f(x)$$



Dedicated resources
Static/rigid/manual provisioning

Shared resources
Automated/elastic provisioning

Change over time : app availability

9 to 5

Maintain Downtime

Upgrade Downtime

Long release cycles

Always On (24/7)

No Downtime

Rolling Upgrades

Frequent Releases

Orchestration: Dynamic, auto, elastic provisioning



App 1

App 5

App 2

App 3

App 4

Logical resource pool:
Compute, storage, IO, network, ...

Physical shared resource pool:
Compute, storage, IO, network, ...

Changes over time: service model



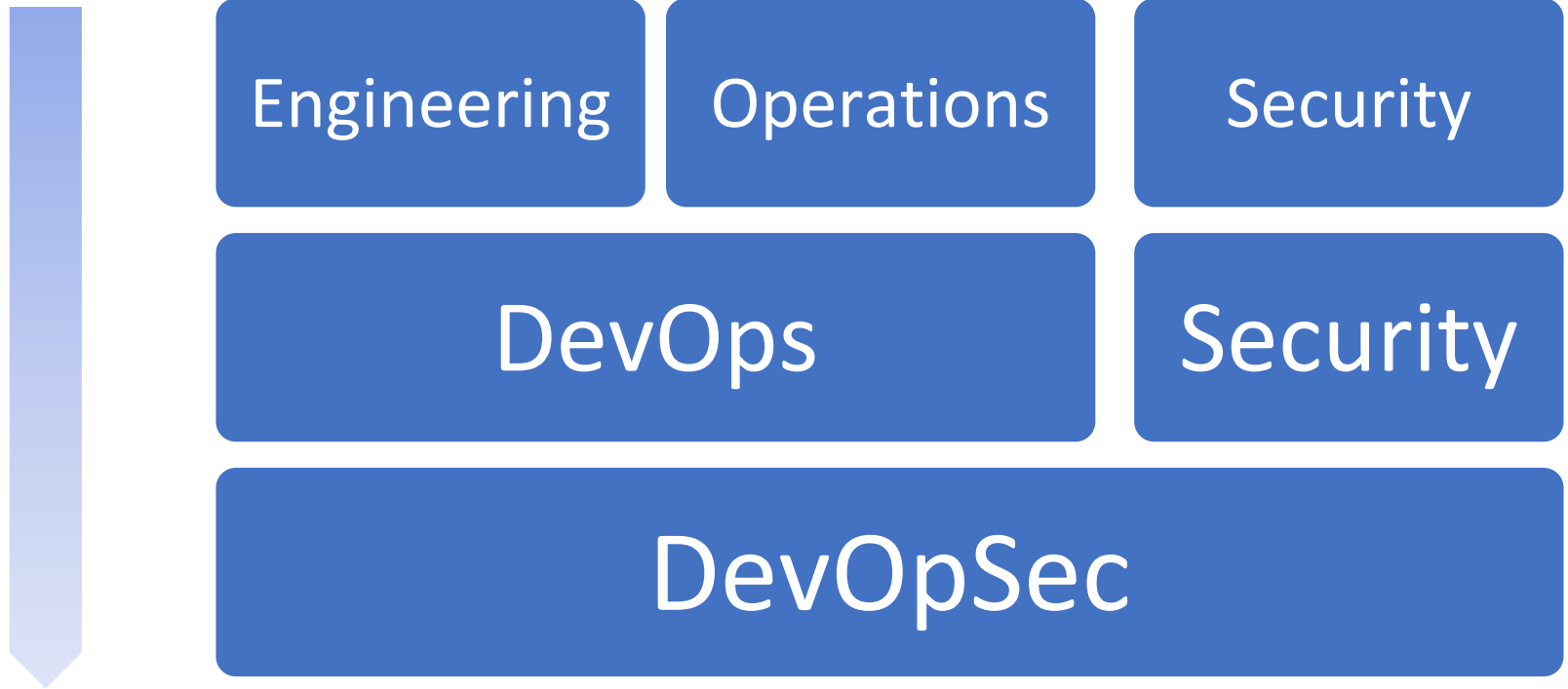
Pets

vs

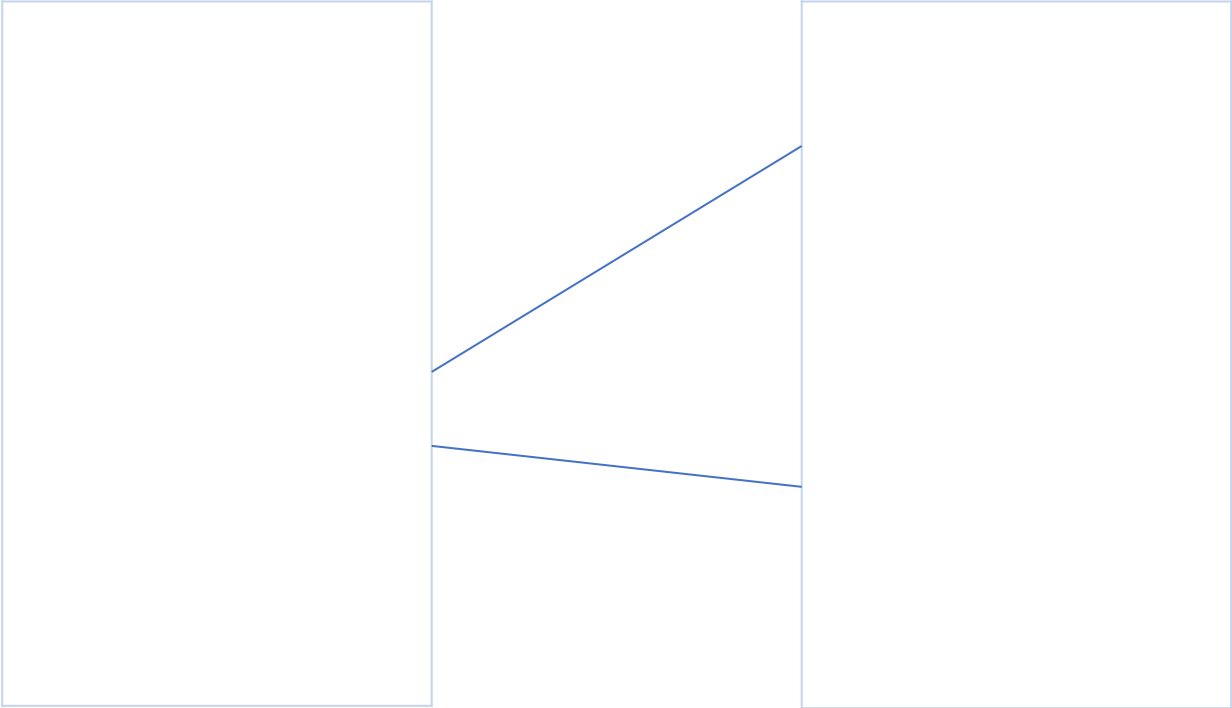


Cattles

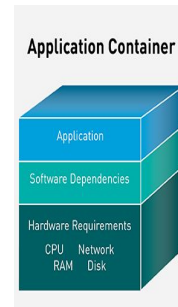
Changes over time : people/roles



Changes over time: app composition



Changes over time: app packaging



Writable layer

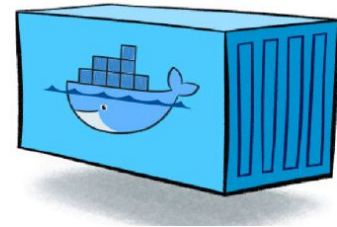
Web App

Spring

Tomcat

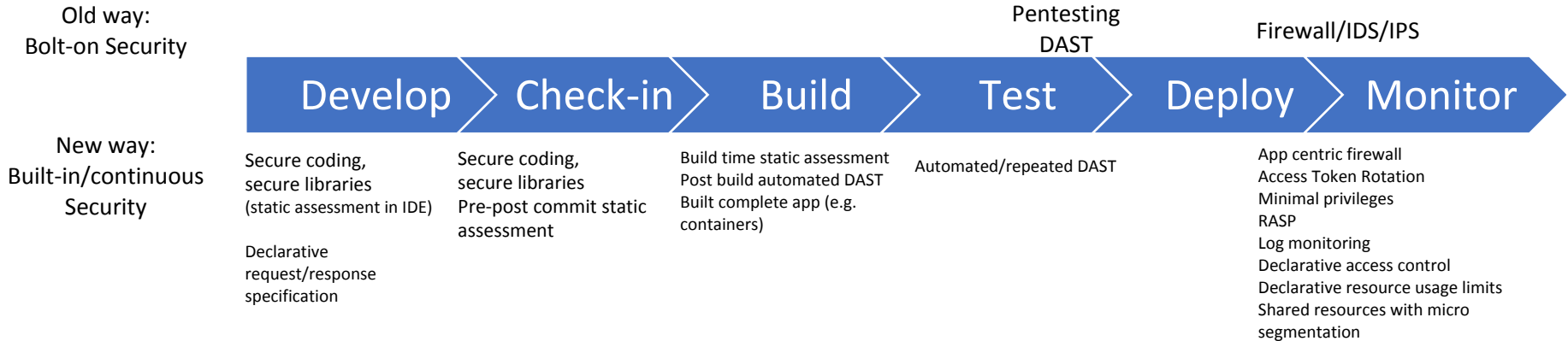
Apache

Base layer: Alpine



Changes over time: Security

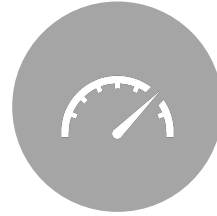
Bolt-on vs built-in/continuous



Pre-requisites for shiftleft and continuous security



Automation



Speed



Accuracy



Declarative specification

Built-in/Continuous/DevOpSec

- IDE
 - Continuous source code scanning (may be as you type)
 - Monitor 3rd party components at inclusion time
- CI/CD
 - Continuous monitoring of vetted/approved dependencies
 - Commit time static scanning
 - Build time security
 - Static assessment
 - SCA
 - Dynamic scanning
- Runtime
 - Dynamic secret management
 - App centric firewall
 - RASP
 - Runtime instrumentation
 - Monitoring
 - Profiling

DevOpSec



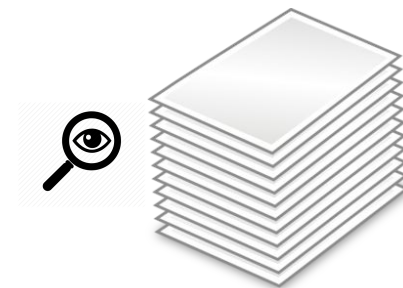


SCA: Software Composition Analysis

- Why its important lately
 - More then 90% of the code in modern apps is 3rd party open source libraries/frameworks
 - Securities issues in 3rd party open source components are known to the world
 - Exploitation often does not need app specific knowledge
 - Exploits become available on internet
 - Easy to launch attack on large number of targets
 - Explosion in CVEs declaration in widely used software components
 - STRUTS RCEs
 - Wordpress RCEs
 - WebLogic RCEs
 - ...

SCA: Software Composition Analysis

- Create inventory/BoM of all 3rd party components used by the app
- Check which components have known vulnerability
- Issues/concerns
 - Coverage: Not all vulnerability have CVEs
 - Noise/FPs: App may be using vulnerable library but may never be calling vulnerable function
- Shiftleft: Integrate SCA in IDE, Artifactory, CI/CD
- Continuous: Detect production apps affected by new vulns



RASP: Runtime Security

- Runtime Instrumentation
 - Agent
 - Built-in the app
 - Instrumented runtime
- Monitor code flow, function calls, system access
- Log/block undesired behavior
 - App specific tuning
- Concerns: Performance overhead, undesired side effects (DoS by FP)


Dynamic secret management

- In pets era:
 - Manual provision of secrets
- In cattle era:
 - Challenge:
 - How to securely make secretes (DB password, API tokens, private key, ...) to dynamically provisioned ephemeral app processes/containers/micro services
 - Risk
 - Secrets sprawl
 - Secrets leakage (via github, ...)
 - Solution
 - Vault, secure introduction (SI) and dynamic tokens



Declarative security automated security

- Declarative network topology
 - Auto generate network access rules
- Declarative request/response
 - Auto generate app firewall rules/filters
- Declarative quota limits
 - Prevent resource exhaustion
- Micro compartments
 - Containment in case of compromise (warrants minimal privileges)



I HAVE CONCERNS
ABOUT THIS CYBERSECURITY
CANDIDATE.

ANYTHING IN
PARTICULAR?

HE'S
AVAILABLE.

We are hiring

<https://www.qualys.com/careers/>

Dev, QA, Support, Ops, Security

WHAT? HOW? WHY? WHO? WHEN? WHERE?
WHO? WHO? WHO? WHO? WHO? WHO? WHO? WHO?
HOW? HOW? WHEN? WHERE?
WHERE? WHERE? WHOSE? WHOSE? WHOSE? WHOSE? WHOSE? WHOSE?
WHICH? WHICH? WHICH? WHICH? WHICH? WHICH? WHICH? WHICH?
HOW? HOW? HOW? HOW? HOW? HOW? HOW? HOW?
WHAT? HOW? WHY? WHO? WHERE? WHAT? HOW?
WHO? WHERE? WHAT? HOW? WHY? HOW? WHERE?
WHO? WHOSE? WHERE? WHAT? HOW?
WHAT? HOW? WHY? WHO? WHERE? WHAT? HOW?
WHO? WHERE? WHAT? HOW? WHY? HOW? WHERE?
WHO? WHOSE? WHERE? WHAT? HOW?
WHAT? HOW? WHY? WHO? WHERE? WHAT? HOW?
WHO? WHERE? WHAT? HOW? WHY? HOW? WHERE?
WHAT? HOW? WHY? WHO? WHERE? WHAT? HOW?
WHO? WHERE? WHAT? HOW? WHY? HOW? WHERE?
WHAT? HOW? WHY? WHO? WHERE? WHAT? HOW?
WHO? WHERE? WHAT? HOW? WHY? HOW? WHERE?

HOW? WHERE? WHO? HOW?
HOW? WHO? WHO? WHO? WHO? WHO? WHO? WHO?
WHO? WHERE? WHAT? HOW? HOW? HOW? HOW?
WHAT? HOW? WHY? WHY? WHY? WHY? WHY? WHY?
WHERE? HOW? HOW? HOW? HOW? HOW? HOW? HOW?
WHAT? WHERE? HOW?

