DevSecOps- Why Should We Embrace It?

Kamal Boolchandani
DevSecOps Specialist

root@presentation:~$ whoami

Kamal Boolchandani

- DevSecOps Specialist Global Payments Pune
- 8 years of Experience in IT
- Cloud Security
- DevSecOps
- IaC Framework
- Tools Integration and Automation
- Cloud Migration

Agenda

1. Cloud Security
2. Challenges for Cloud Security
3. Case Study: Famous Cloud Attacks
4. Misconfiguration in Cloud
5. Why DevSecOps?
6. DevSecOps vs DevOps
7. DevSecOps RoadBlocks
8. DevSecOps Model (Aws)

“Cloud Misconfigurations are by far the biggest threat to cloud security”

– National Security Agency (NSA)

“Cloud Vulnerabilities have grown a whopping 28% since last year, with a 200% increase in cloud accounts offered on the dark web”

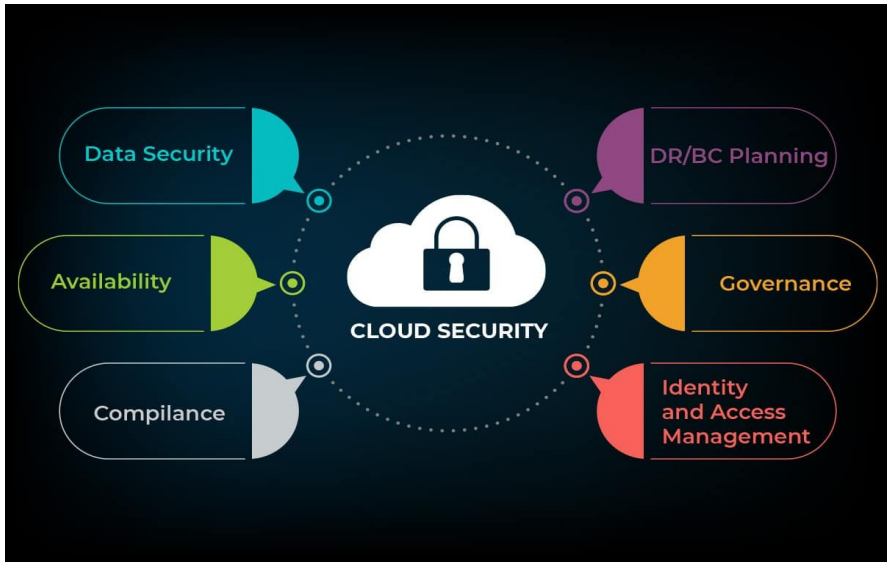
– The 2023 IBM Security X-Force C

"99 percent of all misconfigurations in the public cloud go Unreported"

-**Mcafee**, The IaC Adoption and Risk Report

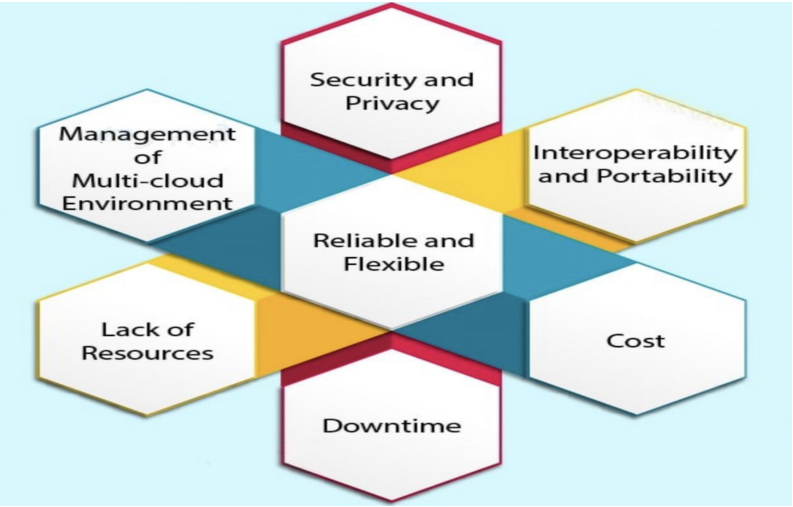
Cloud Security: what makes it different

- Shared Responsibility Model
- Elasticity
- Speed
- Efficient Resource Utilization
- Dissolving Perimeters

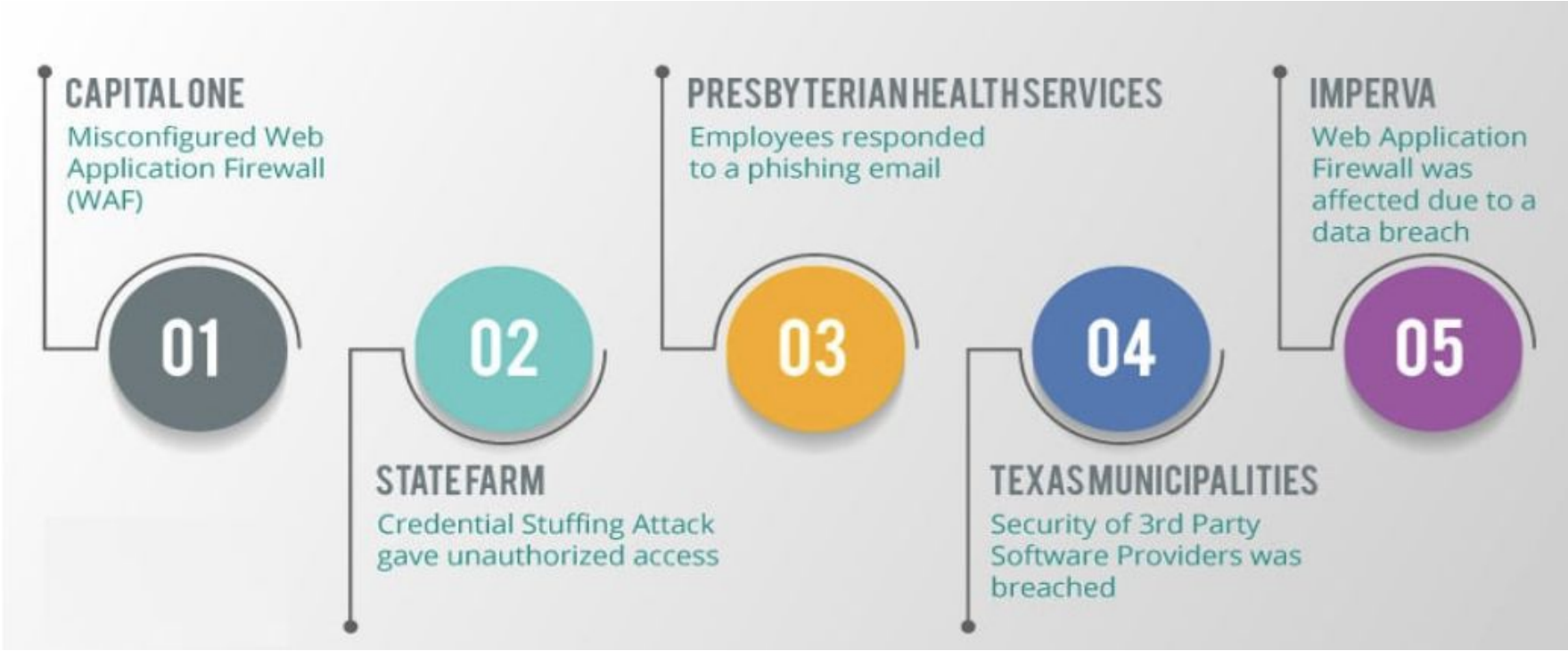


Challenges for Cloud Security

- Increase attack surface
- Lack of visibility
- Dynamic nature (Workloads)
- Granular Access Management
- Complex Environment(Hybrid or multi cloud)



Case Study : Is Cloud Really Less Secure?



Case Study:Dincloud

Misconfiguration: Common and Costly affair

Misconfiguration of cloud infrastructure is a leading contributor to data breaches. If an organization's cloud environment is not configured properly, critical business data and applications may become susceptible to an attack. Misconfiguration is by far the biggest security threat in cloud environment.

Some of the common Misconfigurations are:

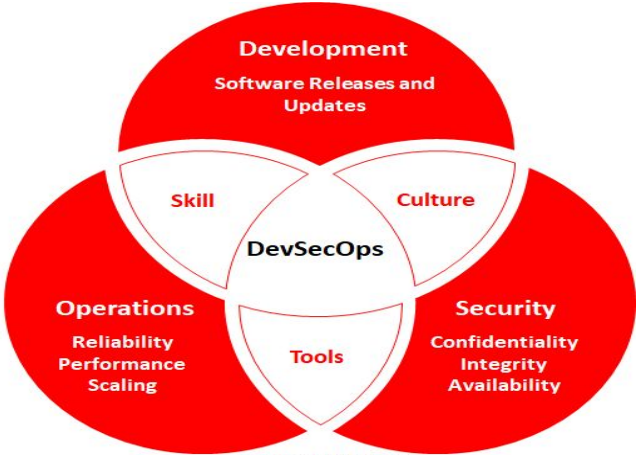
- **IAM Policy Errors**
- **Inappropriate Security Group**
- **Deployment Pipeline Misconfigurations**
- **Backup Storage Location Misconfigurations**
- **Insecure APIs**

DevSecOps

DevSecOps integrates application and infrastructure **security** seamlessly into Agile and DevOps processes and tools. It addresses security issues as they emerge, when they're **easier**, **faster**, and **less expensive** to fix.

Effort to strive for “**Secure by Default**”

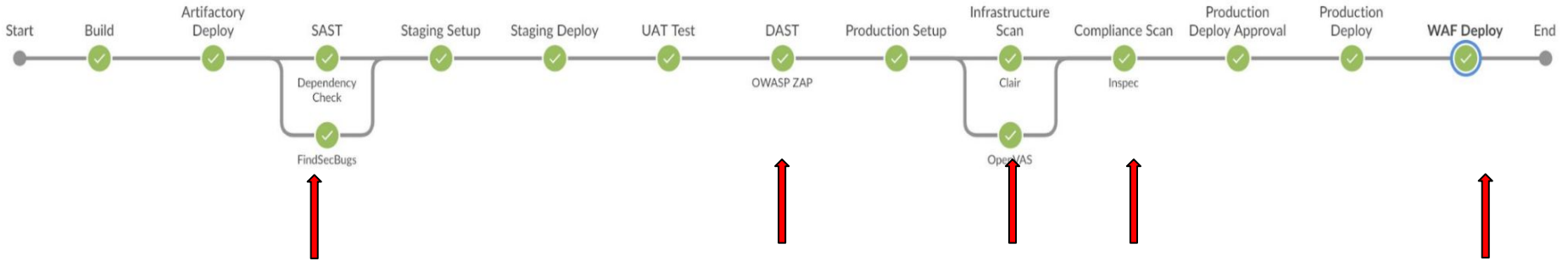
- Integrate Security via tools
- Create Security as Code culture
- Promote cross skilling



DevSecOps Vs DevOps

DevOps

DevSecOps



Static Application Security Testing

Dynamic Application Security Testing

Stages in DevSecOps Pipeline

Stage -1

- Pre-Commit Hooks
- IDE Plugins
- Secrets Management

Stage -2

- Software Composition Analyses
- SAST

Stage -3

- DAST

Stage -4

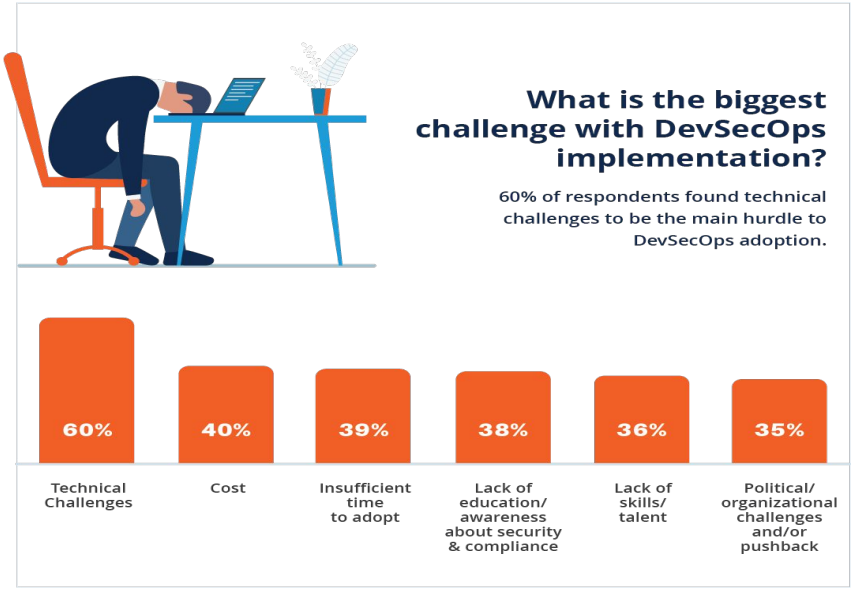
- Infrastructure As a code

Stage -5

- Compliance As a Code

RoadBlocks For DevSecOps

- The cultural shift
- Insufficient skill sets
- Complex tool integrations
- Traditional security tools vs. agile DevOps



Reference- State of DevSecOps 2023

DevSecOps Implementation Steps

- Classify Workloads by segment and deployment models
- Define standards by control area and classification
- Implement security as a code through automation
- Build an support operating model protections.

DevSecOps Model

Service Review

Automated Compliance evaluation and Remediation

Define Security controls

Appropriate Security controls in place

Security Logging

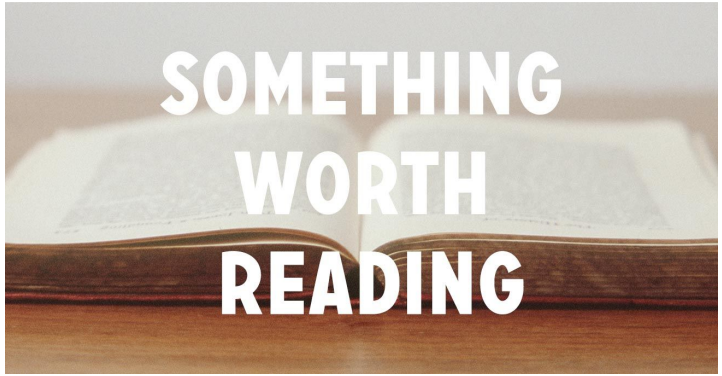
Deploy Services and Infra across accounts

Secure the *Security*

- Did we secure the **Security Controls**
- **DevSecOps**: If attacker controls security tools / build chain It has limitless power
- Security role should not circumvent the rules
- Remember “**Trust but Validate**”
- Efficient Detective Mechanism

References:

- What is Cloud Security
- What is Security as a Code
- “Shifting Left” Best Practices
- Security As a Code
- DevSecOps Overview
- Top 10 Cloud Security Challenges
- Mitigating DevSecOps Challenges
- Misconfiguration - A Hidden Threat



Thank **You**

For Questions and Queries

Kamalboolchandani@gmail.com

https://www.linkedin.com/in/kamal-boolchandani-34737170