Murray Goldschmidt

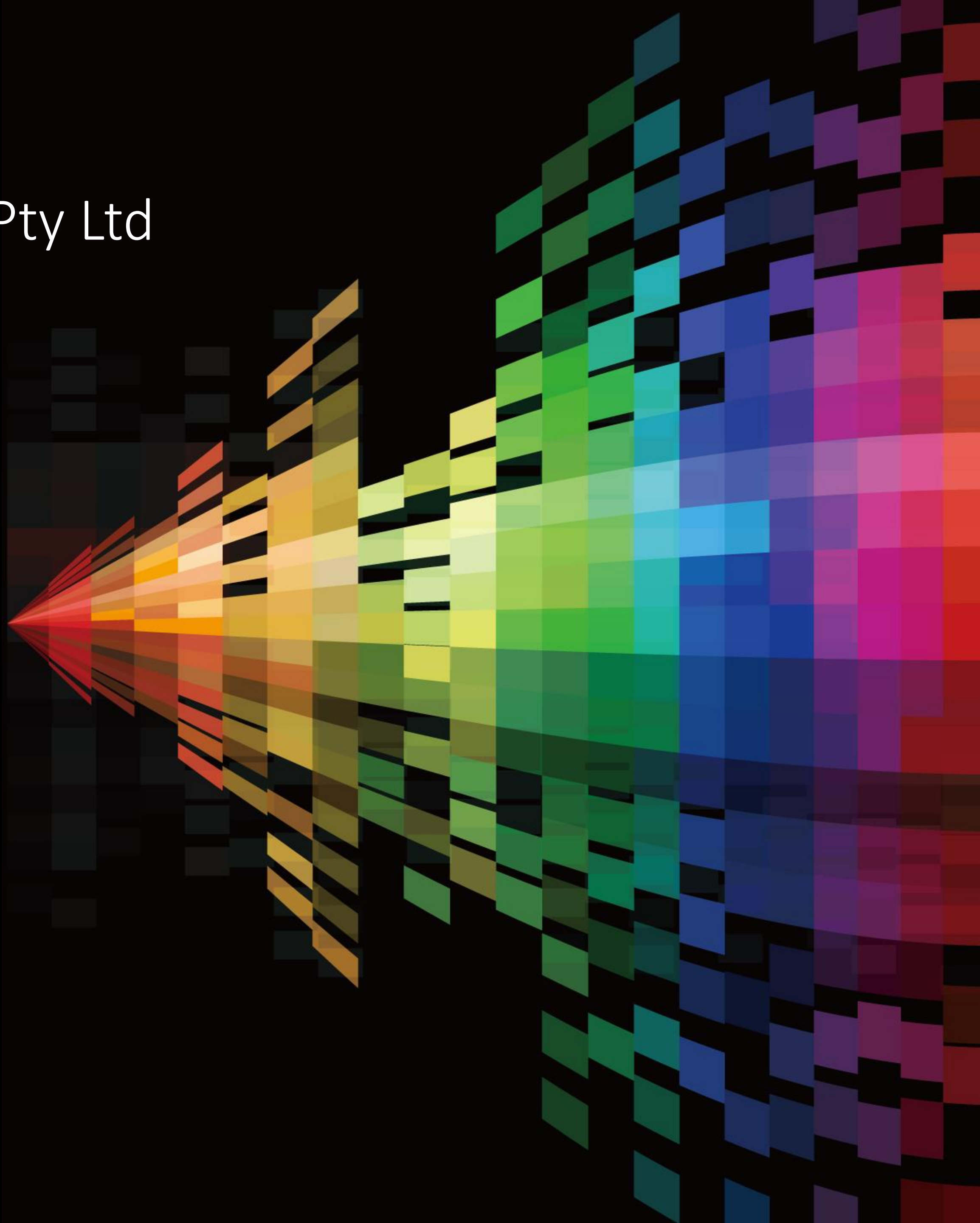
Chief Operating Officer – Sense of Security Pty Ltd

Micro Services, Containers and Serverless PaaS Web Apps? How safe are you?

INNOVATE | DISRUPT | CHANGE

AUSTRALIAN CYBER CONFERENCE

AISA



AGENDA

1	Serverless, Microservices and Container Security	4	CI/CD Integration for Automated Security
2	Key Implications for Penetration Testing Programs		End to End Vulnerability Management
3	Key Security features for Container Deployments		Continuous Monitoring, Governance & Compliance Reporting

Are Containers As Good as it Gets?

Cloud containers are designed to virtualize a single application

*** Modified *** https://searchcloudsecurity.techtarget.com/feature/Cloud-containers-what-they-are-and-how-they-work

As Good as it Gets?

e.g., you have a MySQL container and that's all it does, provide a virtual instance of that application.

*** Modified *** https://searchcloudsecurity.techtarget.com/feature/Cloud-containers-what-they-are-and-how-they-work

As Good as it Gets?

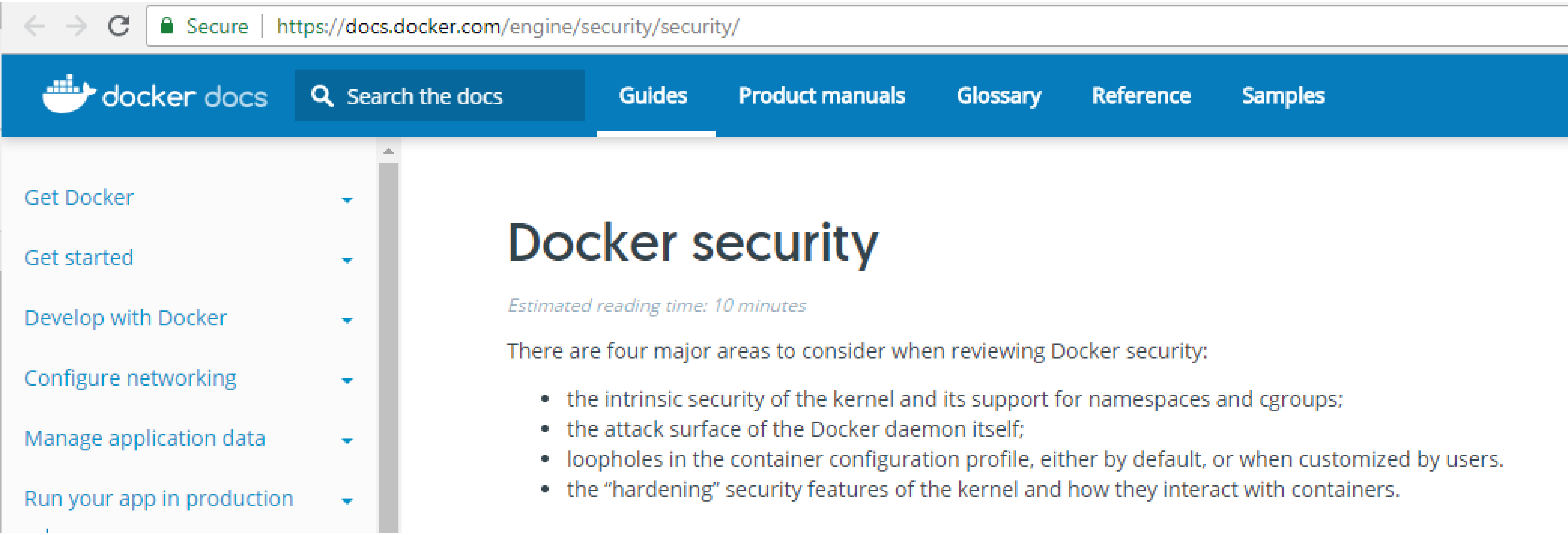
Containers ***SHOULD*** create an *isolation boundary* at the application level rather than at the server level.

*** Modified *** https://searchcloudsecurity.techtarget.com/feature/Cloud-containers-what-they-are-and-how-they-work

As Good as it Gets?

This isolation ***SHOULD*** mean that if anything goes wrong in that single container (e.g., excessive consumption of resources by a process) it only affects that individual container and not the whole VM or whole server.

*** Modified *** https://searchcloudsecurity.techtarget.com/feature/Cloud-containers-what-they-are-and-how-they-work



Docker security

Estimated reading time: 10 minutes

There are four major areas to consider when reviewing Docker security:

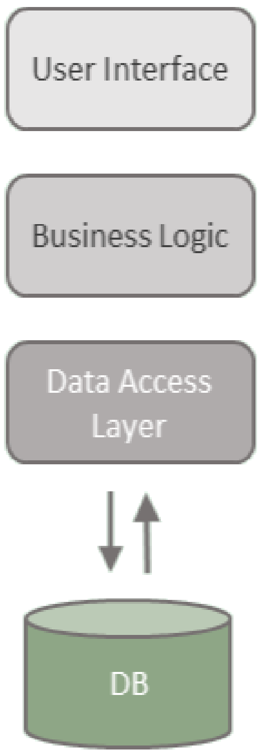
- the intrinsic security of the kernel and its support for namespaces and cgroups;
- the attack surface of the Docker daemon itself;
- loopholes in the container configuration profile, either by default, or when customized by users.
- the "hardening" security features of the kernel and how they interact with containers.

Container Security – Tech Neutral

Security Requirements	Addressed By
Intrinsic Security of the Kernel	Supply Chain Risk Mgt/ Vuln Mgt/ CaaS
Attack Surface Reduction	Hardening/Config Mgt/Vuln Mgt
Container Configuration	Configuration Management
Hardening of the Kernel and how it interacts with Containers	Hardening

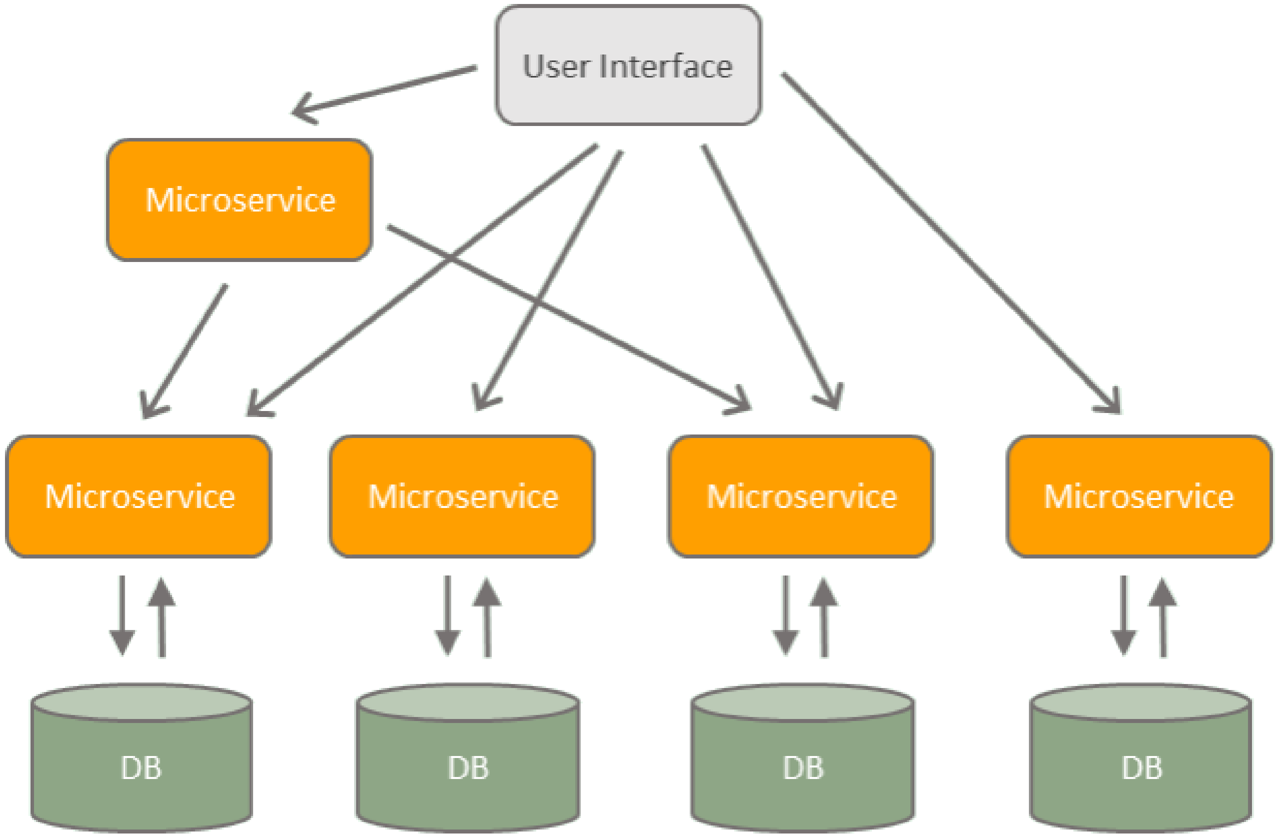
Monolithic vs Microservices Architecture

MONOLITHIC ARCHITECTURE

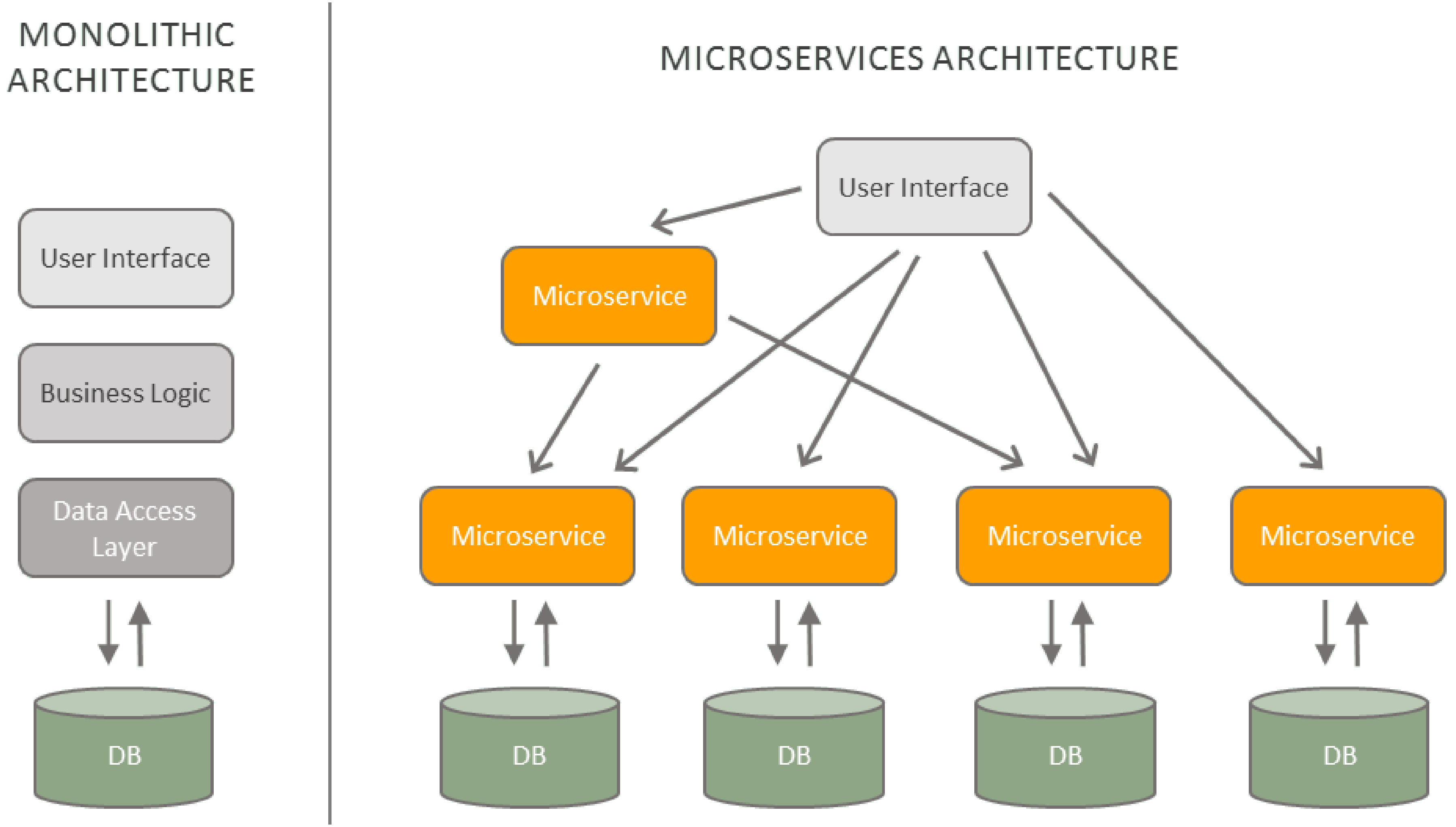


Monolithic vs Microservices Architecture

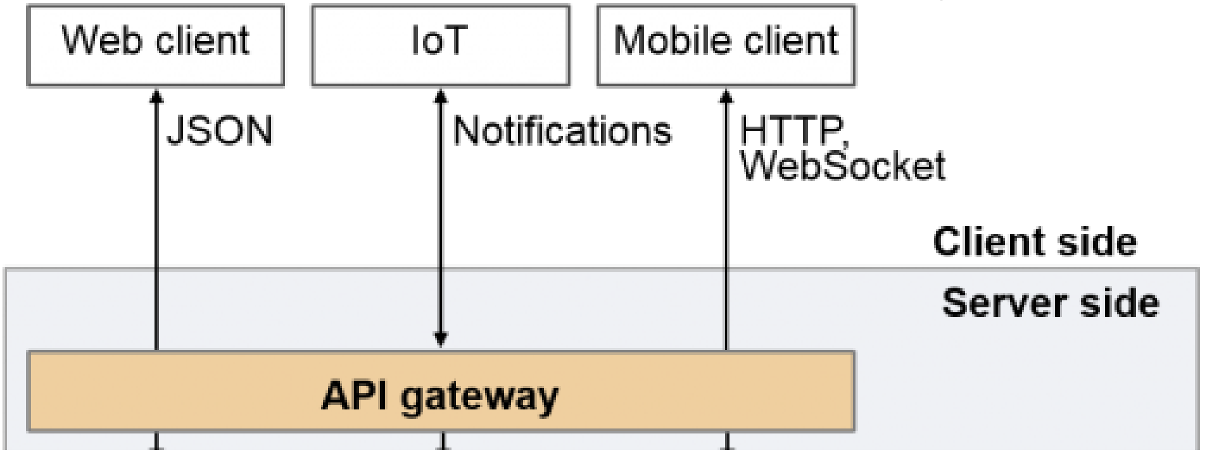
MICROSERVICES ARCHITECTURE



Monolithic vs Microservices Architecture

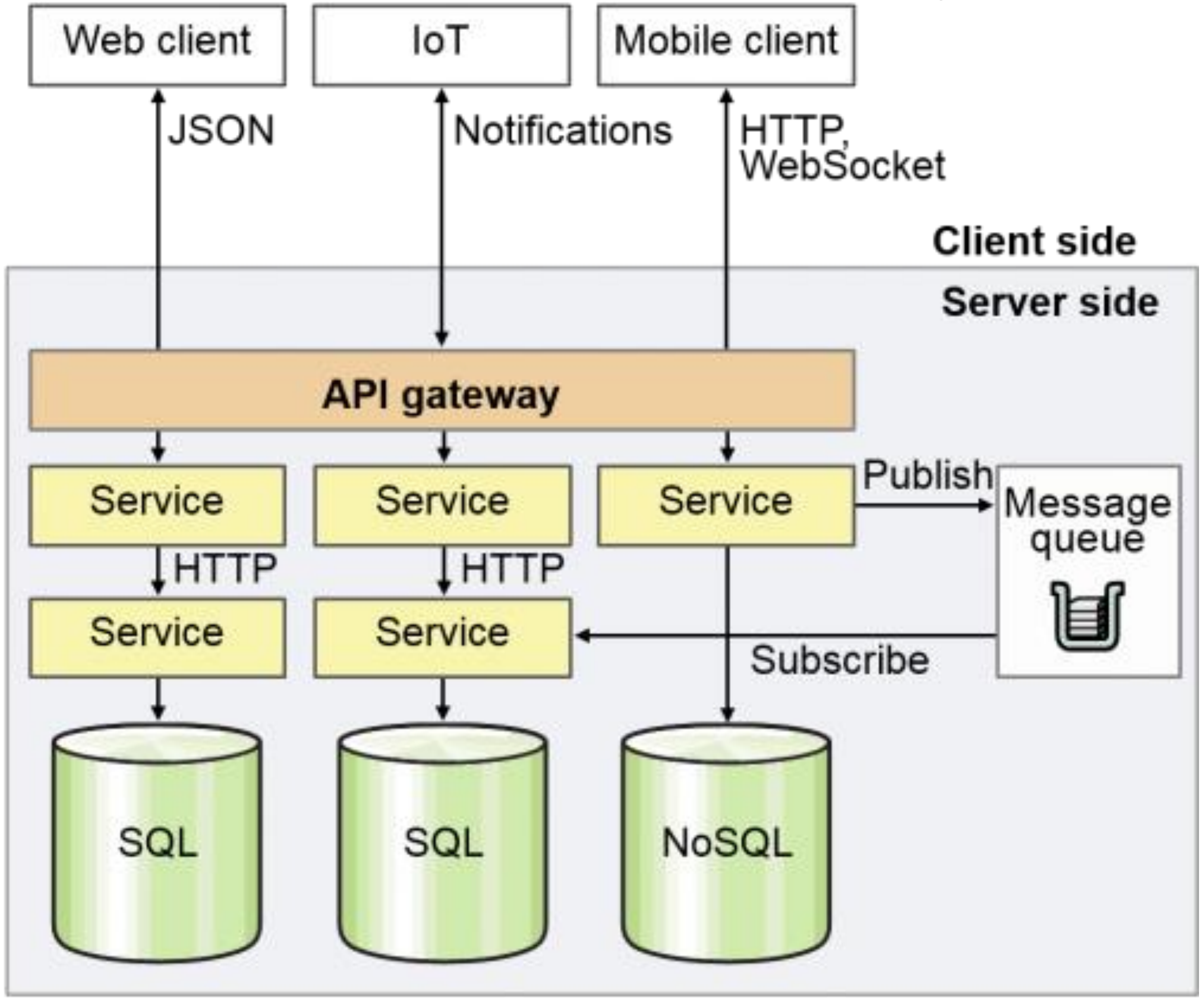


Monolithic vs Micro Services (API Centric)



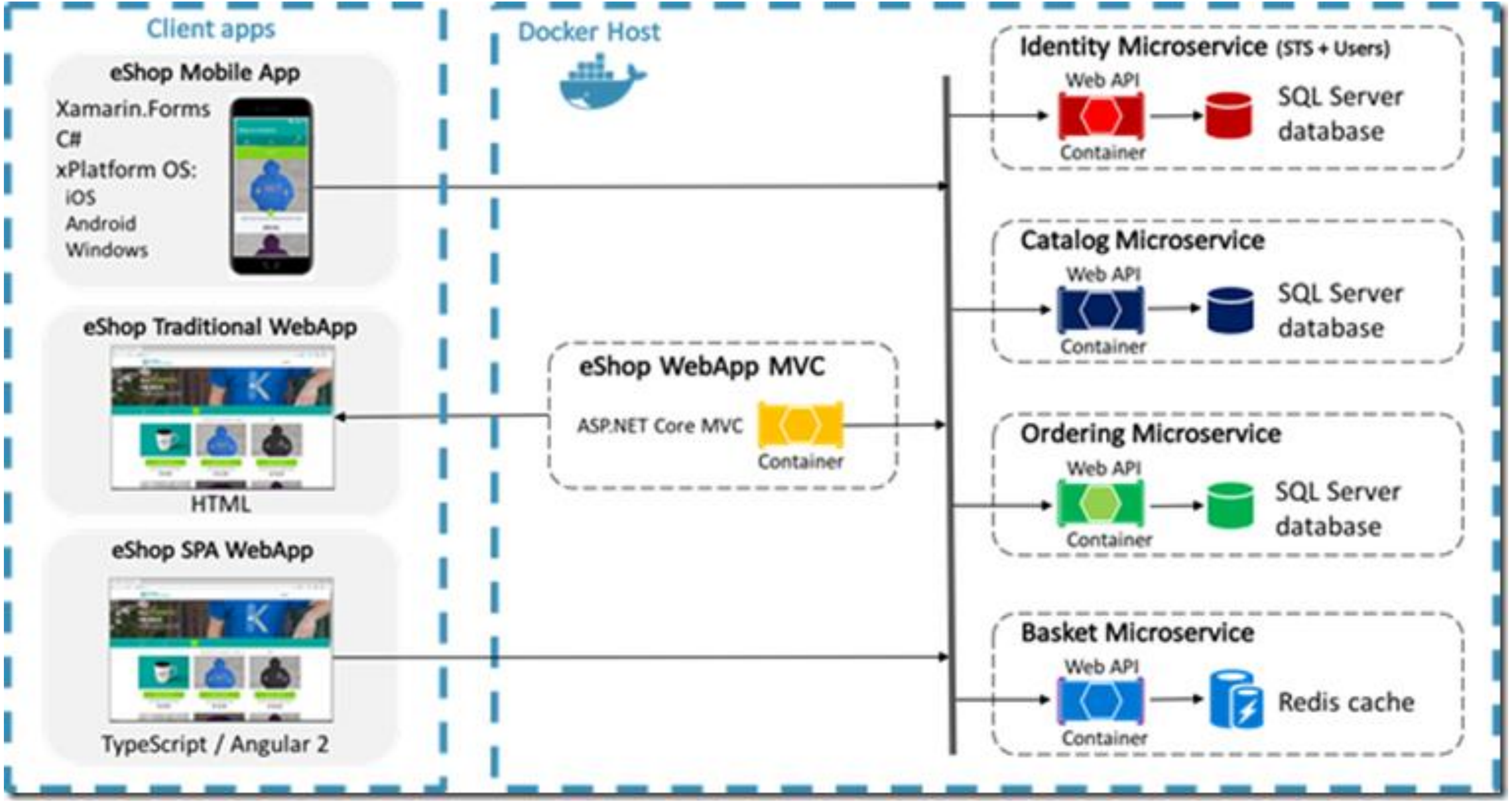
https://developer.ibm.com/courses/monolithic-architecture-versus-microservices-architecture-dwc024/

Monolithic vs Micro Services (API Centric)

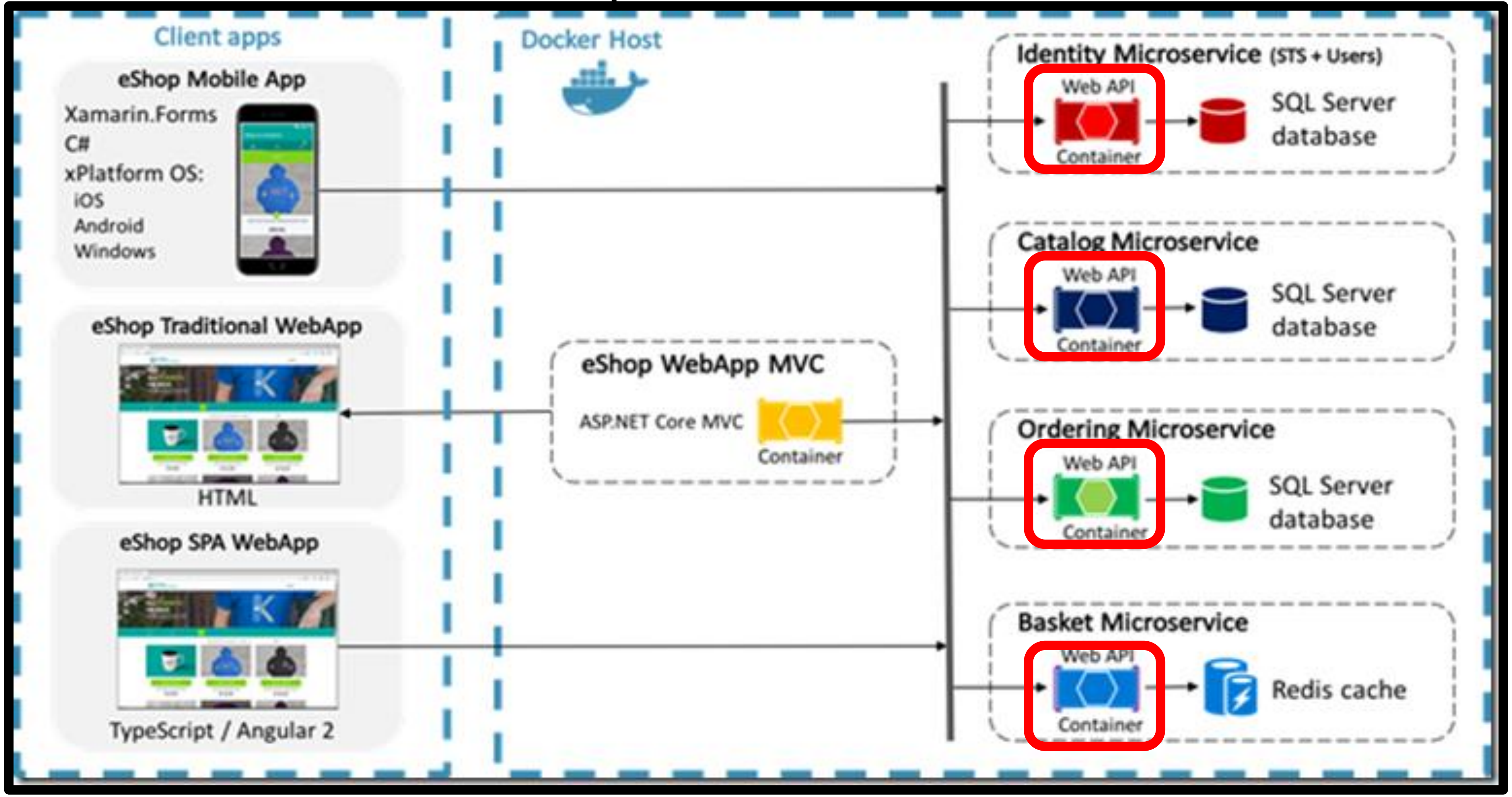


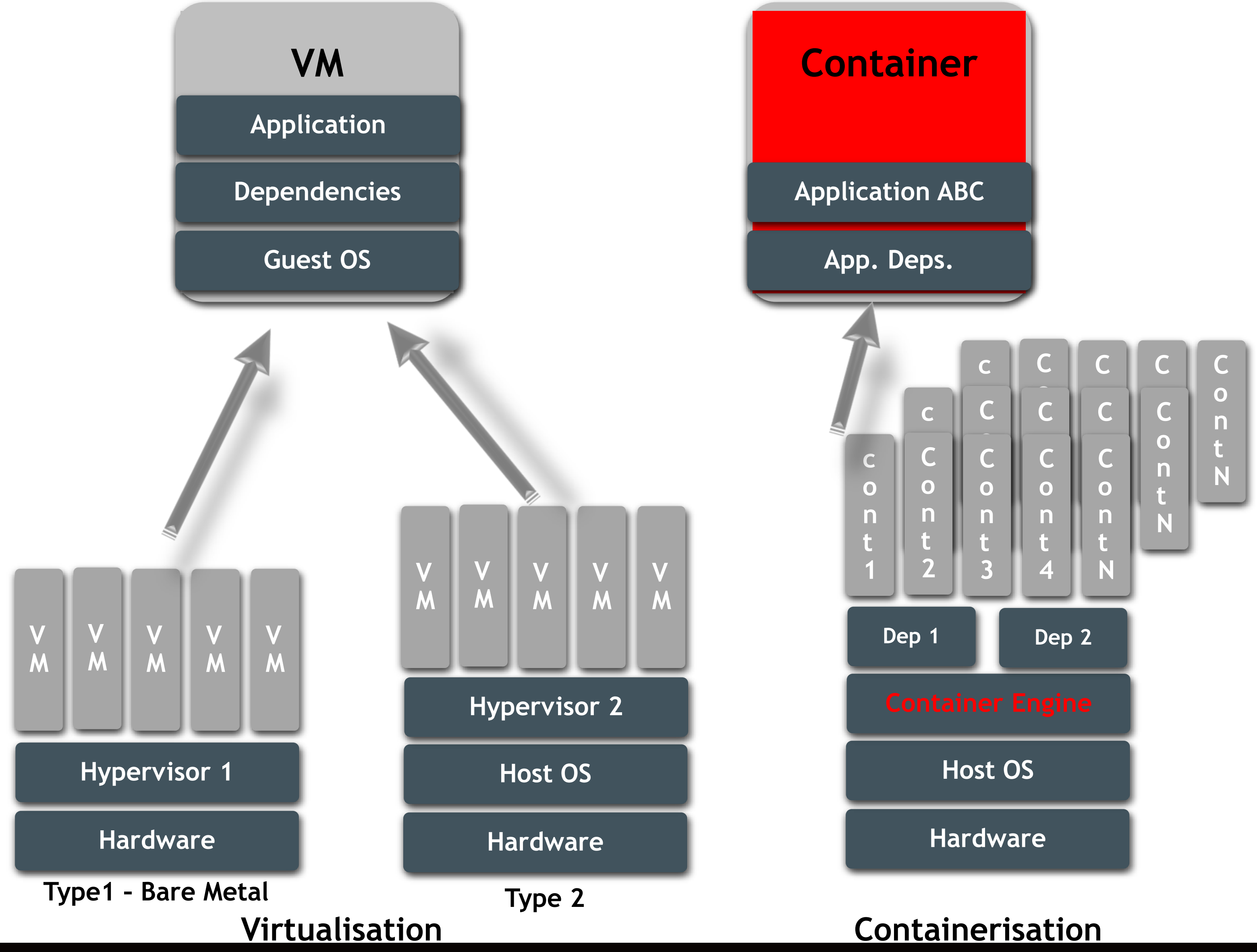
https://developer.ibm.com/courses/monolithic-architecture-versus-microservices-architecture-dwc024/

Example: Microsoft eShop Reference Architecture



Example: Microsoft eShop Reference Architecture



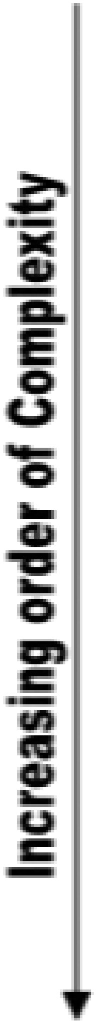


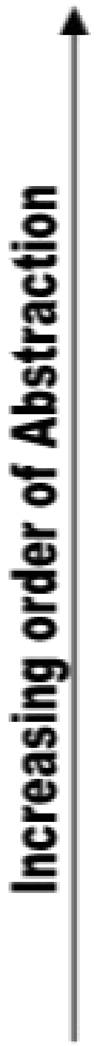
Type1 - Bare Metal

Type 2

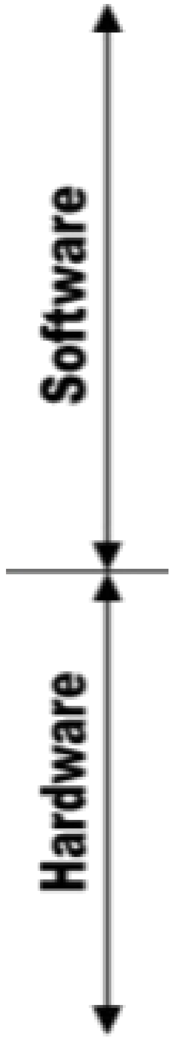
Virtualisation

Containerisation


Increasing order of Complexity



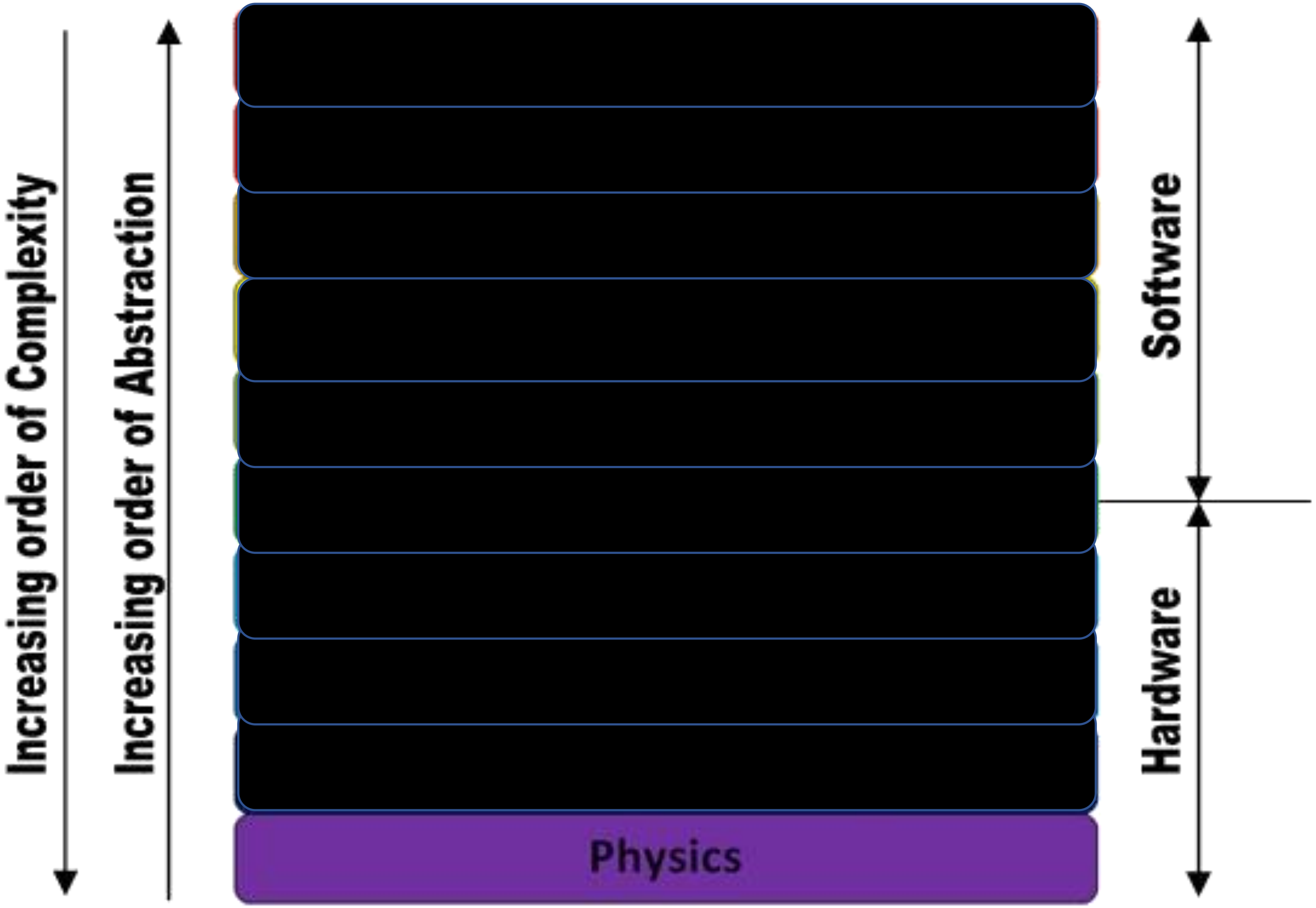
Increasing order of Abstraction



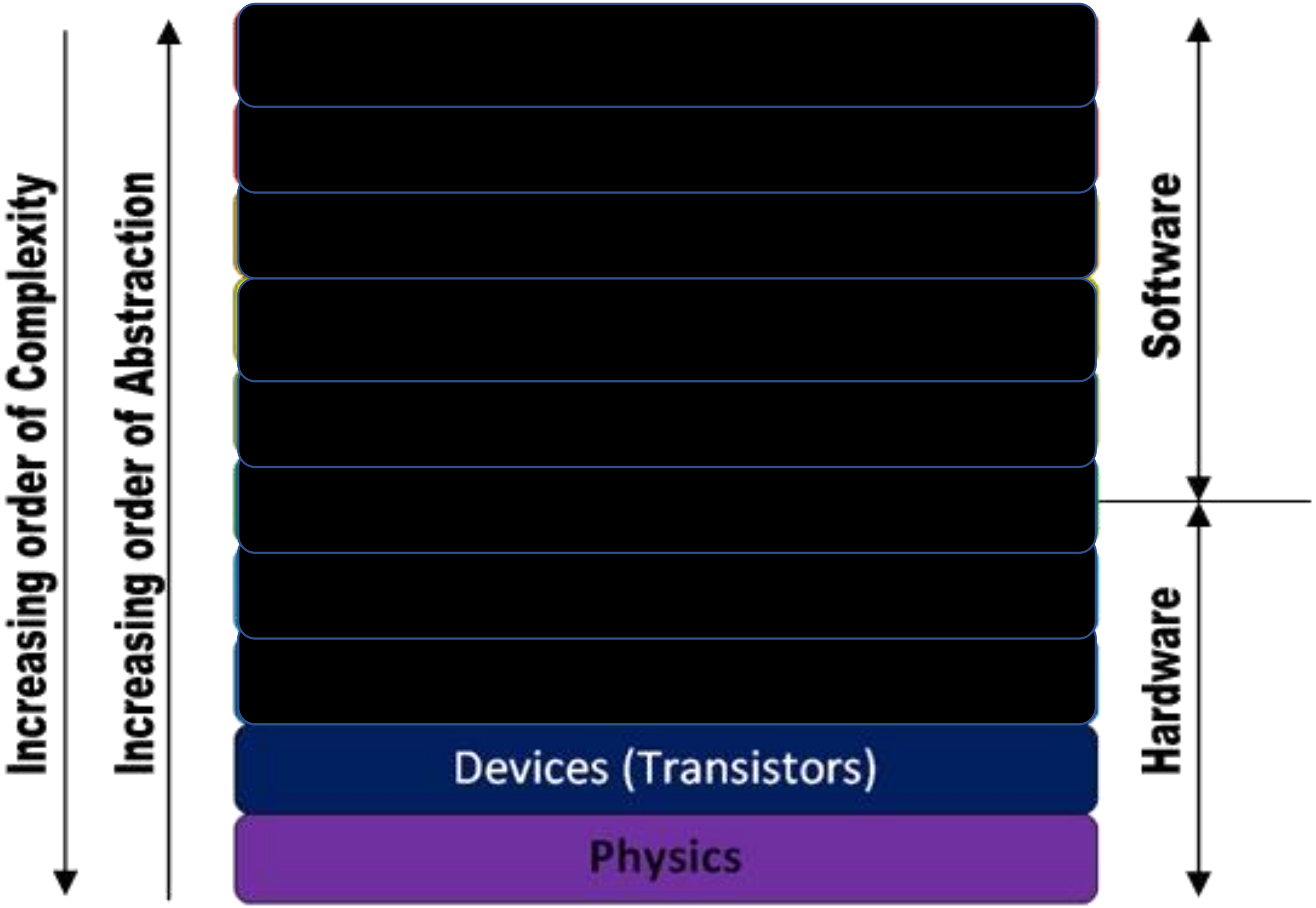
Layers of Abstraction

Increasing order of Complexity
Increasing order of Abstraction
Software
Hardware

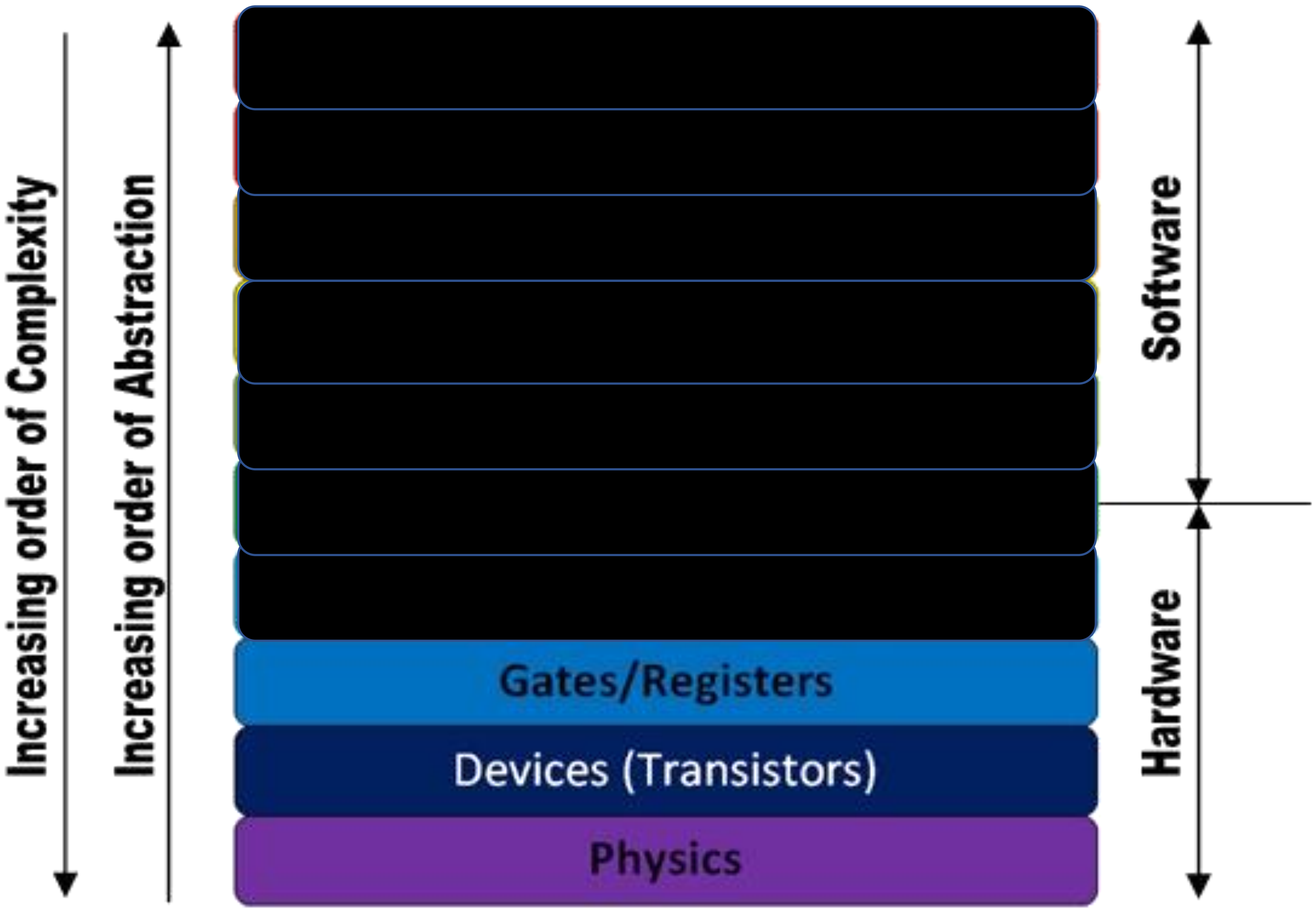
Layers of Abstraction


Increasing order of Complexity
Increasing order of Abstraction
[redacted]
Physics
Software
Hardware

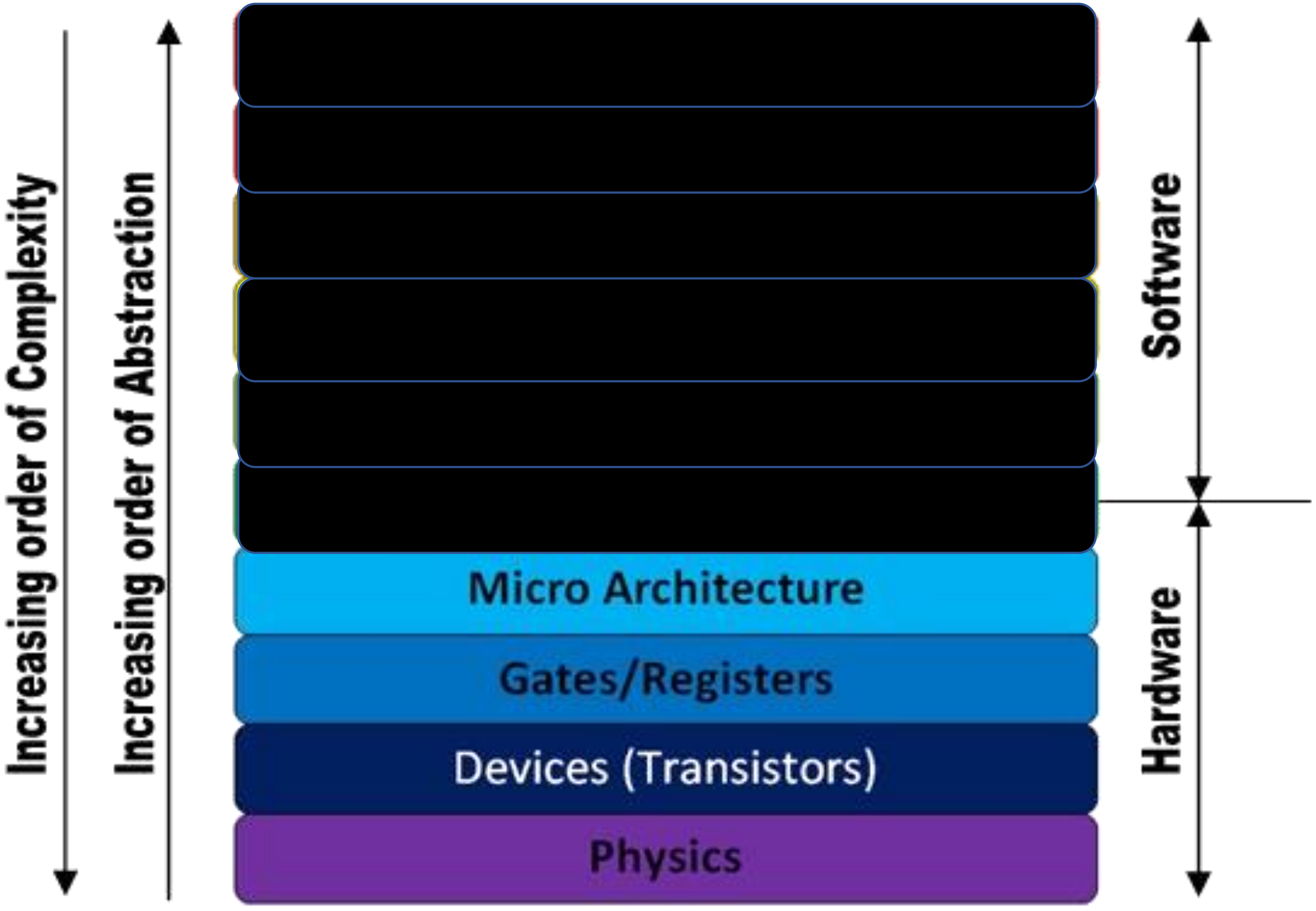
Layers of Abstraction


Increasing order of Complexity
Increasing order of Abstraction
[redacted]
Devices (Transistors)
Physics
Software
Hardware

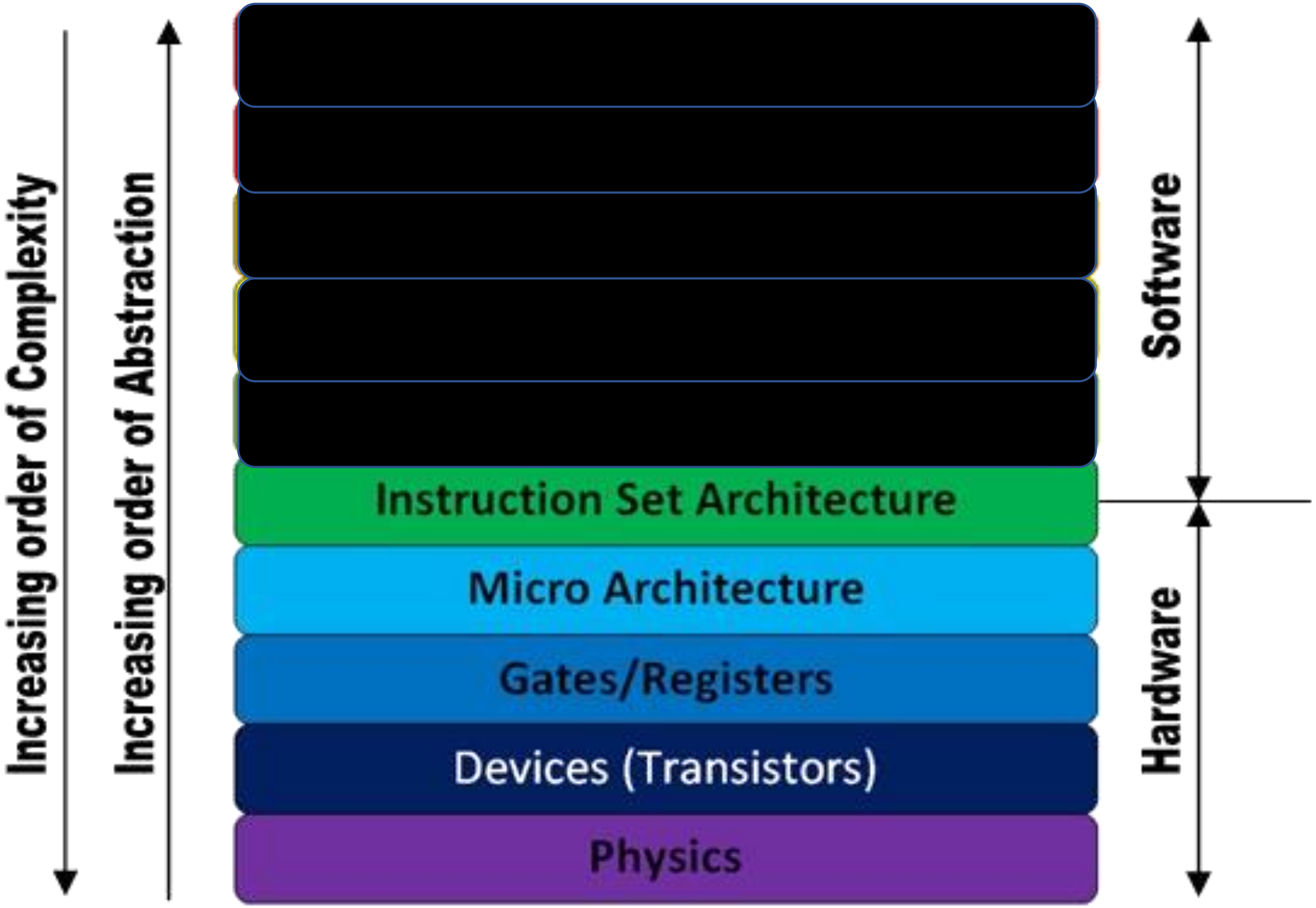
Layers of Abstraction



Layers of Abstraction


[redacted]

Layers of Abstraction



[redacted]

Instruction Set Architecture

Micro Architecture

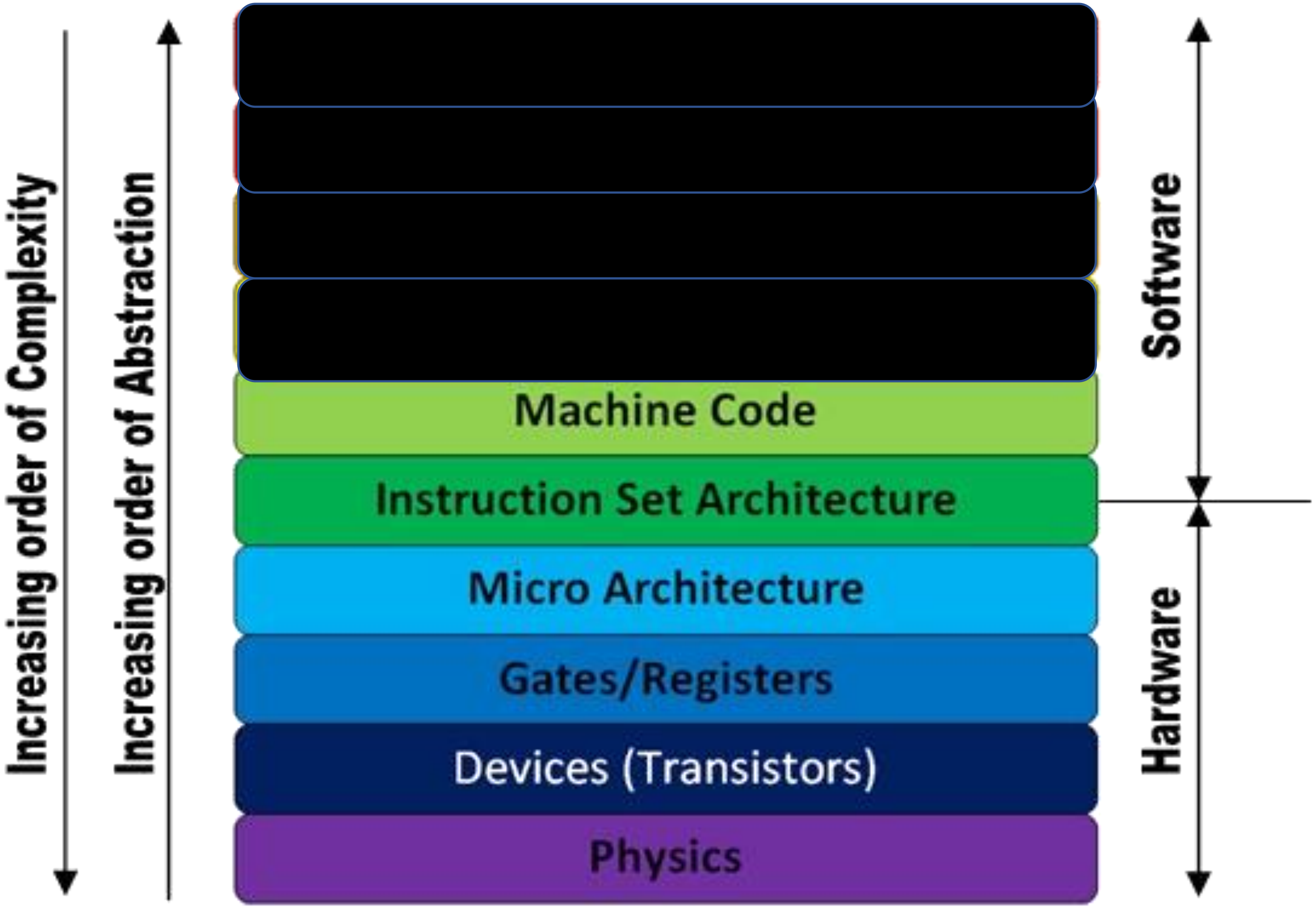
Gates/Registers

Devices (Transistors)

Physics

Increasing order of Complexity
Increasing order of Abstraction
Software
Hardware

Layers of Abstraction



Increasing order of Complexity

Increasing order of Abstraction

[redacted]

[redacted]

[redacted]

[redacted]

Machine Code

Instruction Set Architecture

Micro Architecture

Gates/Registers

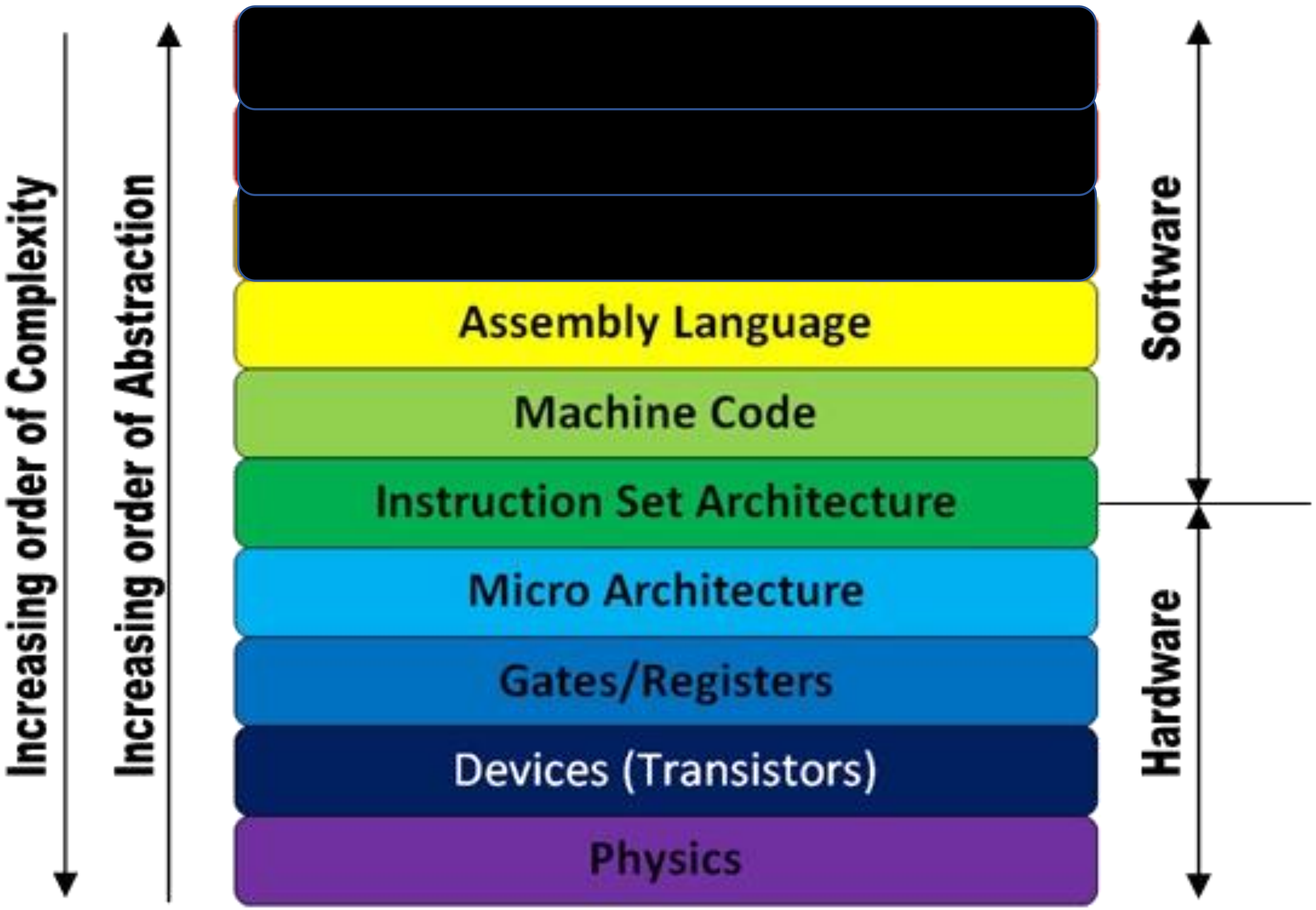
Devices (Transistors)

Physics

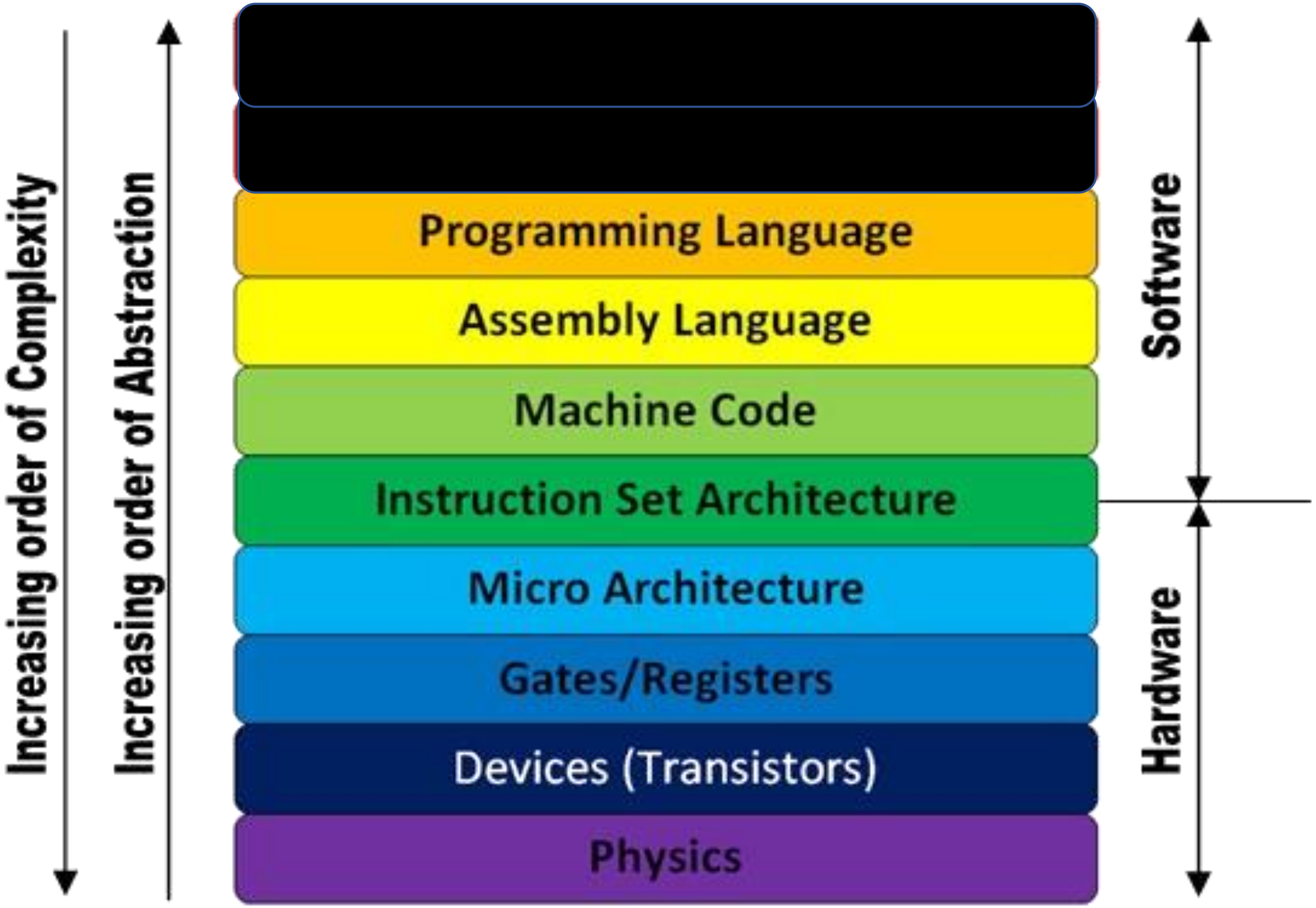
Software

Hardware

Layers of Abstraction


[redacted]
[redacted]
[redacted]

Layers of Abstraction



Increasing order of Complexity

Increasing order of Abstraction

[redacted]

[redacted]

Programming Language

Assembly Language

Machine Code

Instruction Set Architecture

Micro Architecture

Gates/Registers

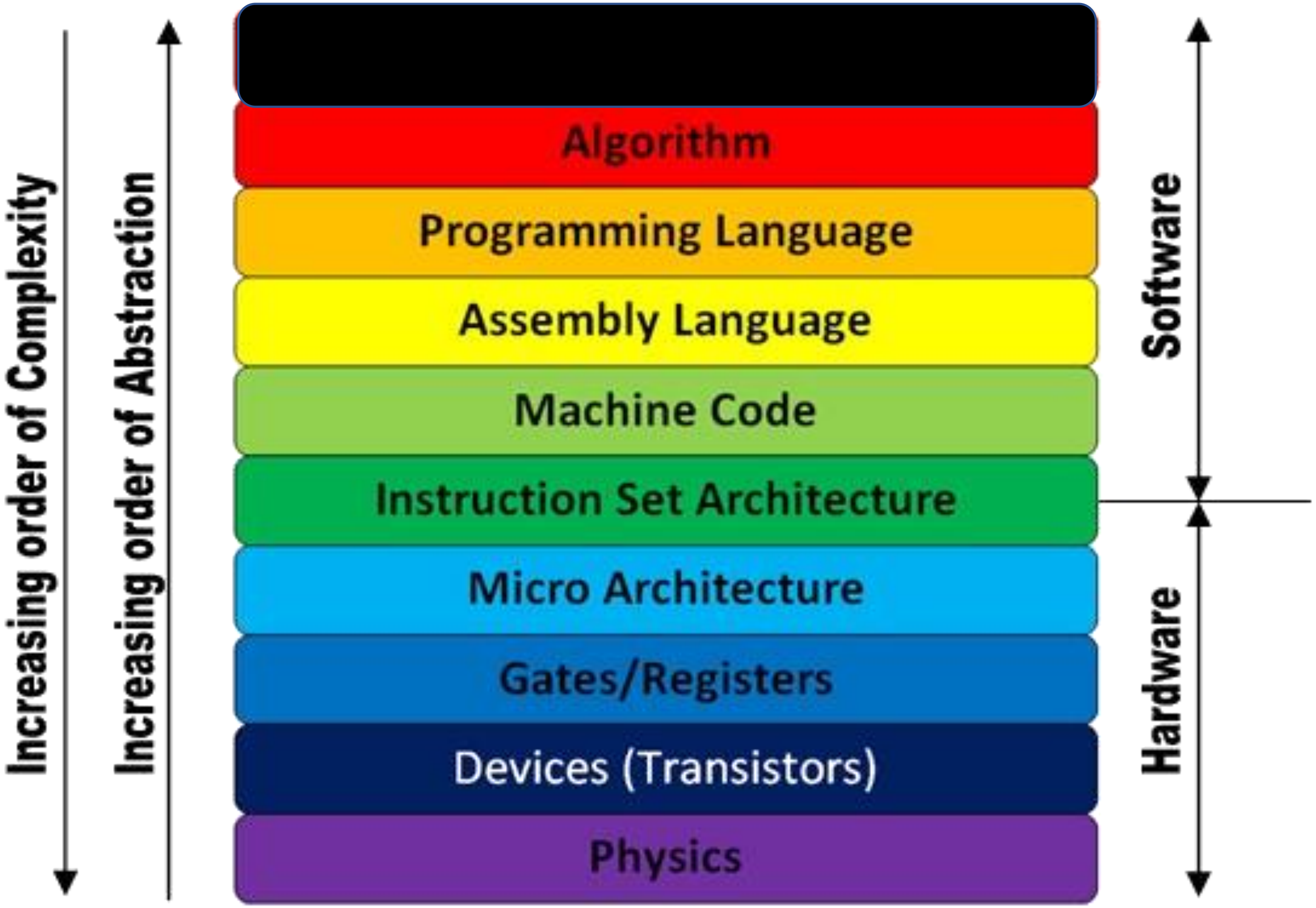
Devices (Transistors)

Physics

Software

Hardware

Layers of Abstraction



Increasing order of Complexity

Increasing order of Abstraction

[redacted]

Algorithm

Programming Language

Assembly Language

Machine Code

Instruction Set Architecture

Micro Architecture

Gates/Registers

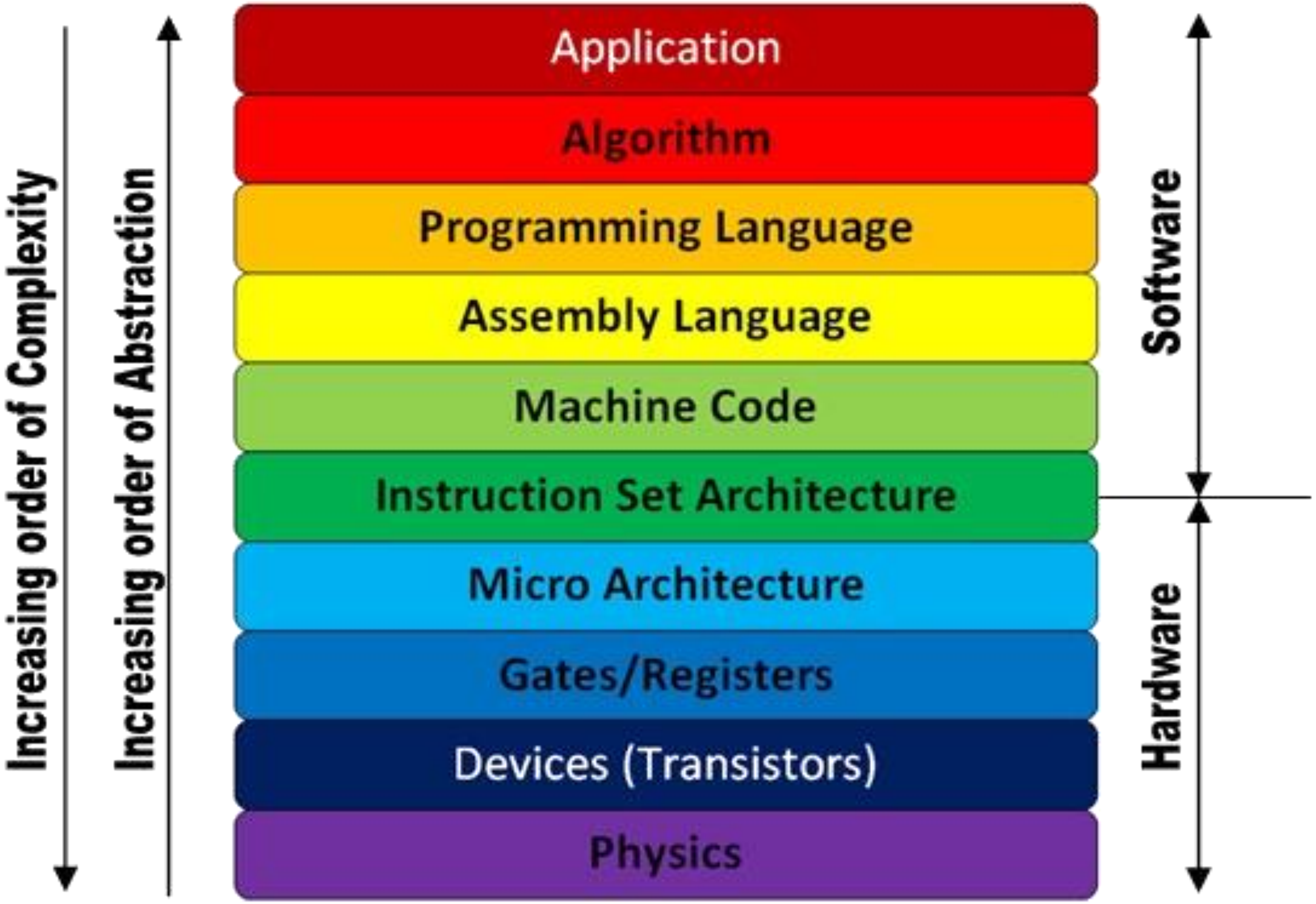
Devices (Transistors)

Physics

Software

Hardware

Layers of Abstraction



Application

Algorithm

Programming Language

Assembly Language

Machine Code

Instruction Set Architecture

Micro Architecture

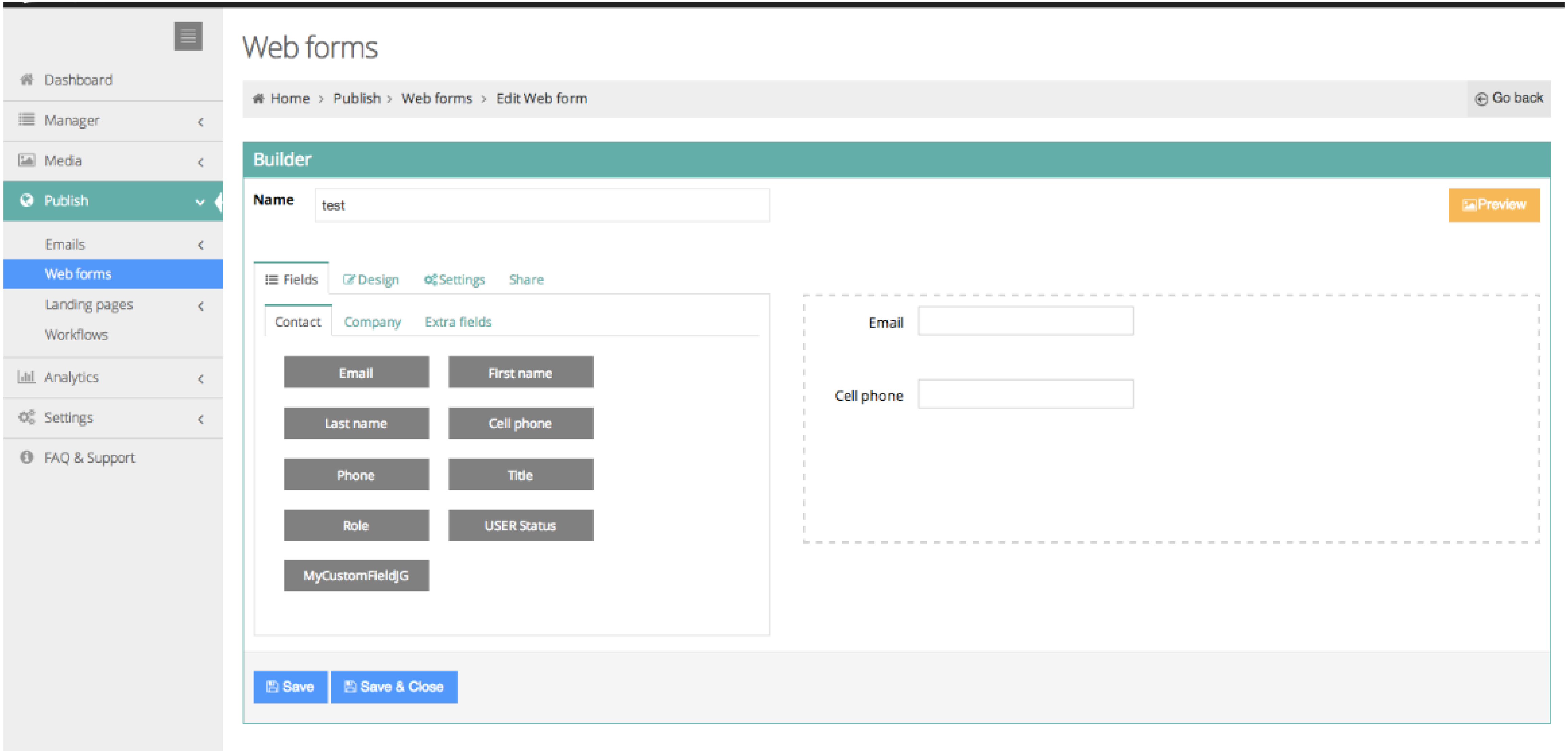
Gates/Registers

Devices (Transistors)

Physics

Increasing order of Complexity
Increasing order of Abstraction
Software
Hardware

Layers of Abstraction



Name: test

Email

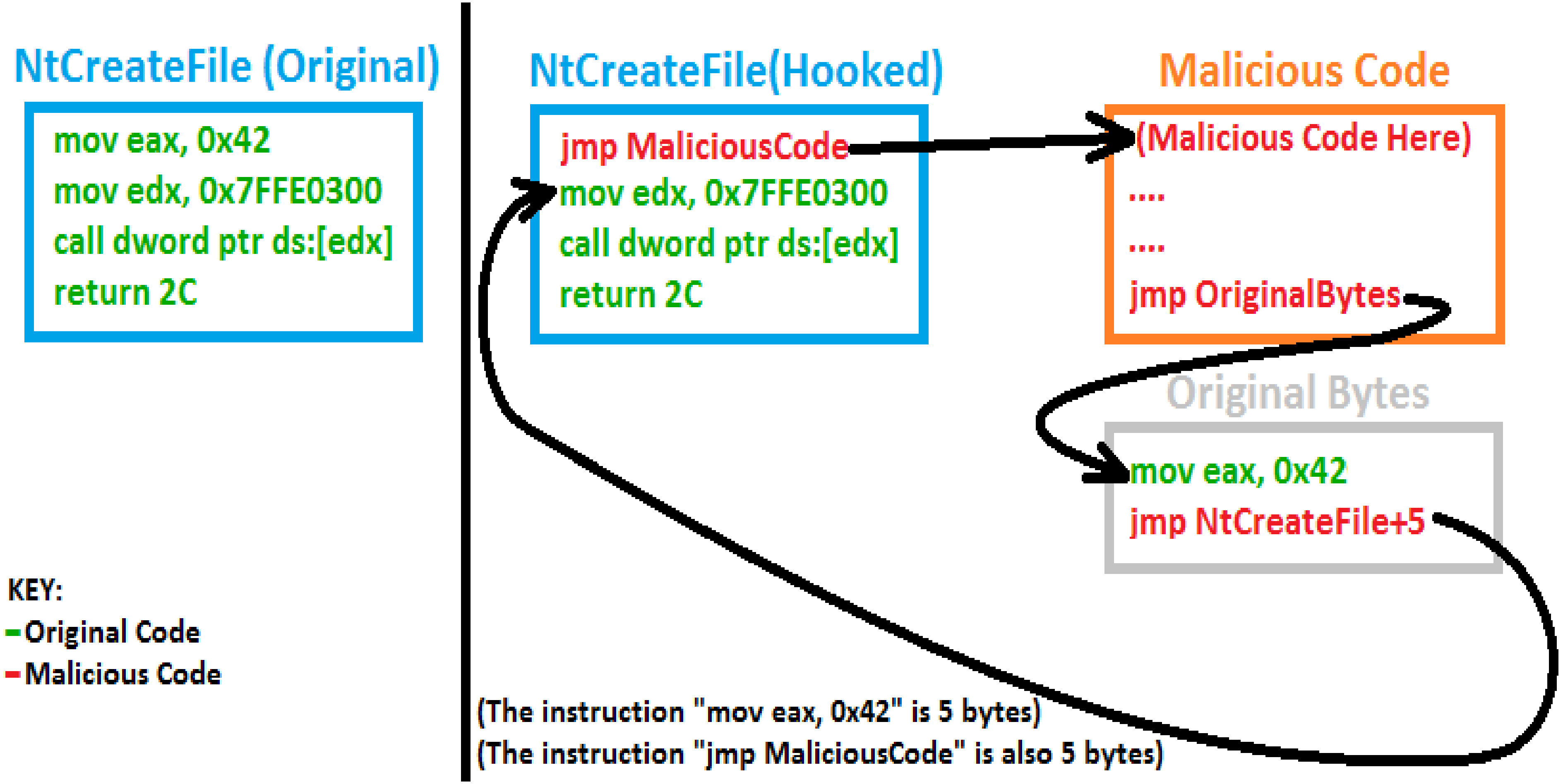
Cell phone

Developers

Hackers

RACE TO THE BOTTOM

Hooking Lowest Wins



NtCreateFile (Original)

```
mov eax, 0x42
mov edx, 0x7FFE0300
call dword ptr ds:[edx]
return 2C
```

NtCreateFile(Hooked)

```
jmp MaliciousCode
mov edx, 0x7FFE0300
call dword ptr ds:[edx]
return 2C
```

Malicious Code

```
(Malicious Code Here)
....
....
jmp OriginalBytes
```

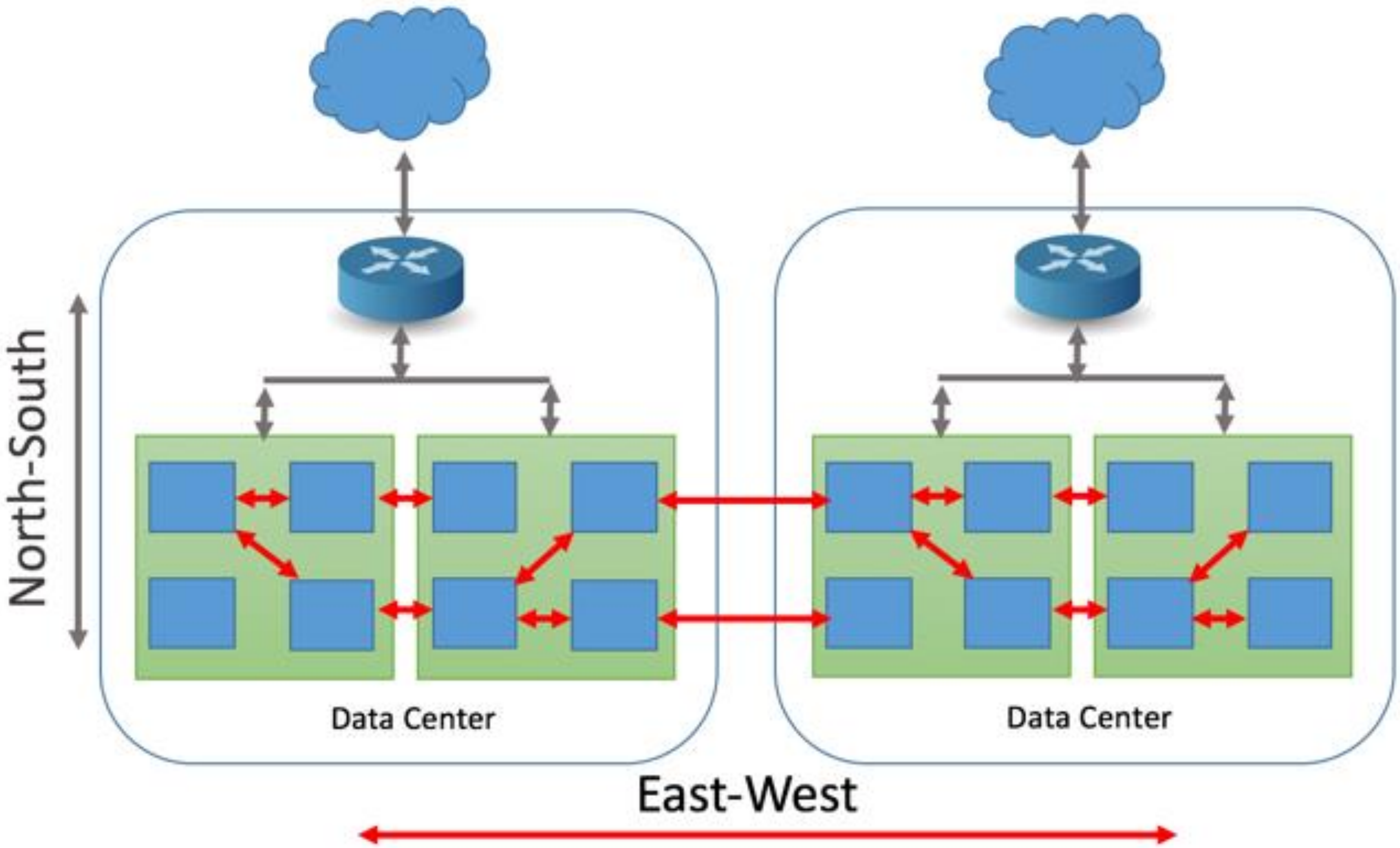
Original Bytes

```
mov eax, 0x42
jmp NtCreateFile+5
```

KEY:
🟩 Original Code
🟥 Malicious Code

(The instruction "mov eax, 0x42" is 5 bytes)
(The instruction "jmp MaliciousCode" is also 5 bytes)

North-South & East-West Attacks and Pivots



https://neuvector.com/network-security/securing-east-west-traffic-in-container-based-data-center/

Break-In

Entry Point is usually a “Pin Hole” issue

For example a known application issue

VAULT DOOR
WEIGHT: 22 1/2 Tons
THICKNESS: 22 Inches
STEEL: 11 Layers of Special Cutting and Drill Resistant
LOCKS: 4 Hamilton Watch Movements for Time Locks

Containers – The “Contained” Challenge

IF you can Break-In

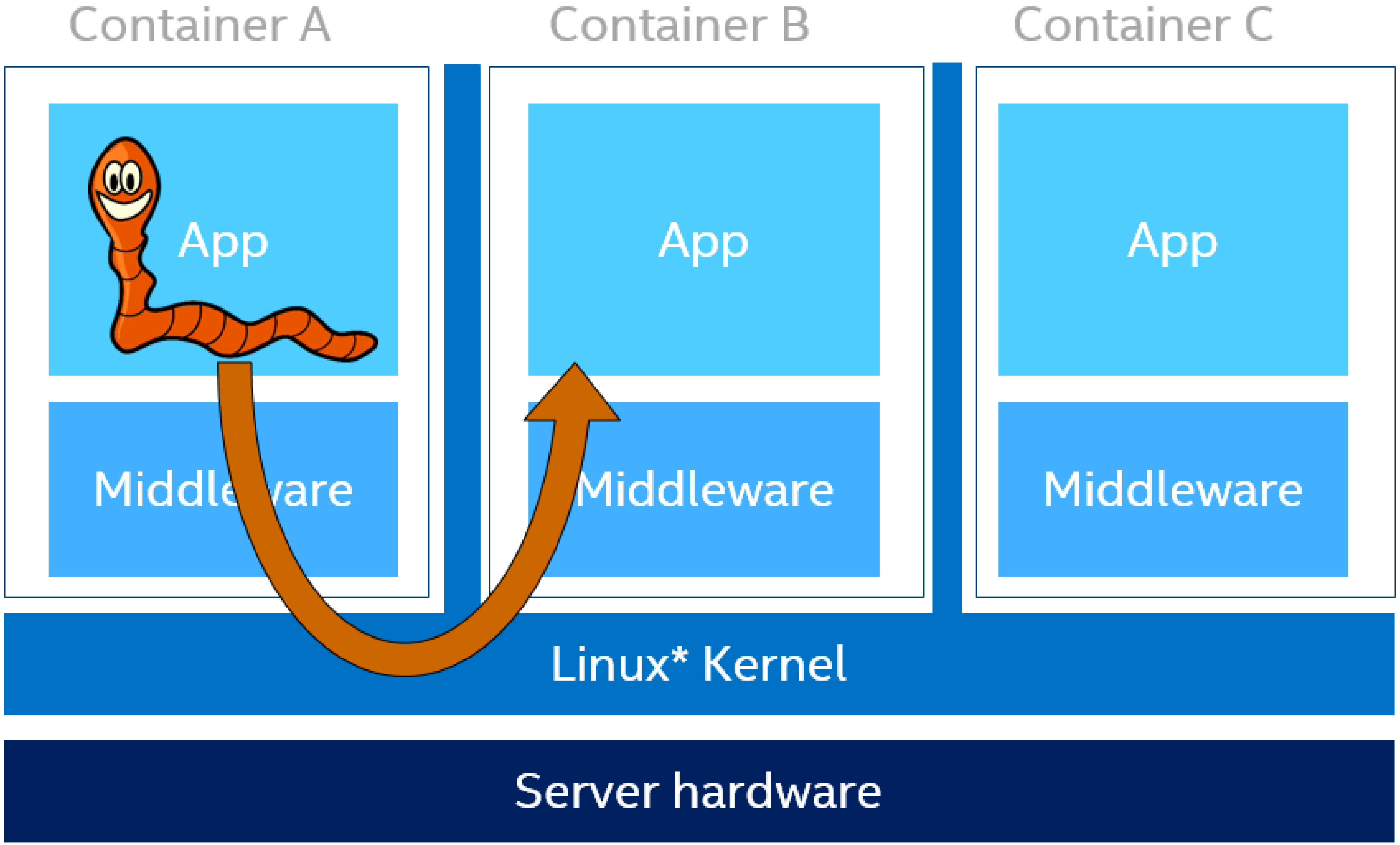
You then Need to Break-Out

http://www.marvinfrancismaninacage.com/

<goWest

goEast>

Either Find a Container Vuln & Exploit



Or - Living off the Land

Relying on misconfiguration, ability to use native tools, or download new and execute

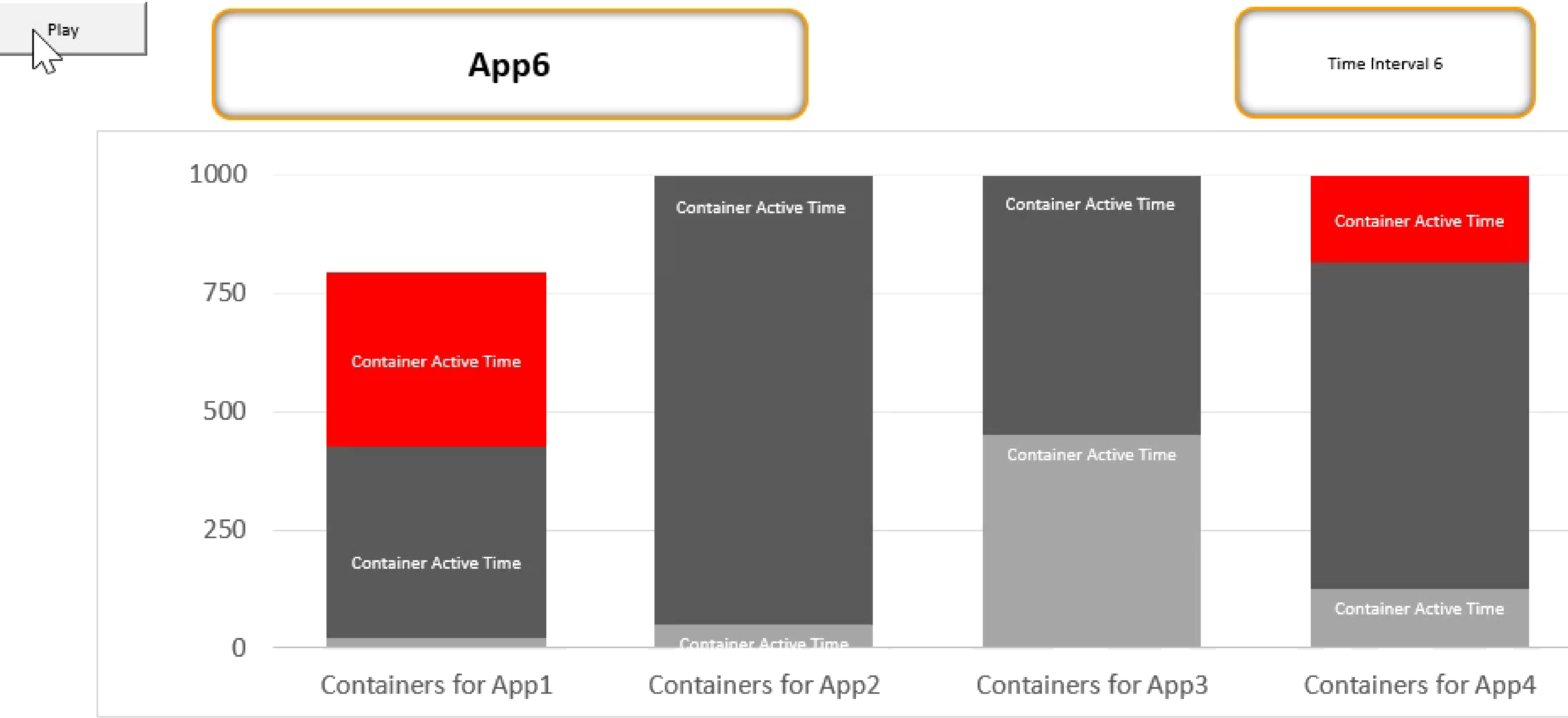
Low TTL Bi-Product
Hacked container may very soon be pulled down.
Much harder for hacker persistence.
Ability to refresh environment quickly – Vuln Mgt improvements e.g. Secure @ Source

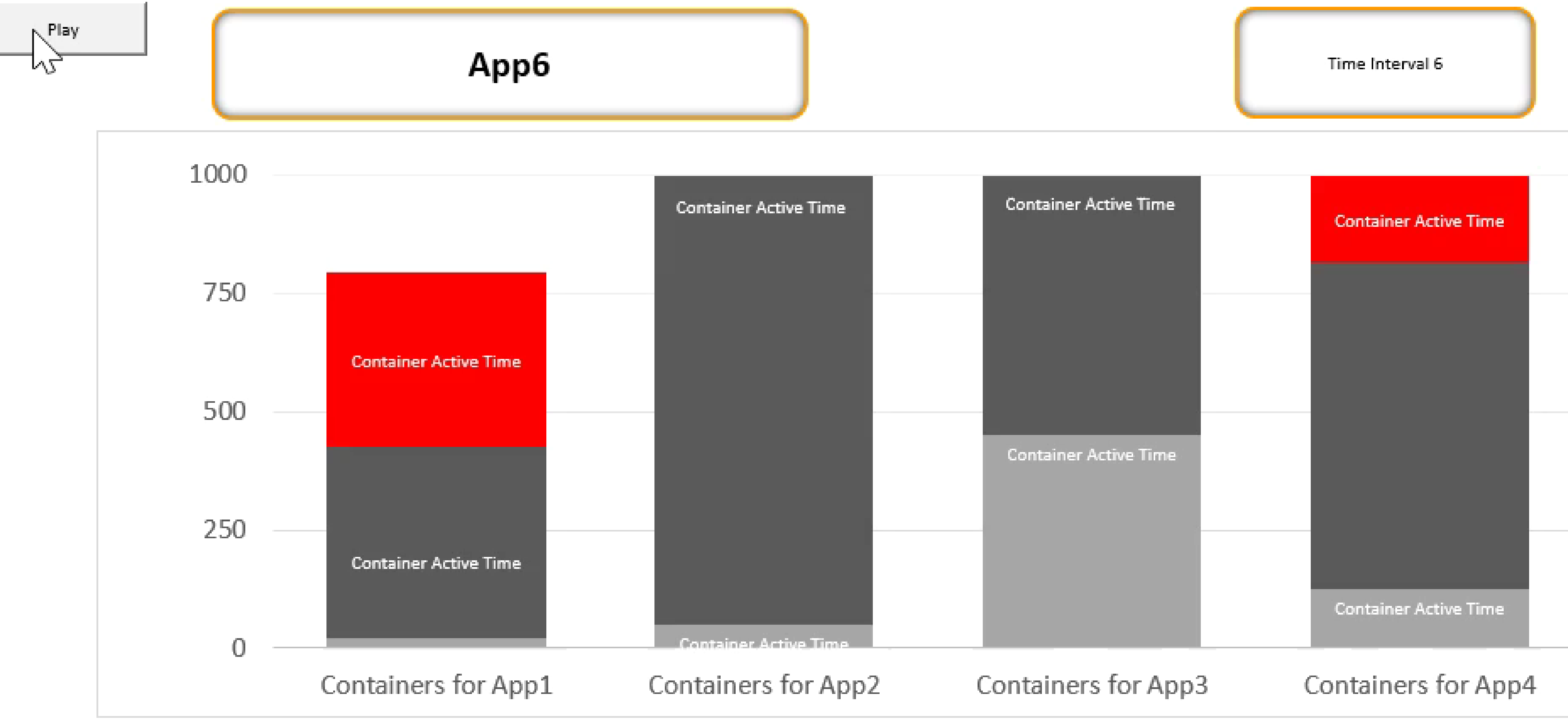
Low TTL Challenge
Hard for Forensics and Monitoring
Vuln Mgt – environment constantly changing
Config Mgt – environment constantly changing

Container TTL

Low TTL Bi-Product	Low TTL Challenge
Hacked container may very soon be pulled down.	Hard for Forensics and Monitoring
Much harder for hacker persistence.	Vuln Mgt – environment constantly changing
Ability to refresh environment quickly – Vuln Mgt improvements e.g. Secure @ Source	Config Mgt – environment constantly changing

Content Slide Layout


Play
App6
Time Interval 6
1000
750
500
250
0
Container Active Time
Container Active Time
Container Active Time
Container Active Time
Container Active Time
Container Active Time
Container Active Time
Container Active Time
Containers for App1
Containers for App2
Containers for App3
Containers for App4



App6

Time Interval 6

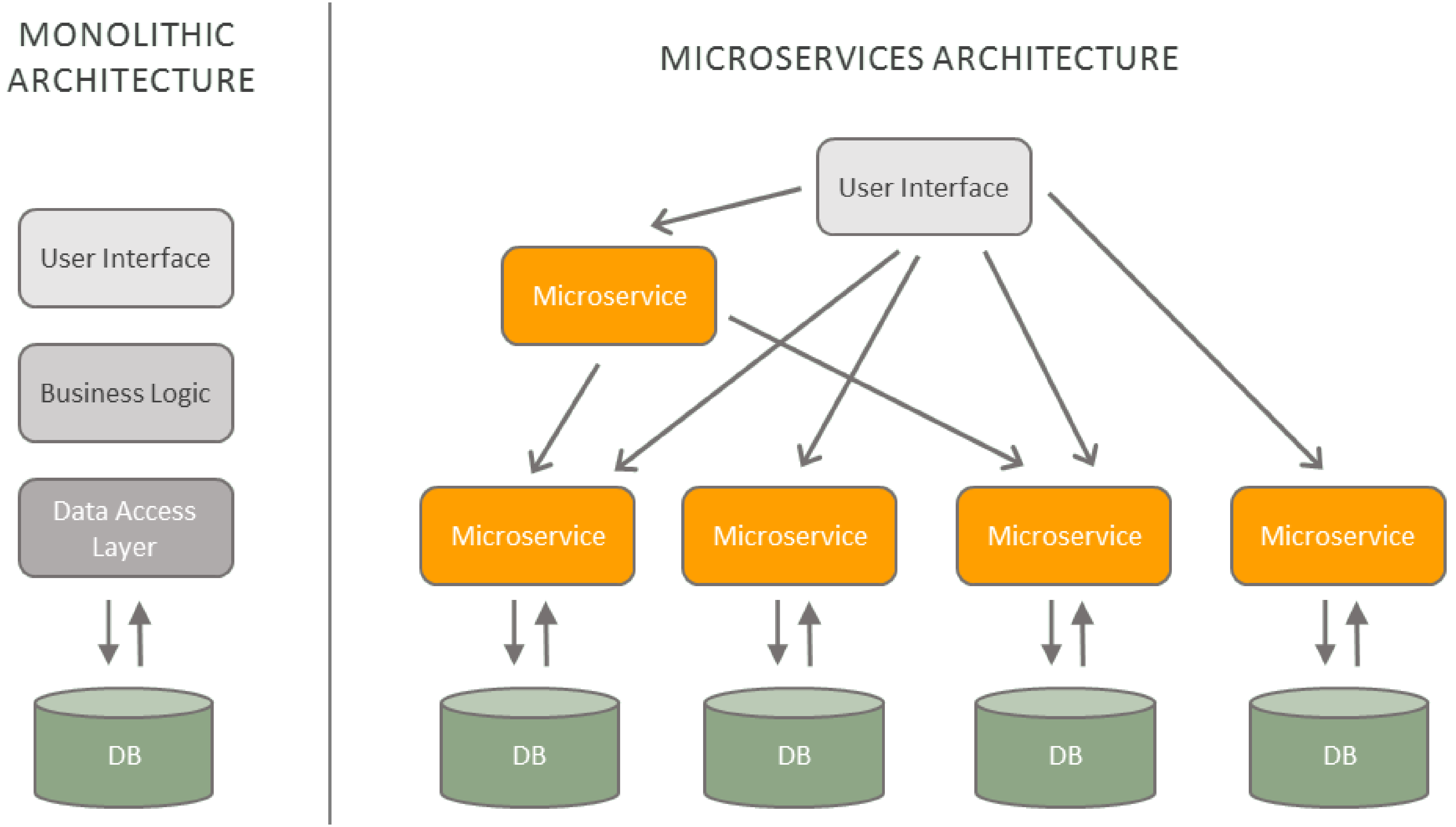
How to Upgrade your Vuln Mgt Program

What to expect from a Pen Test	Implications for CaaS
Supply Chain Risk	DevSecOps

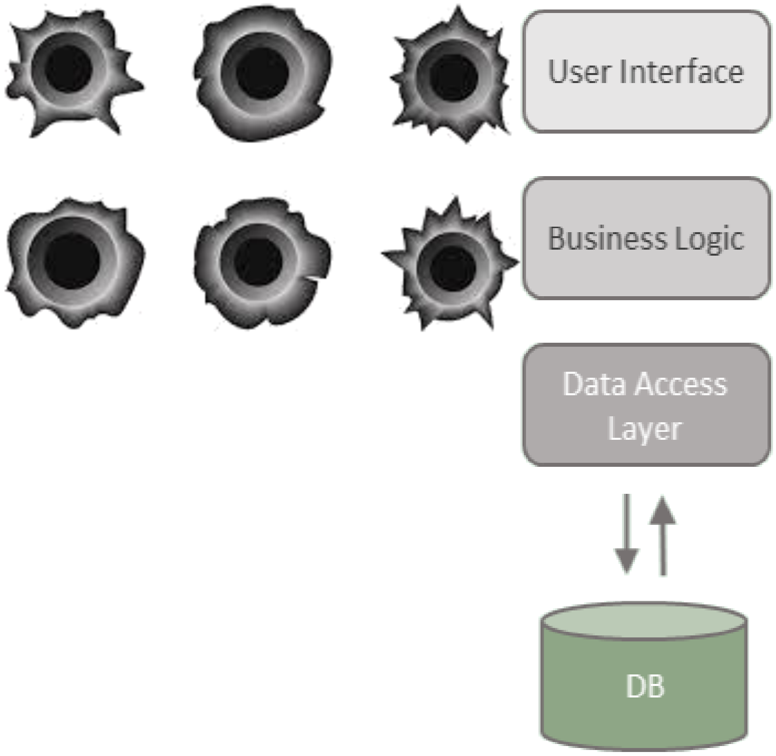
Pen Test – Mechanical Attack

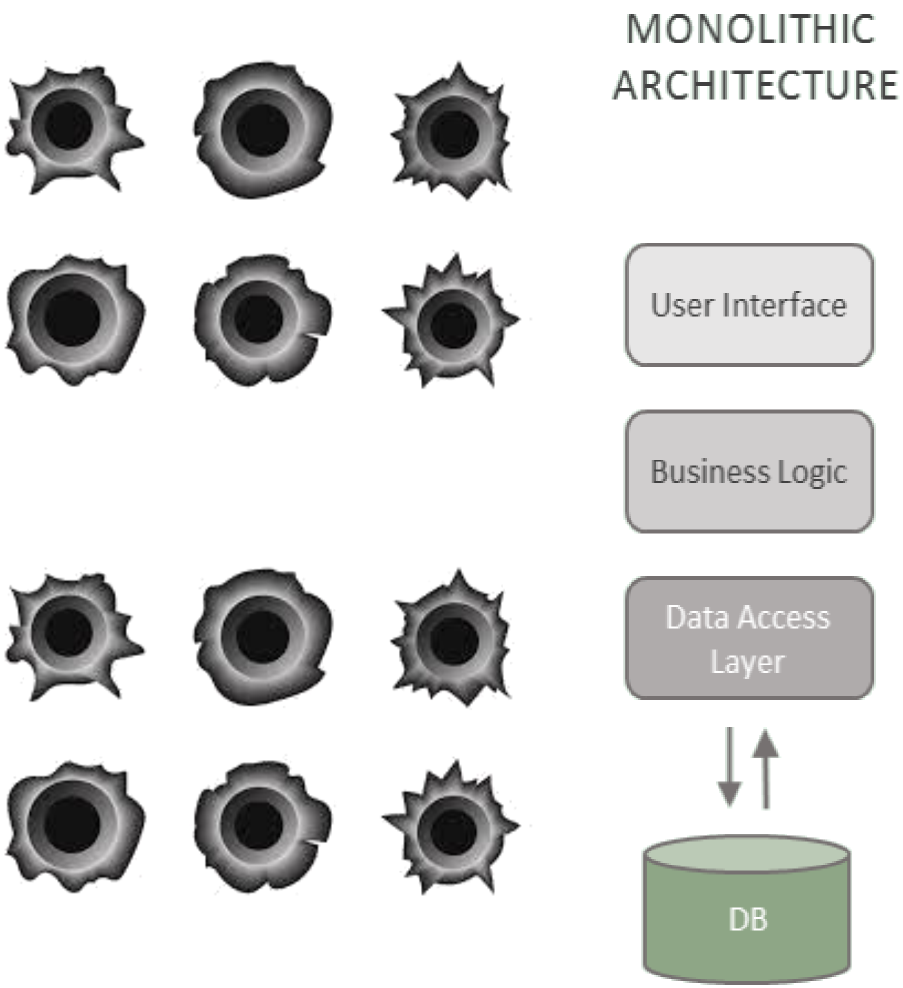
vs Knowledge & Finesse

Monolithic vs Microservices Architecture

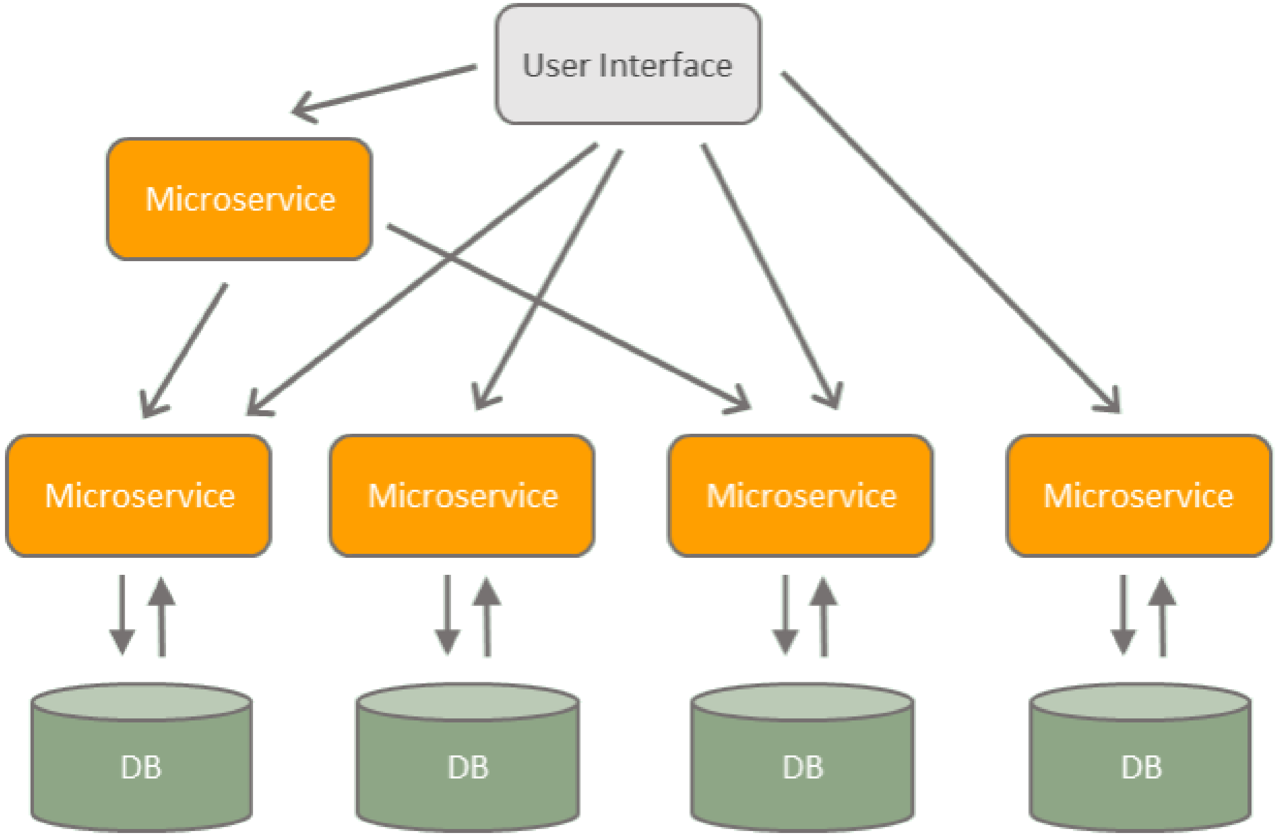


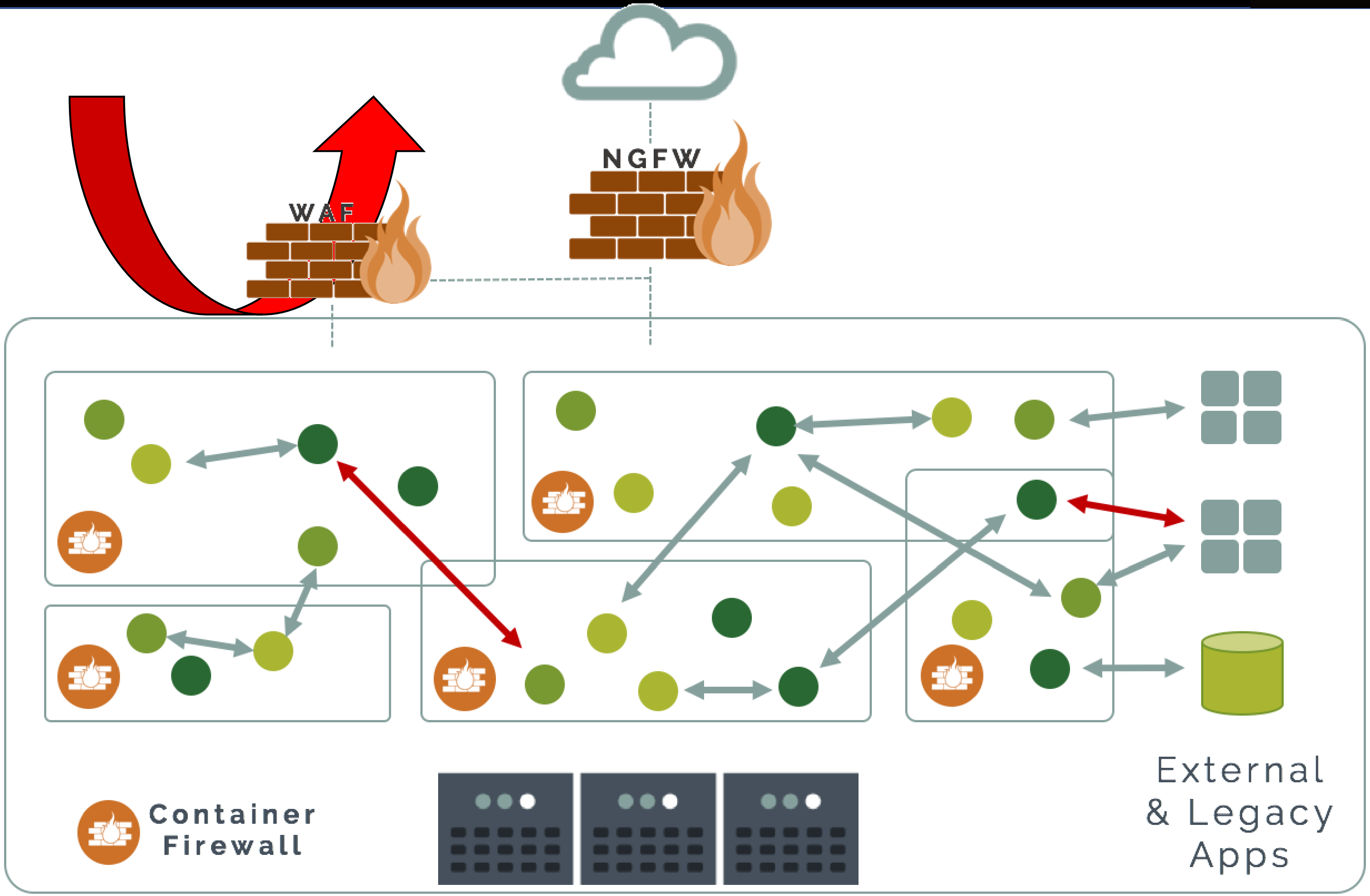
MONOLITHIC ARCHITECTURE



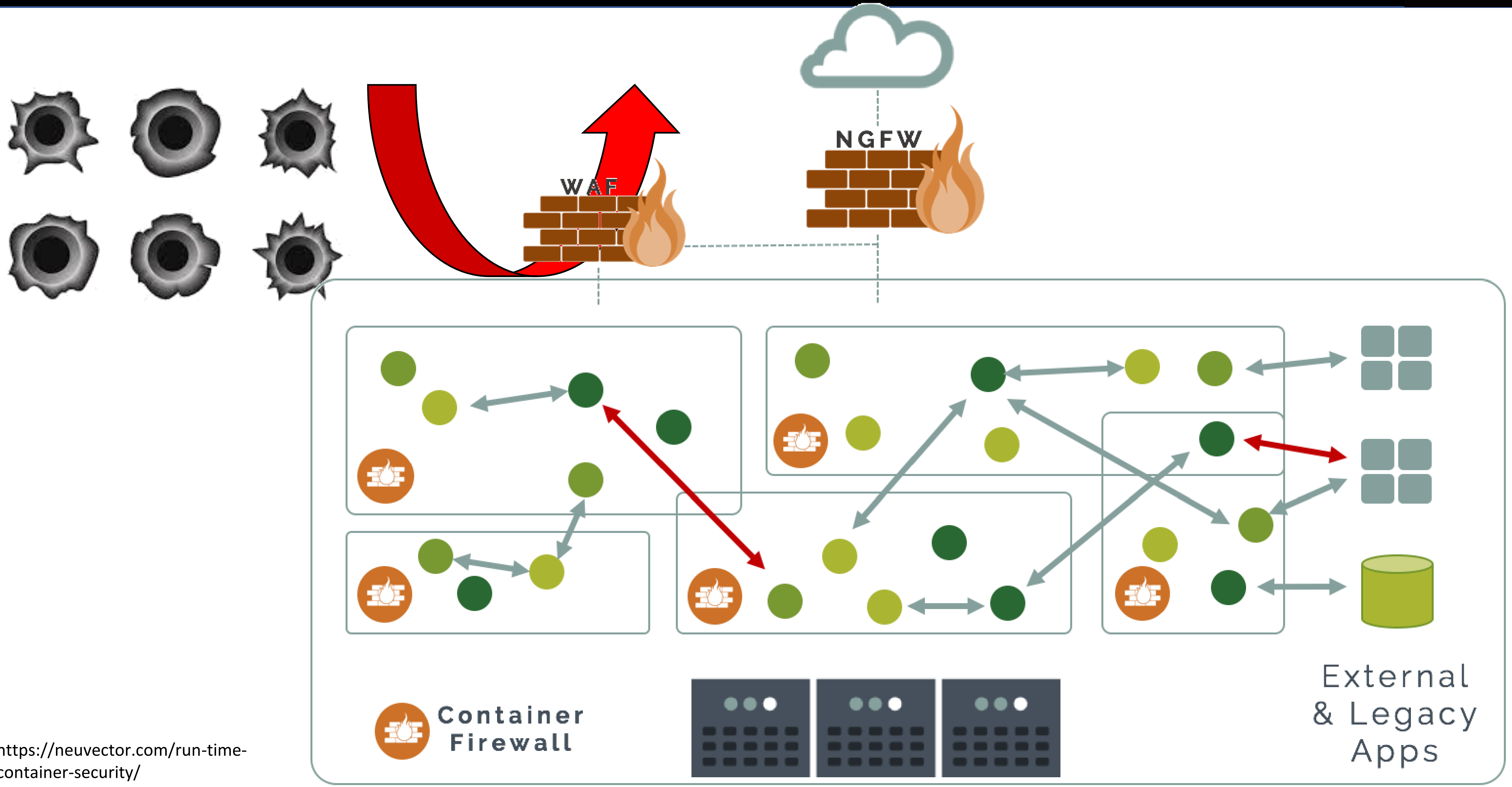

MONOLITHIC
ARCHITECTURE
User Interface
Business Logic
Data Access Layer
DB

MICROSERVICES ARCHITECTURE

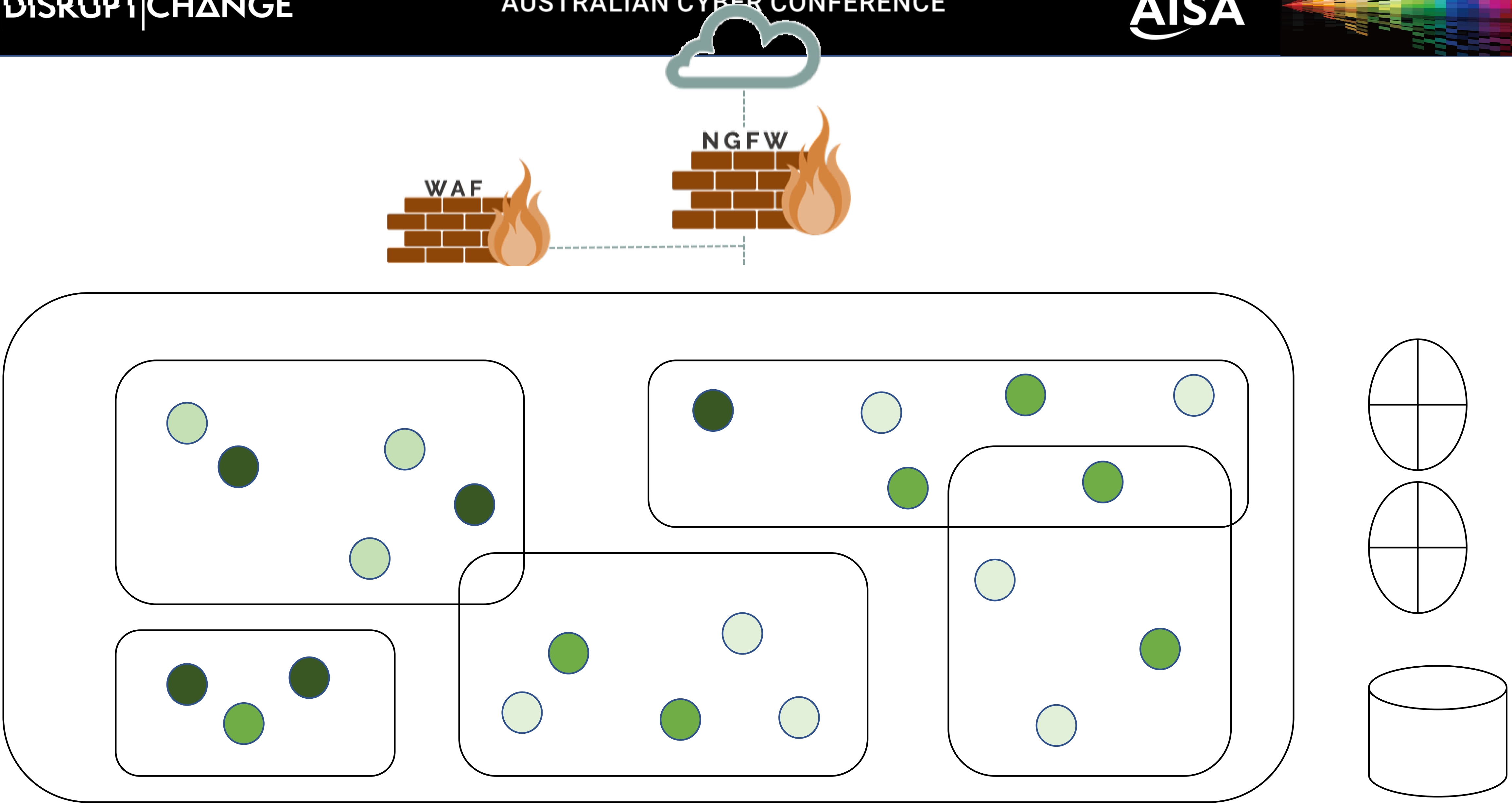


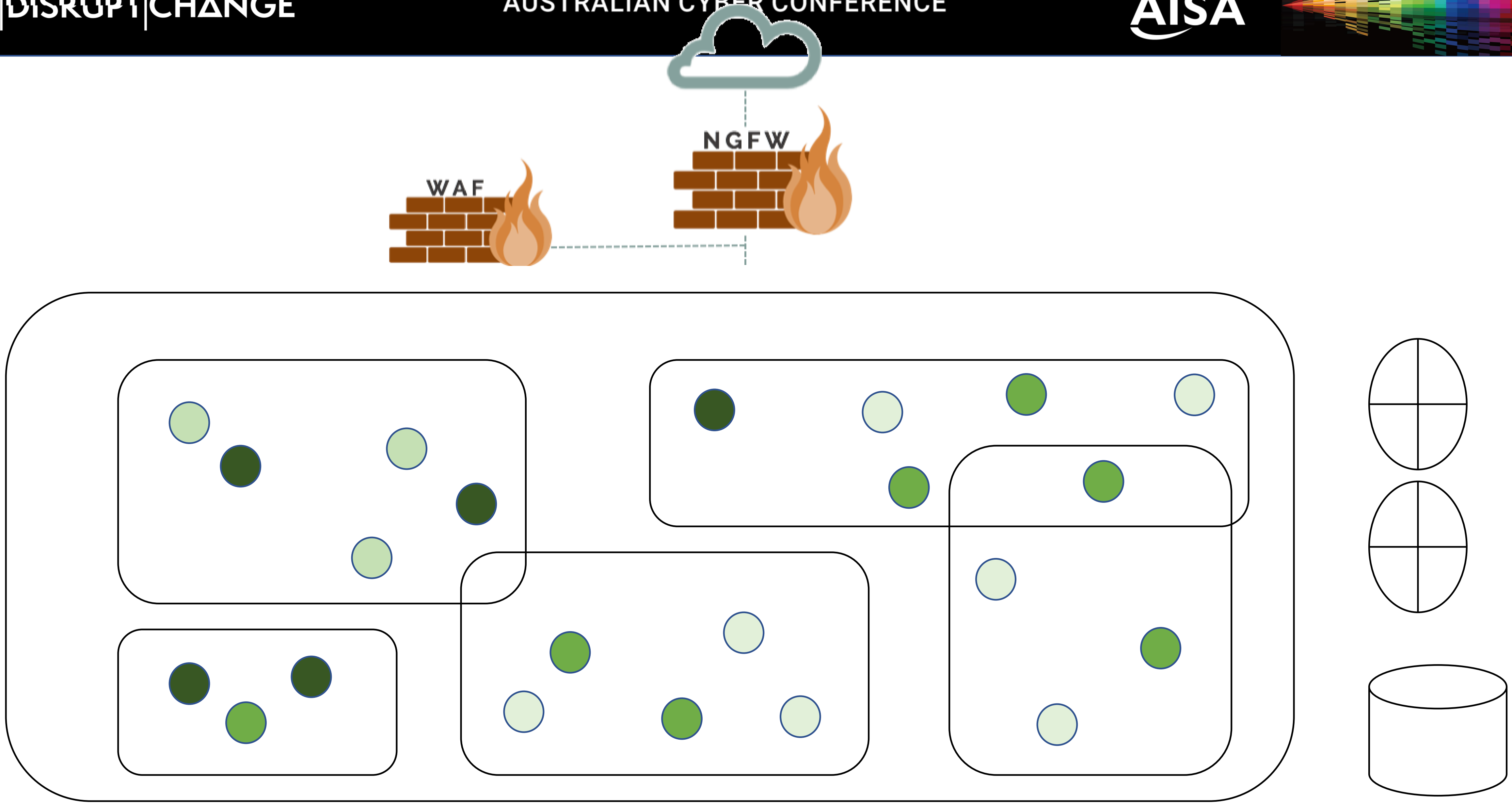


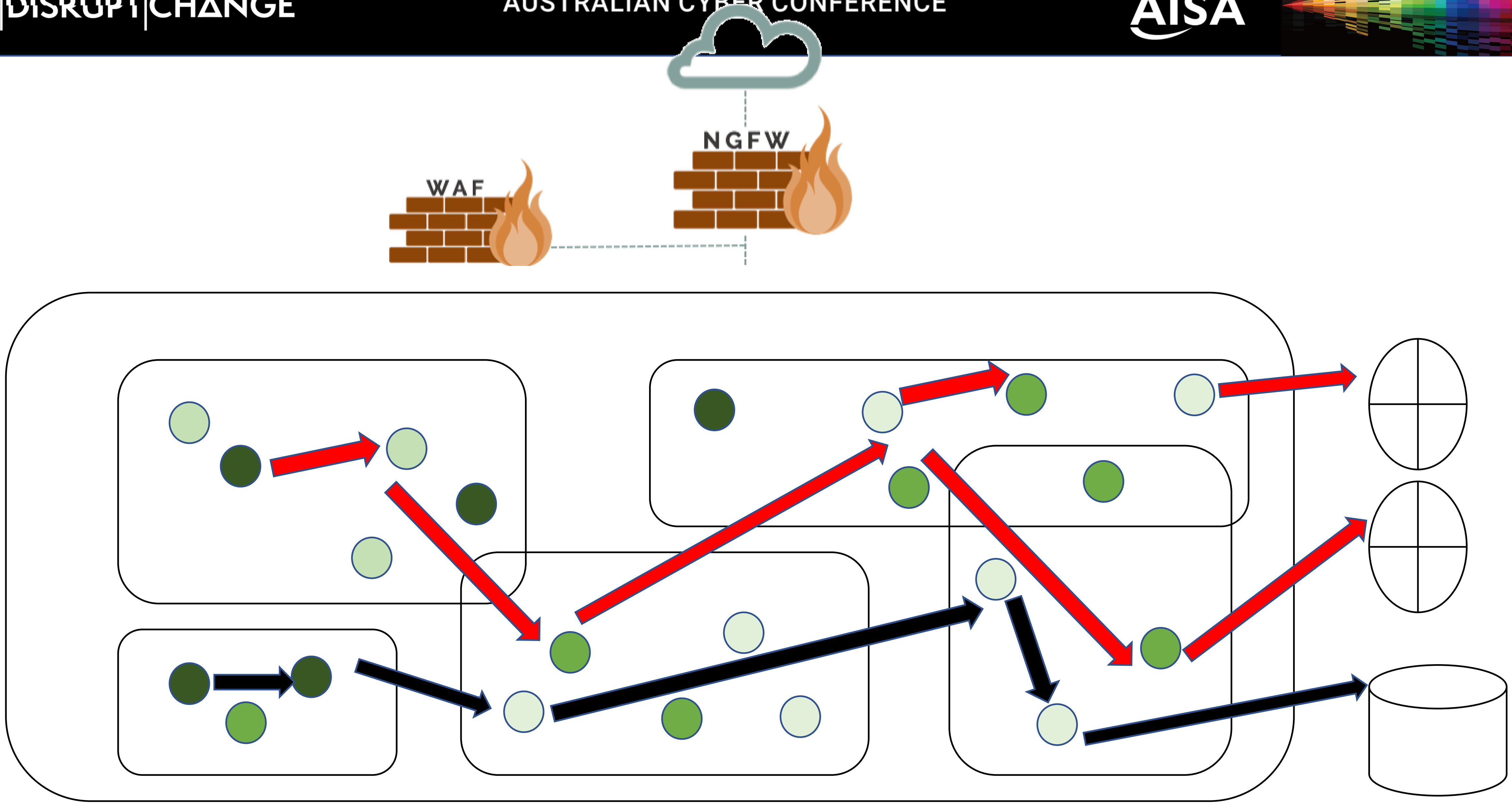
https://neuvector.com/run-time-container-security/

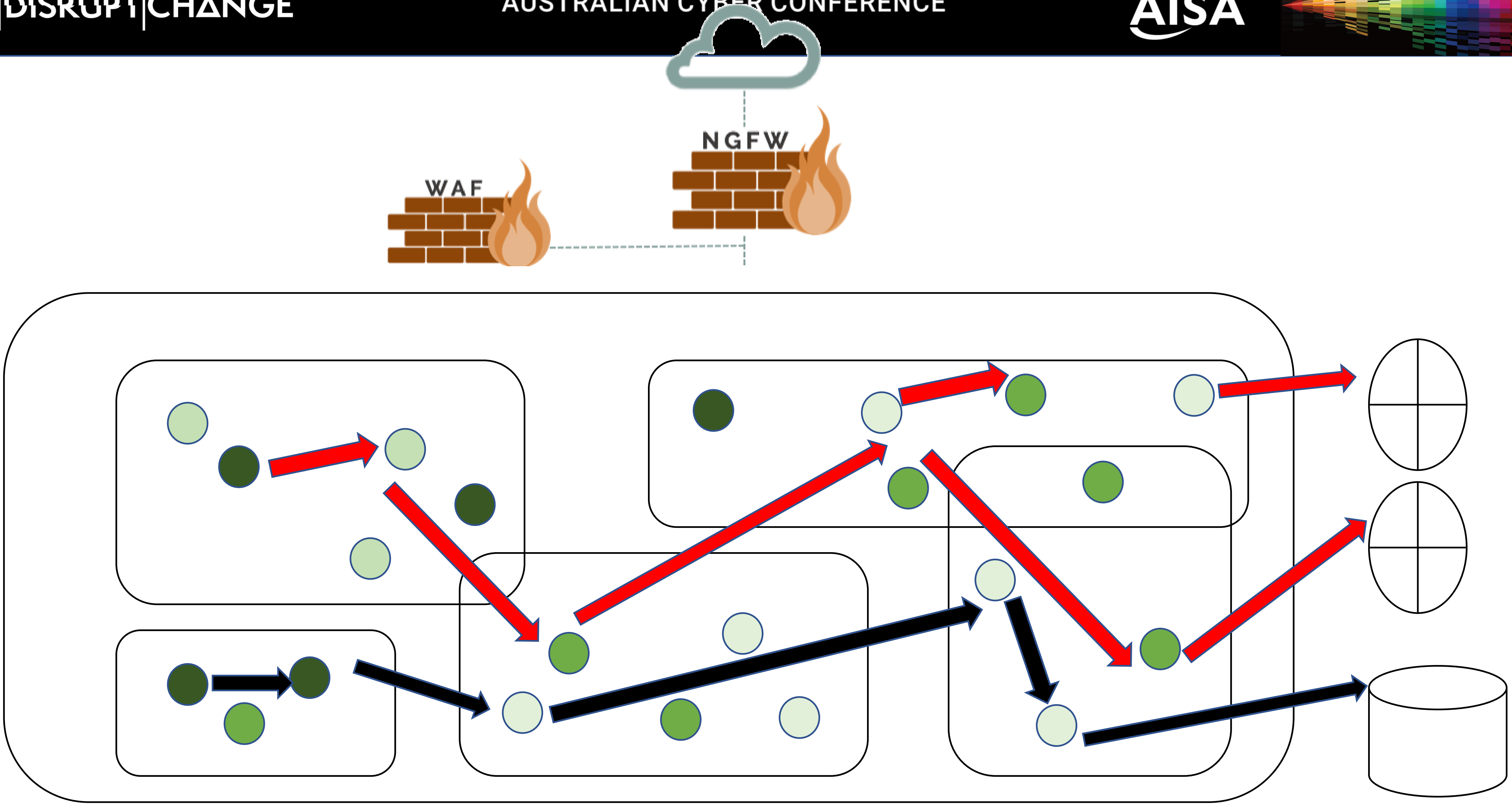


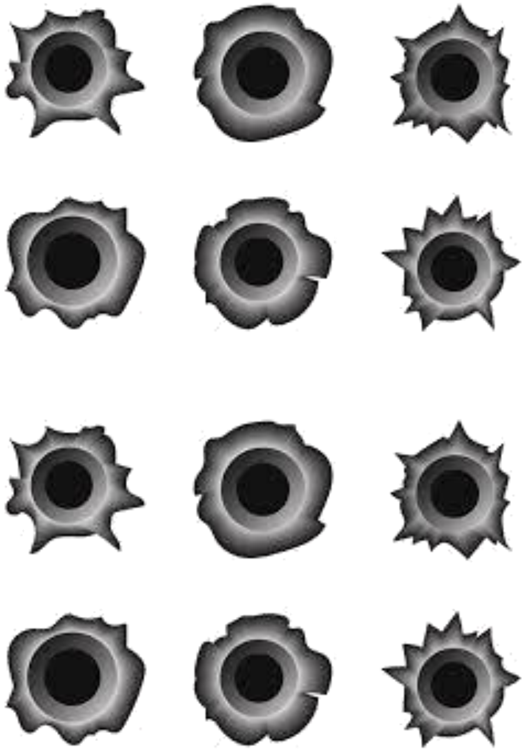
https://neuvector.com/run-time-container-security/

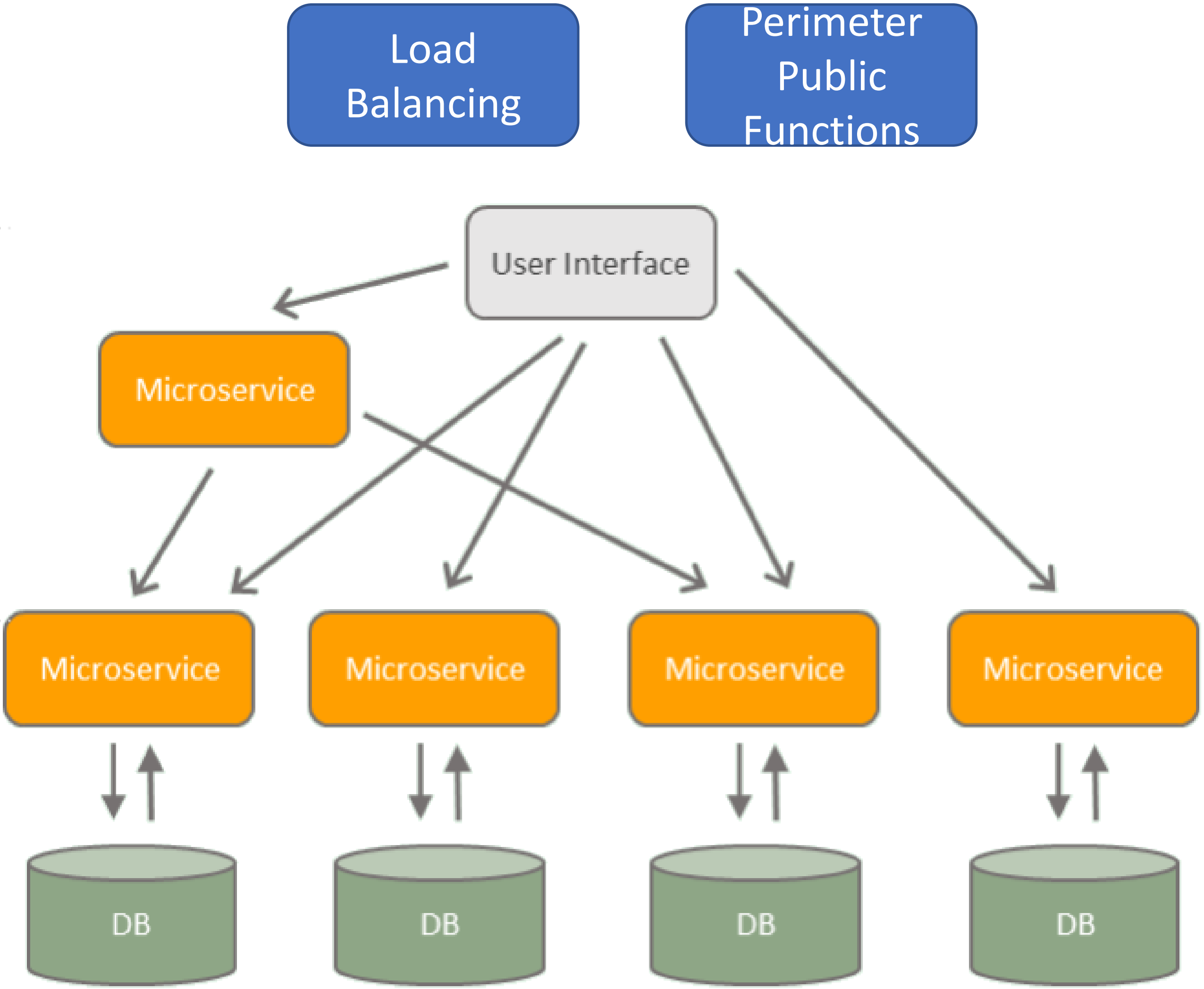

NGFW
WAF

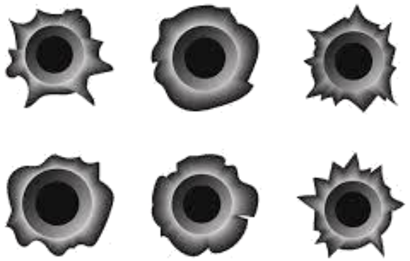

NGFW
WAF

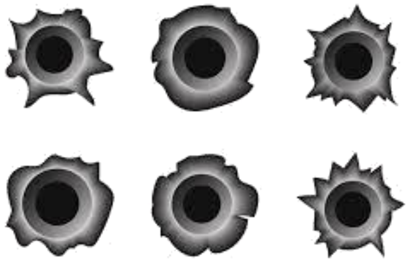

NGFW
WAF

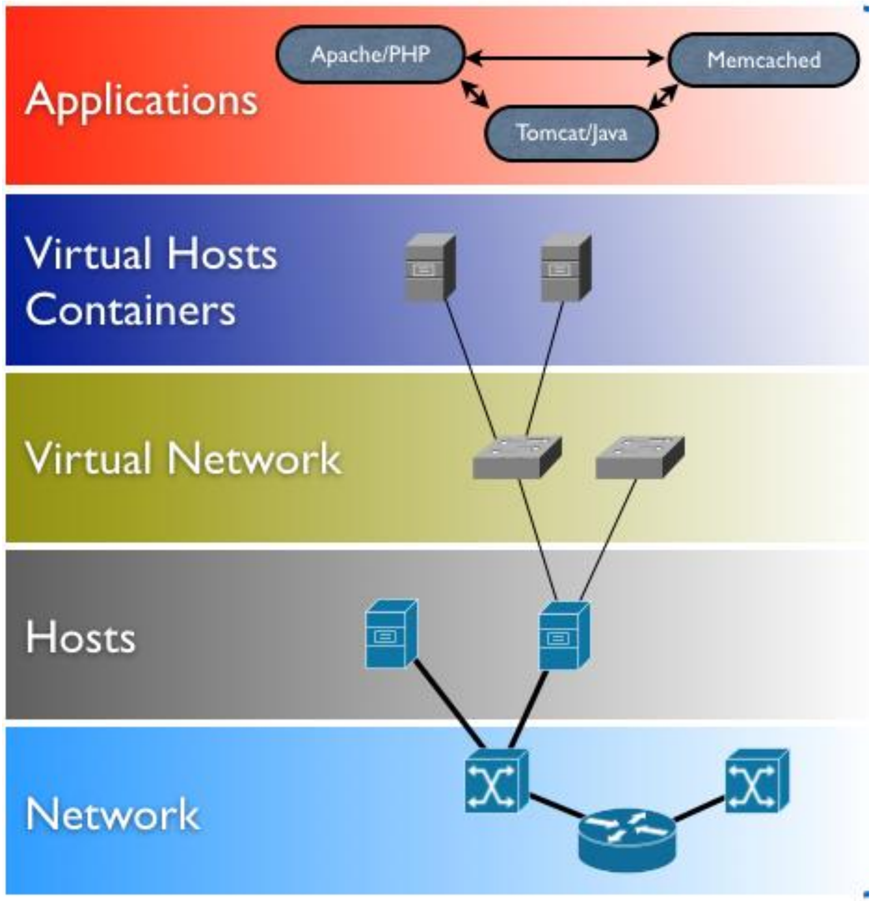

NGFW
WAF

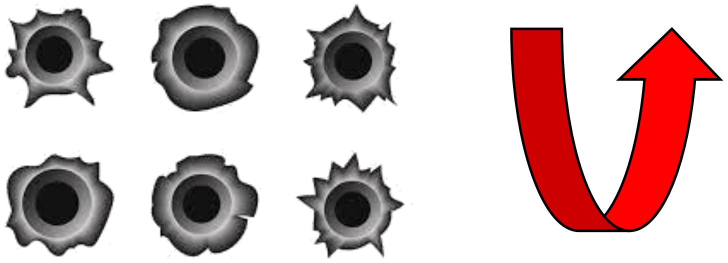


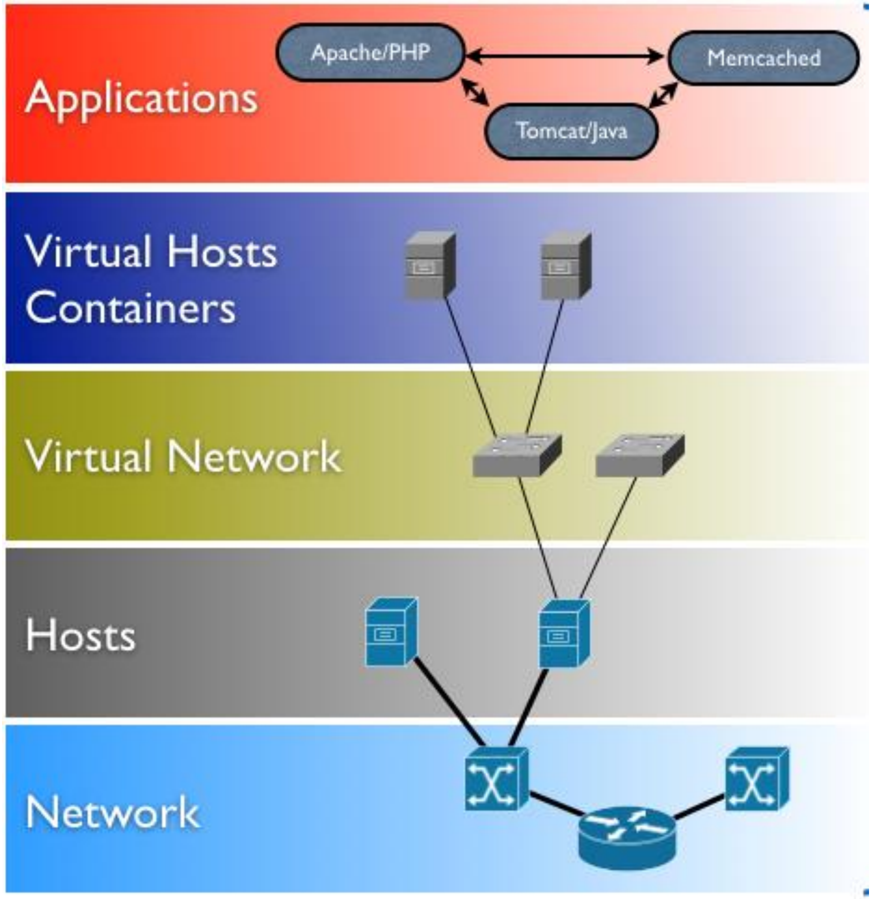

Load Balancing
Perimeter Public Functions
User Interface
Microservice
Microservice
Microservice
Microservice
Microservice
DB
DB
DB
DB

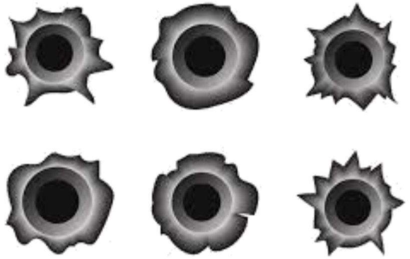


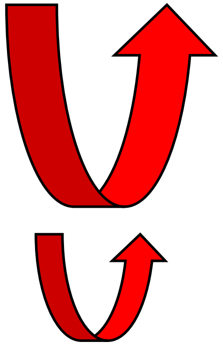


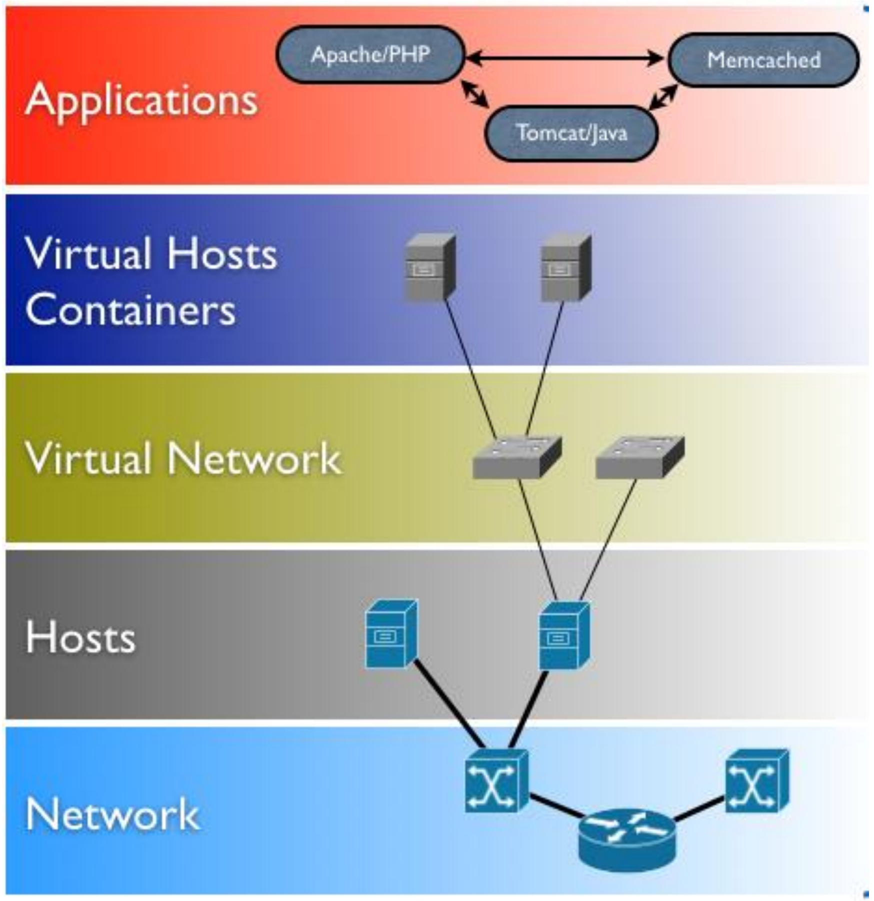

Applications
Apache/PHP
Memcached
Tomcat/Java
Virtual Hosts
Containers
Virtual Network
Hosts
Network




Applications
Apache/PHP
Memcached
Tomcat/Java
Virtual Hosts
Containers
Virtual Network
Hosts
Network





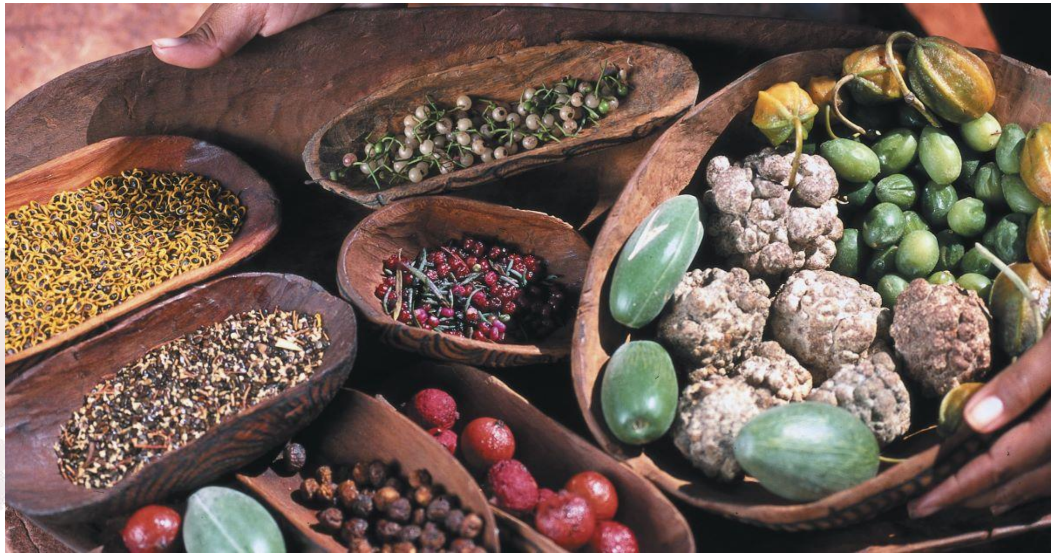

Applications
Apache/PHP
Memcached
Tomcat/Java
Virtual Hosts
Containers
Virtual Network
Hosts
Network

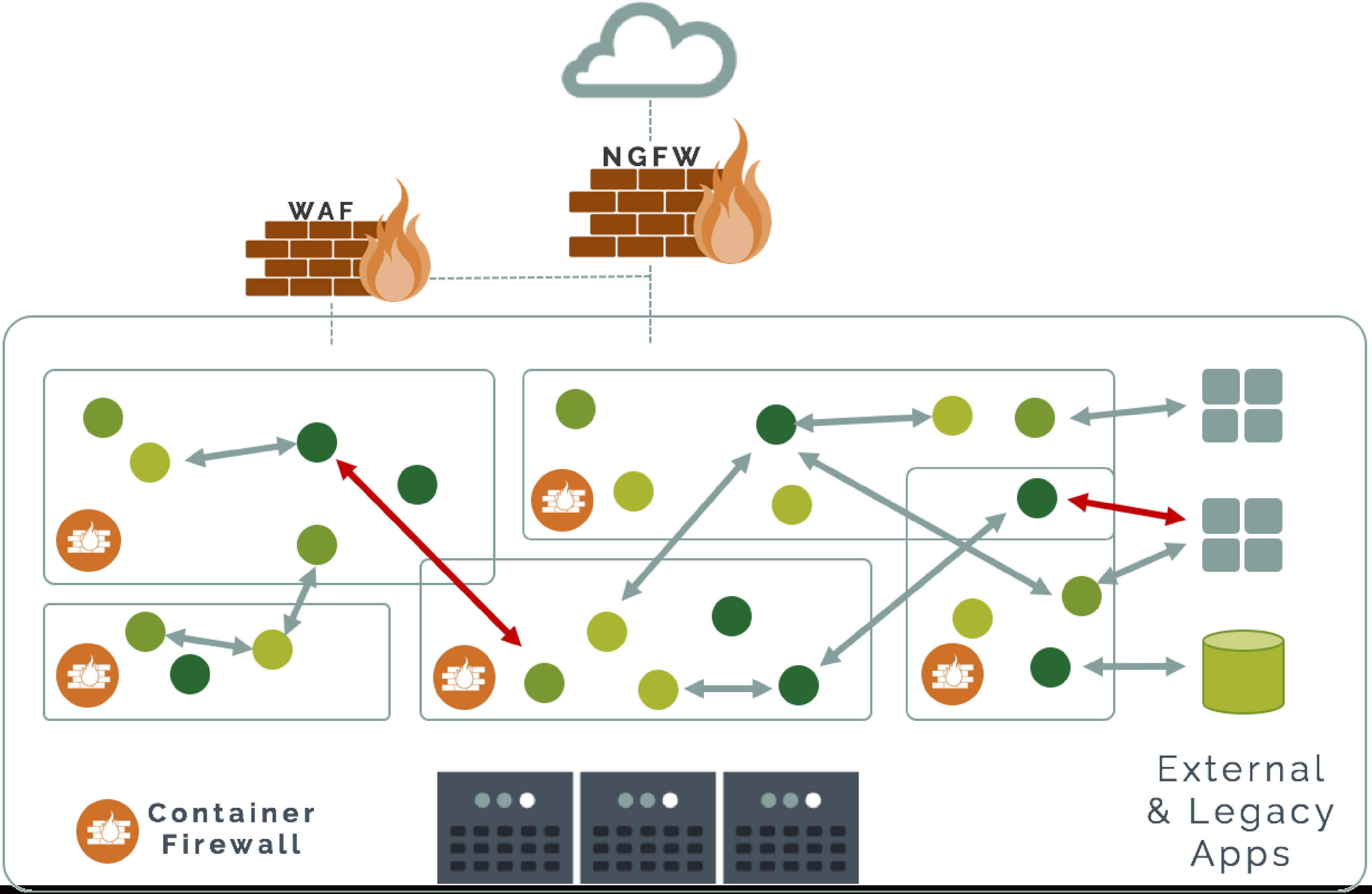
Hack Transformation

Hack Transformation

Hack Transformation

Hack Transformation





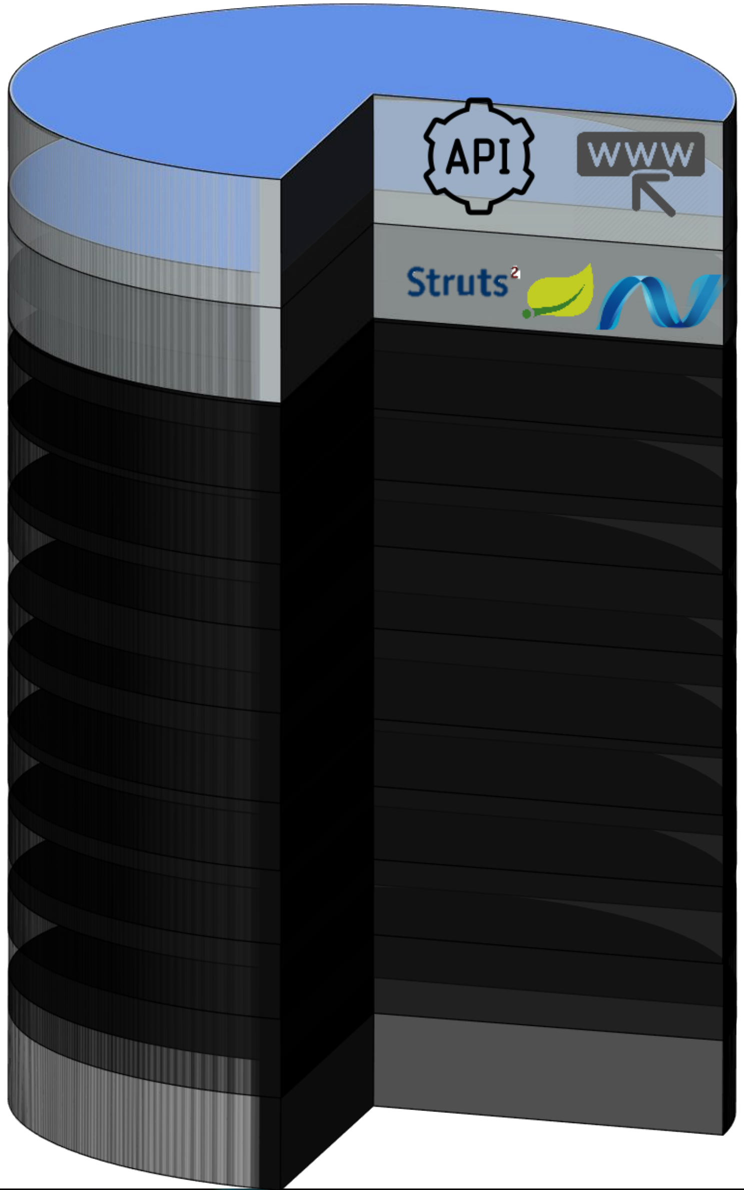
https://neuvector.com/network-security/next-generation-firewall-vs-container-firewall/

Security Testing Needs to Go Down The Stack

Security Testing Needs to Go Down The Stack

User Interface (WebApps, forms, logons, API's)

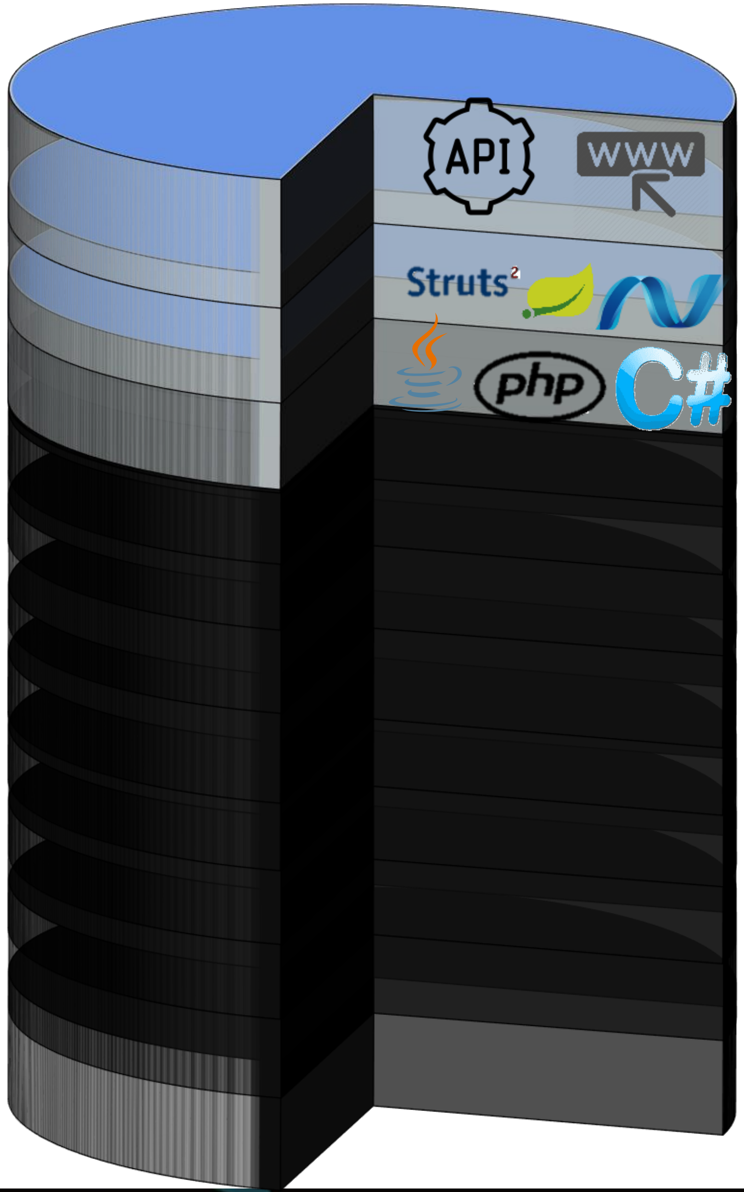
Security Testing Needs to Go Down The Stack



User Interface (WebApps, forms, logons, API’s)

Framework (Struts, Spring, .NET)

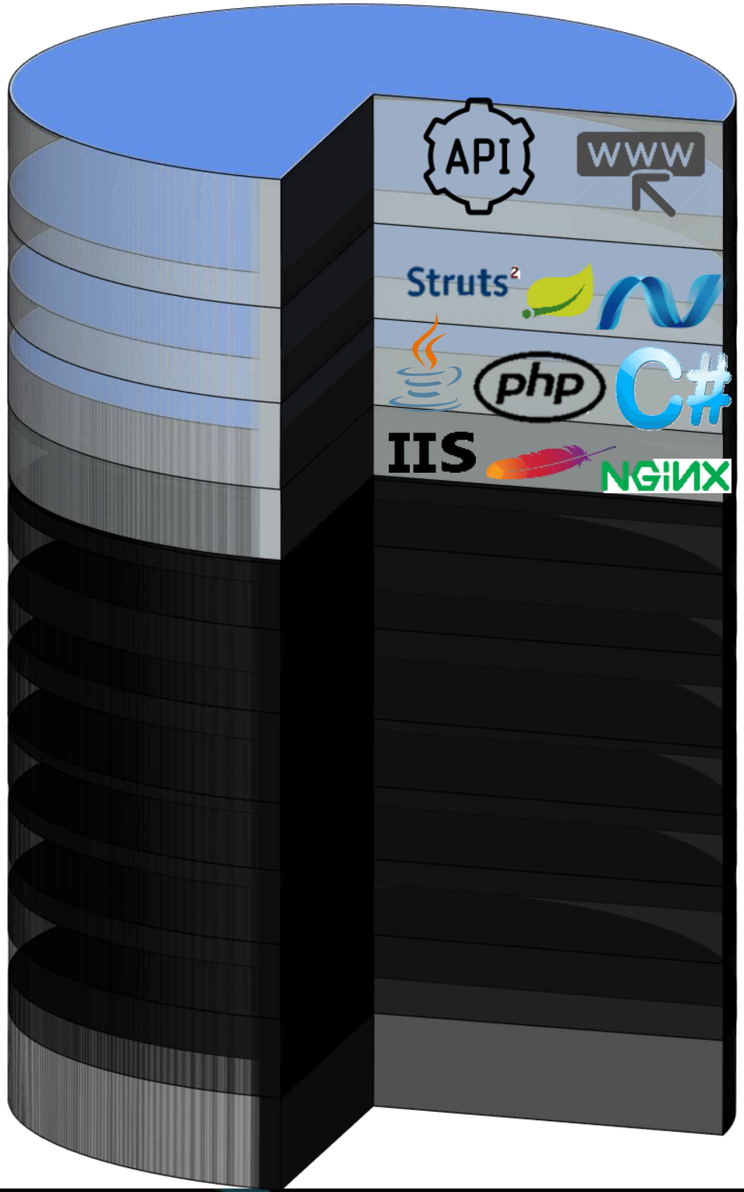
Security Testing Needs to Go Down The Stack



User Interface (WebApps, forms, logons, API’s)

Framework (Struts, Spring, .NET)

Language (Java, PHP, .NET)



Security Testing Needs to Go Down The Stack

User Interface (WebApps, forms, logons, API’s)

Framework (Struts, Spring, .NET)

Language (Java, PHP, .NET)

AppServer (IIS, Apache, Nginx)

Security Testing Needs to Go Down The Stack

User Interface (WebApps, forms, logons, API’s)

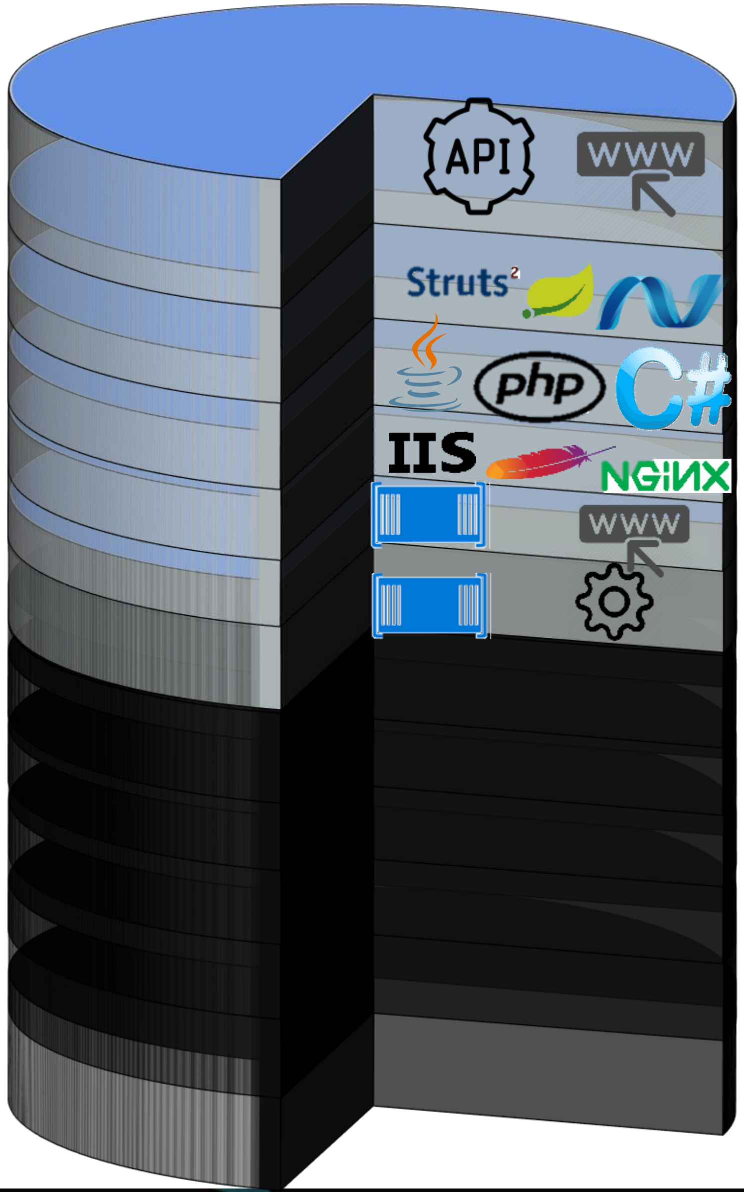
Framework (Struts, Spring, .NET)

Language (Java, PHP, .NET)

AppServer (IIS, Apache, Nginx)

Process UI (Container, presentation layer)

Security Testing Needs to Go Down The Stack



User Interface (WebApps, forms, logons, API’s)

Framework (Struts, Spring, .NET)

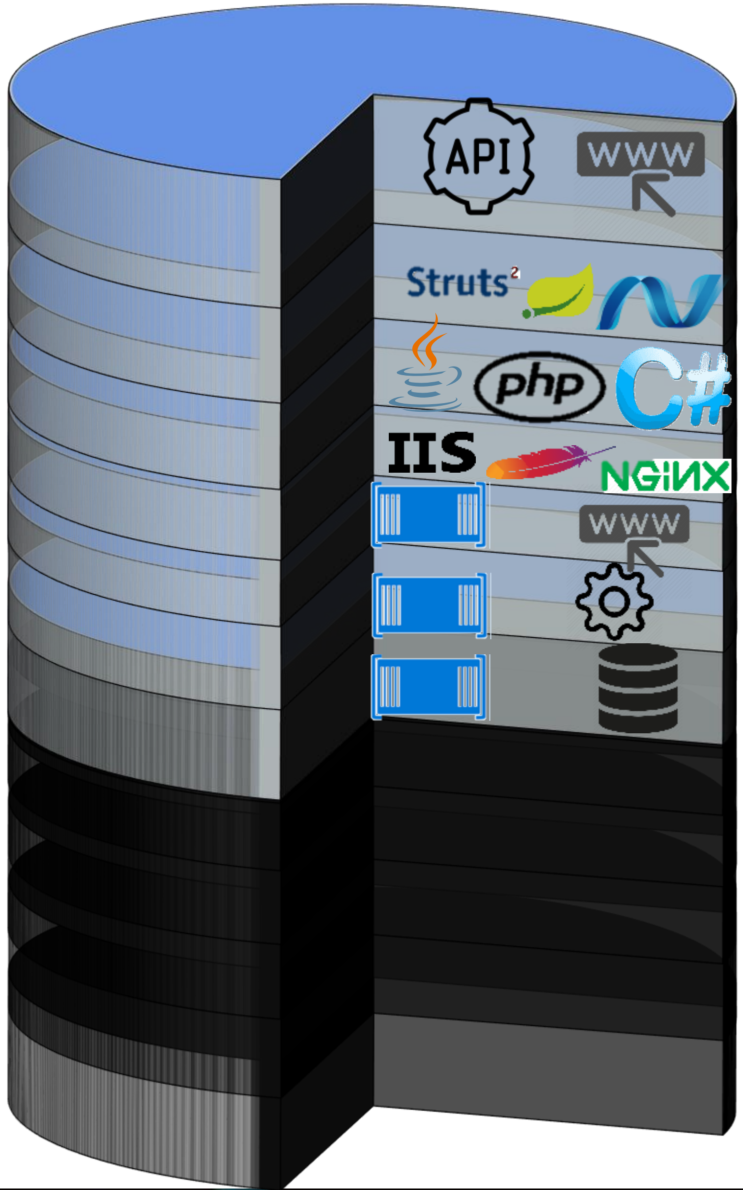
Language (Java, PHP, .NET)

AppServer (IIS, Apache, Nginx)

Process UI (Container, presentation layer)

Process App (Container, application processing)

Security Testing Needs to Go Down The Stack



User Interface (WebApps, forms, logons, API's)

Framework (Struts, Spring, .NET)

Language (Java, PHP, .NET)

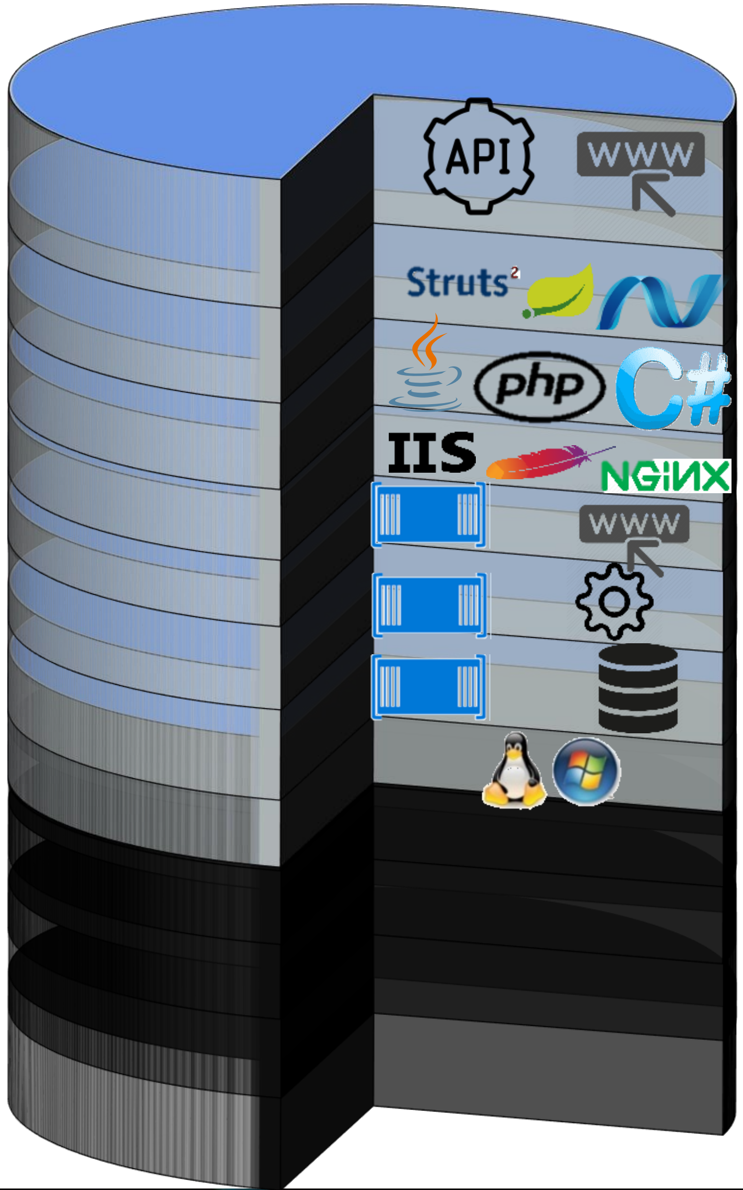
AppServer (IIS, Apache, Nginx)

Process UI (Container, presentation layer)

Process App (Container, application processing)

Process BackEnd (Container, database)

Security Testing Needs to Go Down The Stack



- User Interface (WebApps, forms, logons, API's)
- Framework (Struts, Spring, .NET)
- Language (Java, PHP, .NET)
- AppServer (IIS, Apache, Nginx)
- Process UI (Container, presentation layer)
- Process App (Container, application processing)
- Process BackEnd (Container, database)
- Operating System (Linux, Windows)

Security Testing Needs to Go Down The Stack

API
WWW
Struts2
php
C#
IIS
NGINX
WWW

User Interface (WebApps, forms, logons, API's)

Framework (Struts, Spring, .NET)

Language (Java, PHP, .NET)

AppServer (IIS, Apache, Nginx)

Process UI (Container, presentation layer)

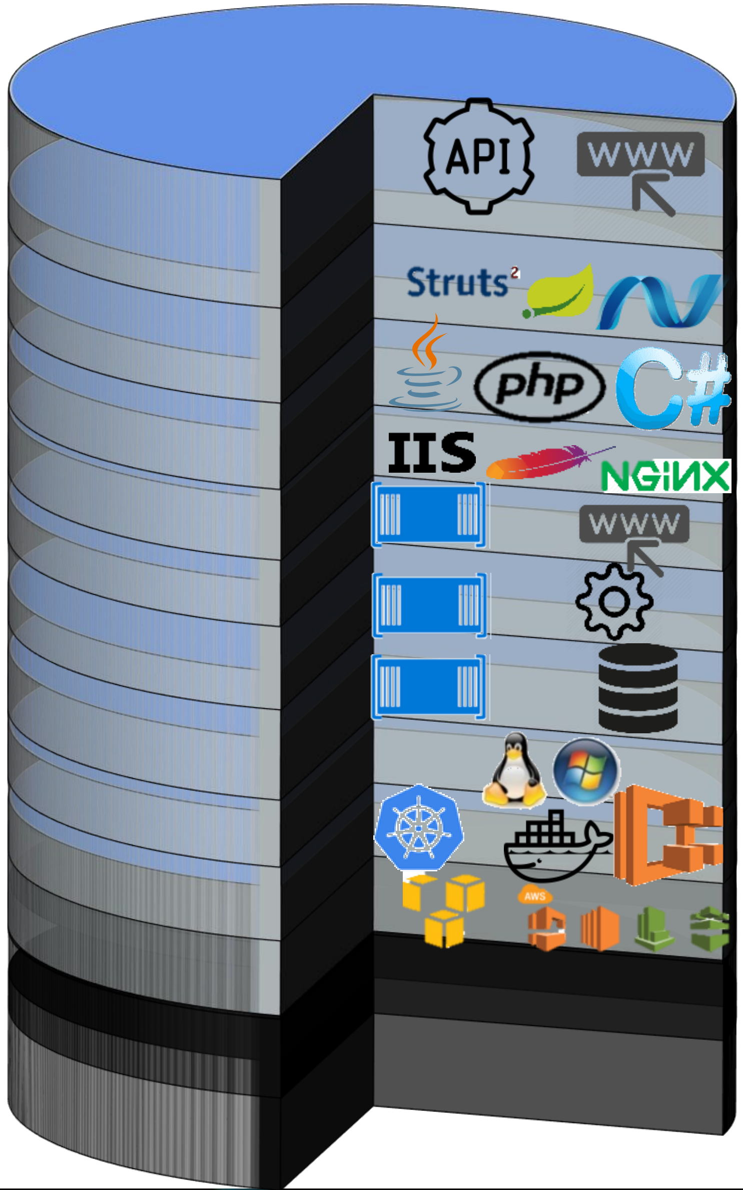
Process App (Container, application processing)

Process BackEnd (Container, database)

Operating System (Linux, Windows)

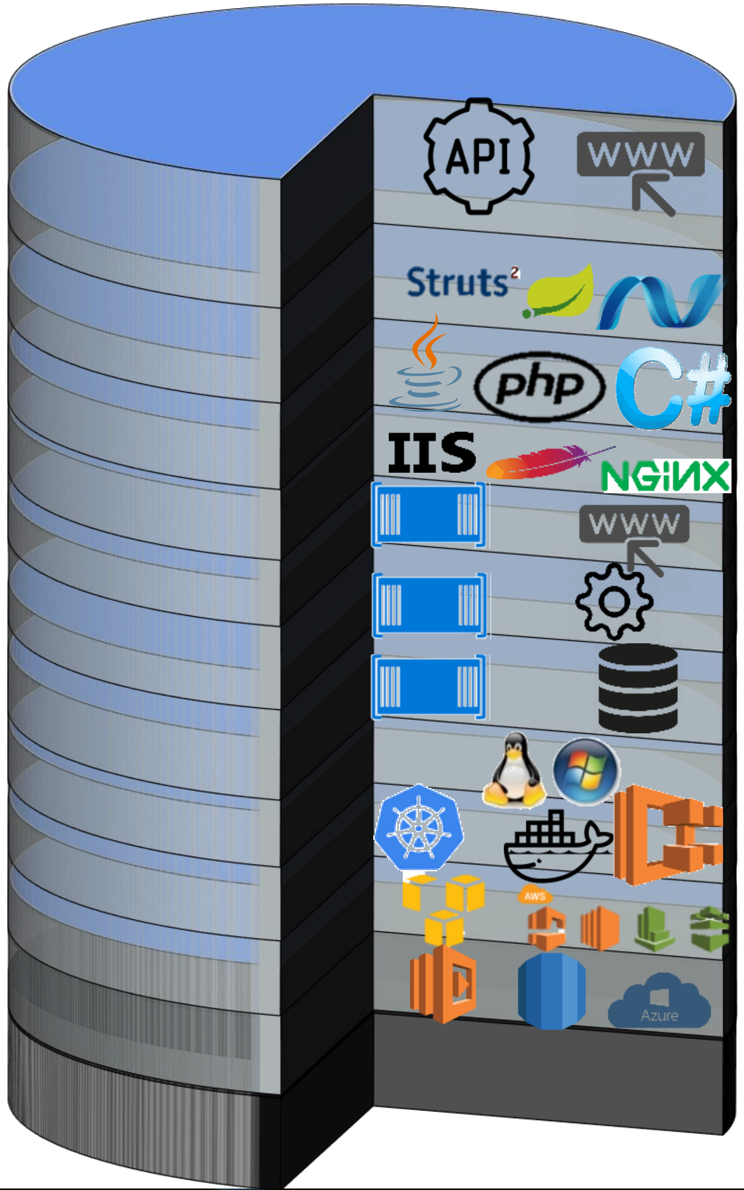
Clustering/Orchestration (CaaS, Swarm, Kubernetes)

Security Testing Needs to Go Down The Stack



- User Interface (WebApps, forms, logons, API's)
- Framework (Struts, Spring, .NET)
- Language (Java, PHP, .NET)
- AppServer (IIS, Apache, Nginx)
- Process UI (Container, presentation layer)
- Process App (Container, application processing)
- Process BackEnd (Container, database)
- Operating System (Linux, Windows)
- Clustering/Orchestration (CaaS, Swarm, Kubernetes)
- Networking (SDN, SecGroups)

Security Testing Needs to Go Down The Stack



User Interface (WebApps, forms, logons, API’s)

Framework (Struts, Spring, .NET)

Language (Java, PHP, .NET)

AppServer (IIS, Apache, Nginx)

Process UI (Container, presentation layer)

Process App (Container, application processing)

Process BackEnd (Container, database)

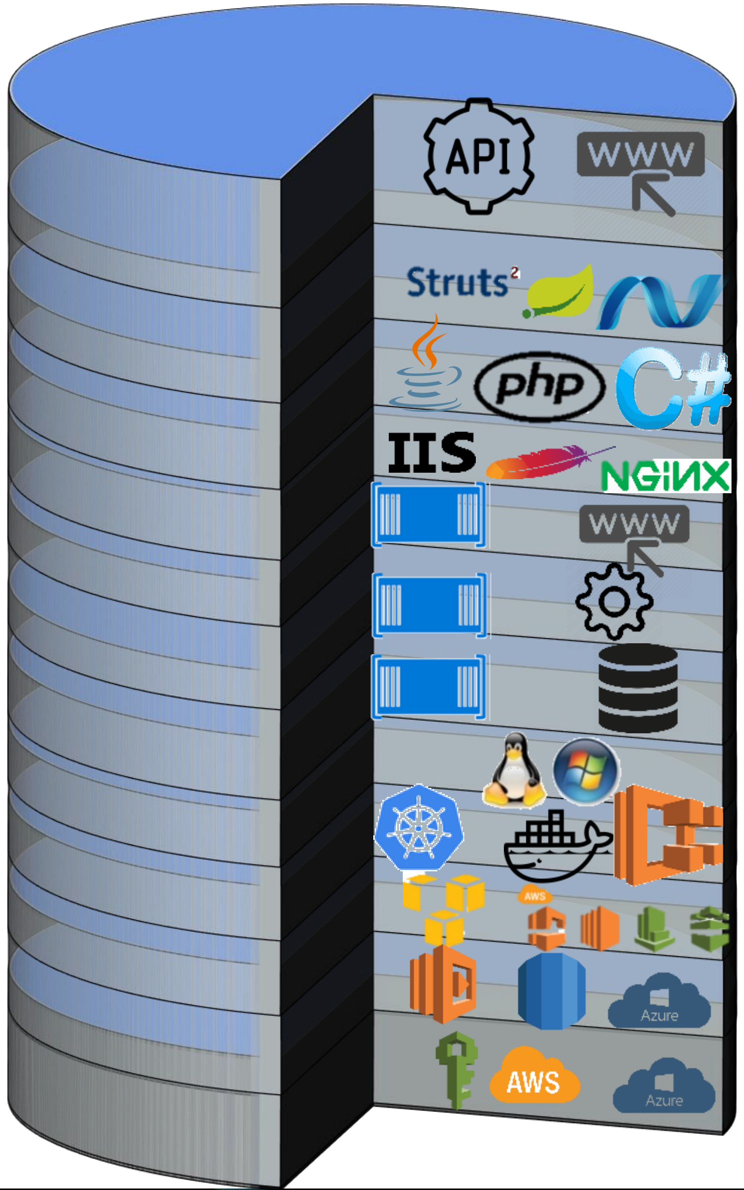
Operating System (Linux, Windows)

Clustering/Orchestration (CaaS, Swarm, Kubernetes)

Networking (SDN, SecGroups)

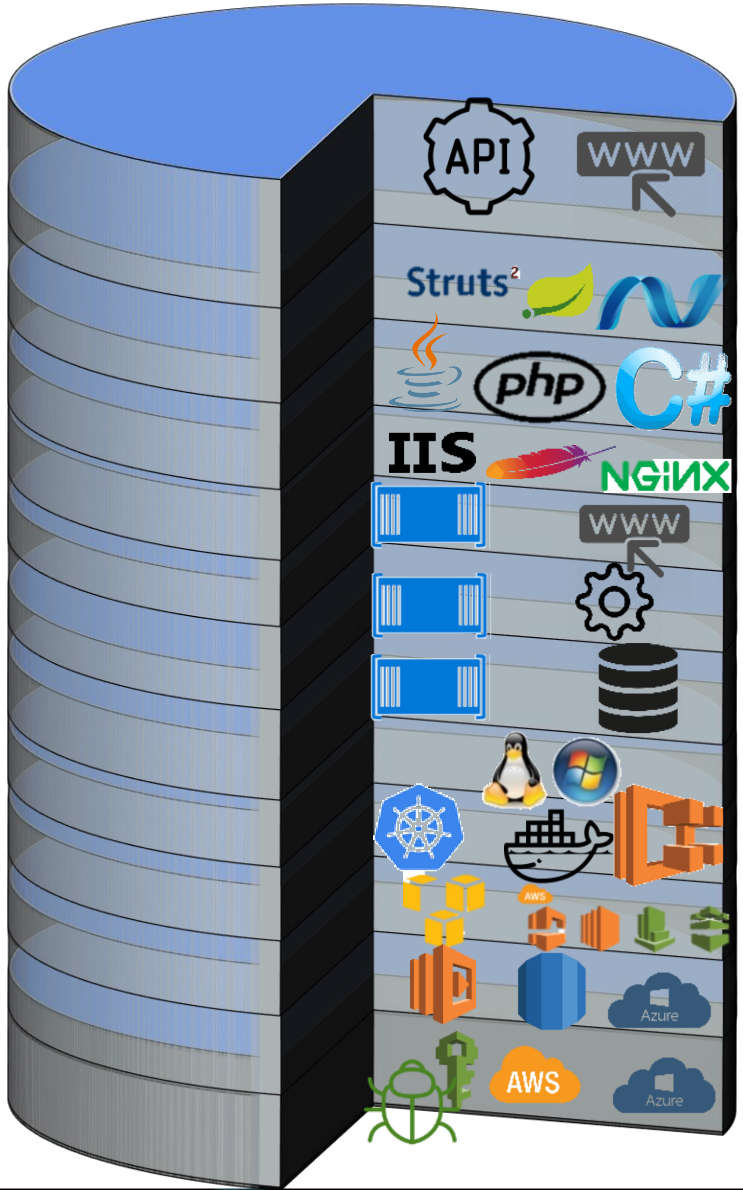
Cloud Platform

Security Testing Needs to Go Down The Stack



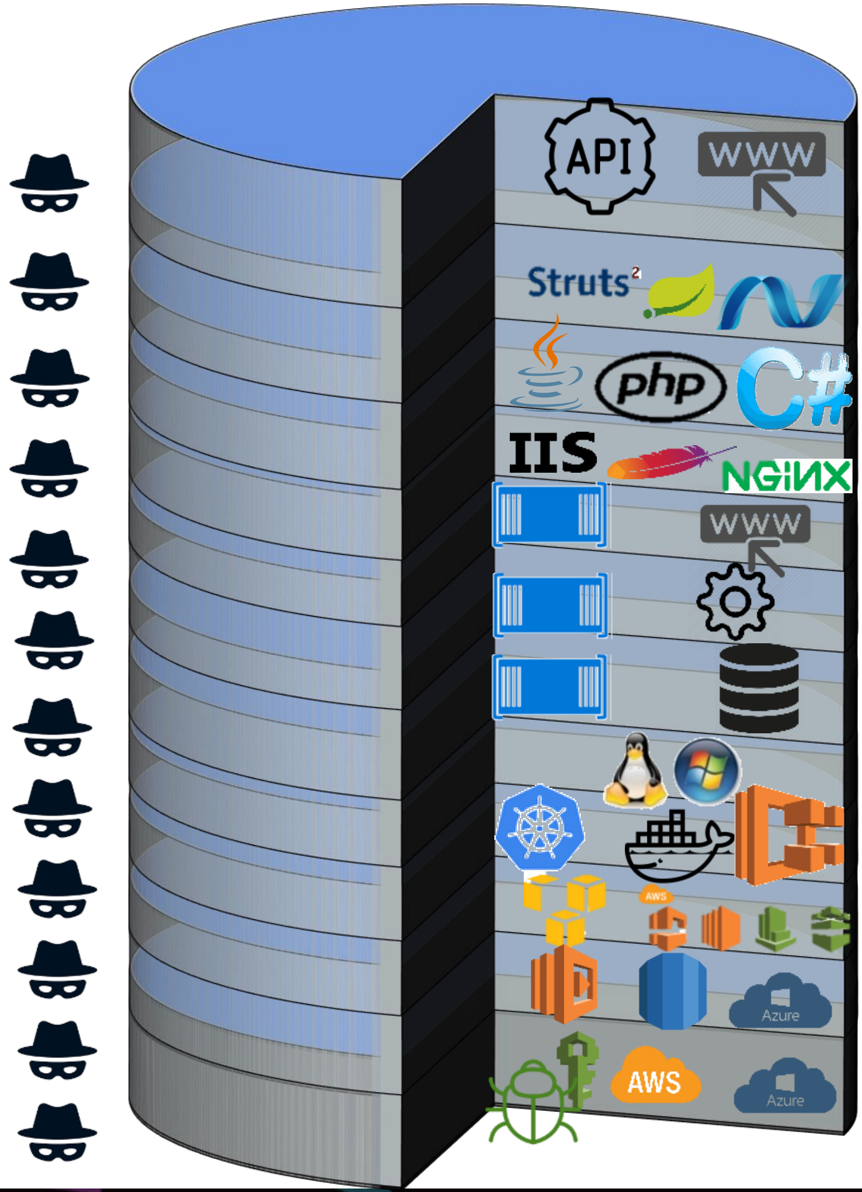
- User Interface (WebApps, forms, logons, API's)
- Framework (Struts, Spring, .NET)
- Language (Java, PHP, .NET)
- AppServer (IIS, Apache, Nginx)
- Process UI (Container, presentation layer)
- Process App (Container, application processing)
- Process BackEnd (Container, database)
- Operating System (Linux, Windows)
- Clustering/Orchestration (CaaS, Swarm, Kubernetes)
- Networking (SDN, SecGroups)
- Cloud Platform
- Core Infrastructure

Security Testing Needs to Go Down The Stack



- User Interface (WebApps, forms, logons, API's)
- Framework (Struts, Spring, .NET)
- Language (Java, PHP, .NET)
- AppServer (IIS, Apache, Nginx)
- Process UI (Container, presentation layer)
- Process App (Container, application processing)
- Process BackEnd (Container, database)
- Operating System (Linux, Windows)
- Clustering/Orchestration (CaaS, Swarm, Kubernetes)
- Networking (SDN, SecGroups)
- Cloud Platform
- Core Infrastructure

Security Testing Needs to Go Down The Stack



User Interface (WebApps, forms, logons, API's)

Framework (Struts, Spring, .NET)

Language (Java, PHP, .NET)

AppServer (IIS, Apache, Nginx)

Process UI (Container, presentation layer)

Process App (Container, application processing)

Process BackEnd (Container, database)

Operating System (Linux, Windows)

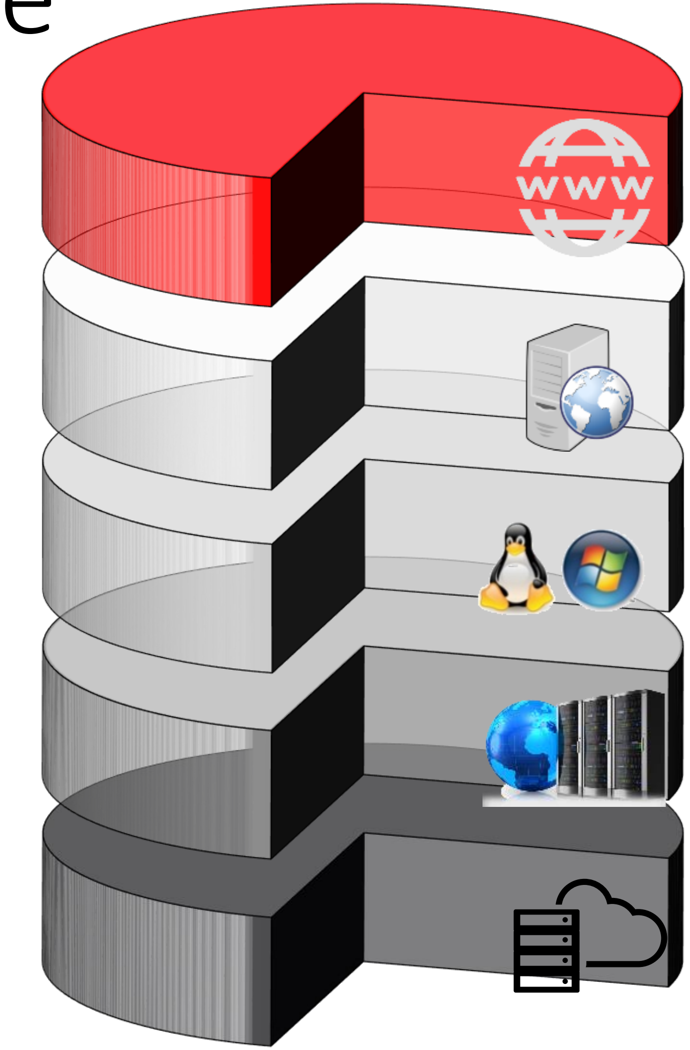
Clustering/Orchestration (CaaS, Swarm, Kubernetes)

Networking (SDN, SecGroups)

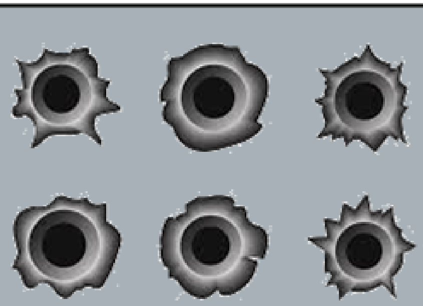
Cloud Platform

Core Infrastructure

Finesse



Right Decision
Wrong Decision



Lower Cost

Predictable

Even if a Web App/Service Pen Test not suitable for current technologies

Doesn’t really assess the threats

More North-South than East-West

Check Box

More considered

Requires expert capability, R&D

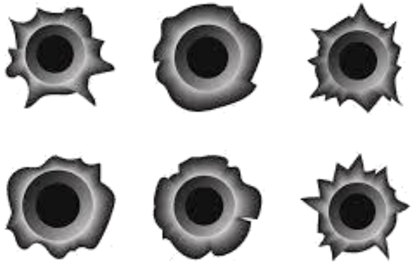
Requires understanding of the full stack incl implications of -aaS

Requires persistence in an ephemeral setting

Yes – it will cost more

Assurance, Validation & Compliance

There are Pen Tests & There are Pen Tests!

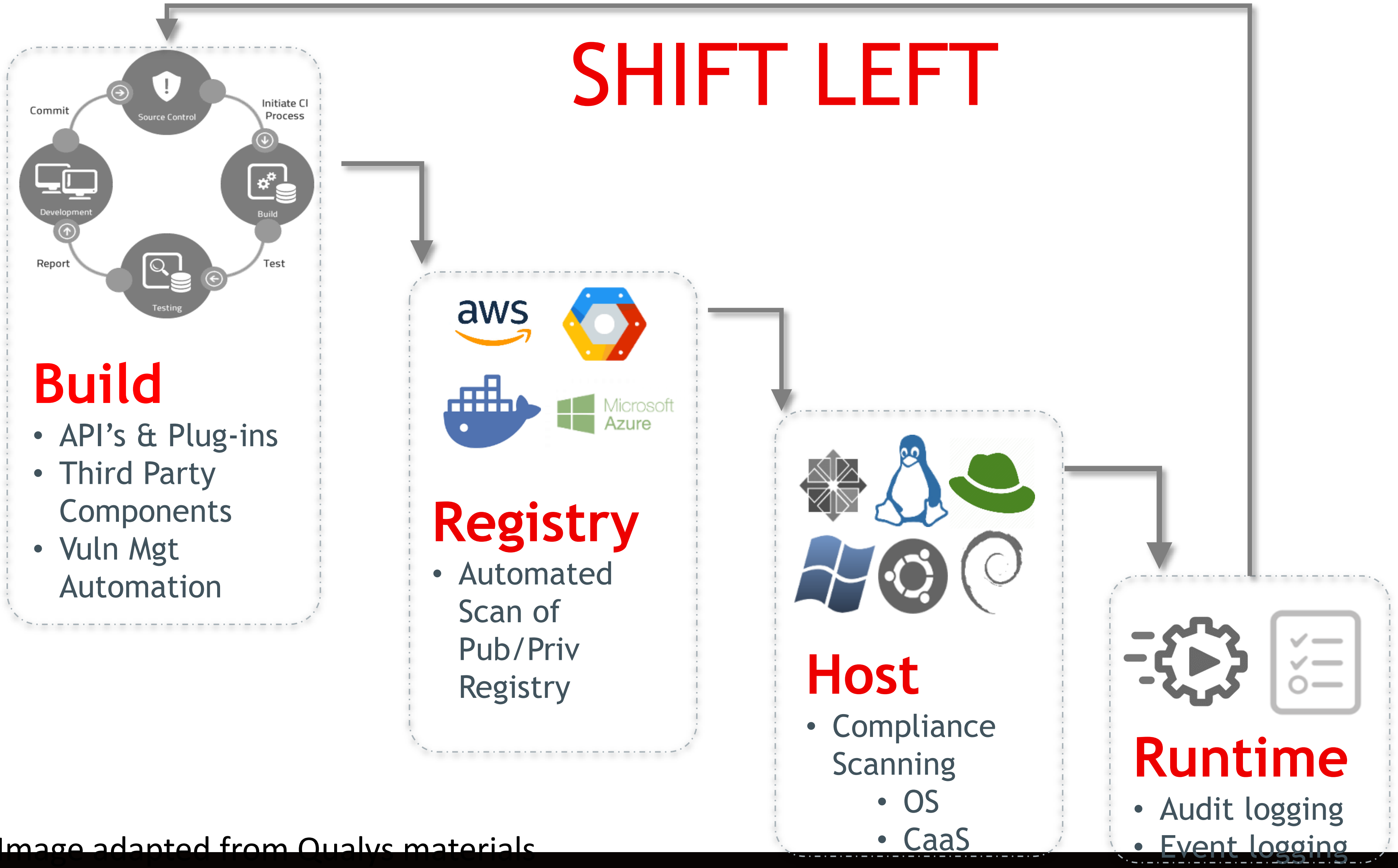


Lower Cost	More considered
Predictable	Requires expert capability, R&D
Even if a Web App/Service Pen Test not suitable for current technologies	Requires understanding of the full stack incl implications of -aaS
Doesn’t really assess the threats	Requires persistence in an ephemeral setting
More North-South than East-West	Yes – it will cost more
Check Box	Assurance, Validation & Compliance

Blue Team: Key Steps to App Container Security

1	**End-to-End Vulnerability Management**
2	Container Attack Surface Reduction
3	User Access Control
4	Hardening the Host OS & the Container
5	SDLC Automation (DevOps)

SHIFT LEFT



Build

- API’s & Plug-ins
- Third Party Components
- Vuln Mgt Automation

Registry

- Automated Scan of Pub/Priv Registry

Host

- Compliance Scanning
  - OS
  - CaaS

Runtime

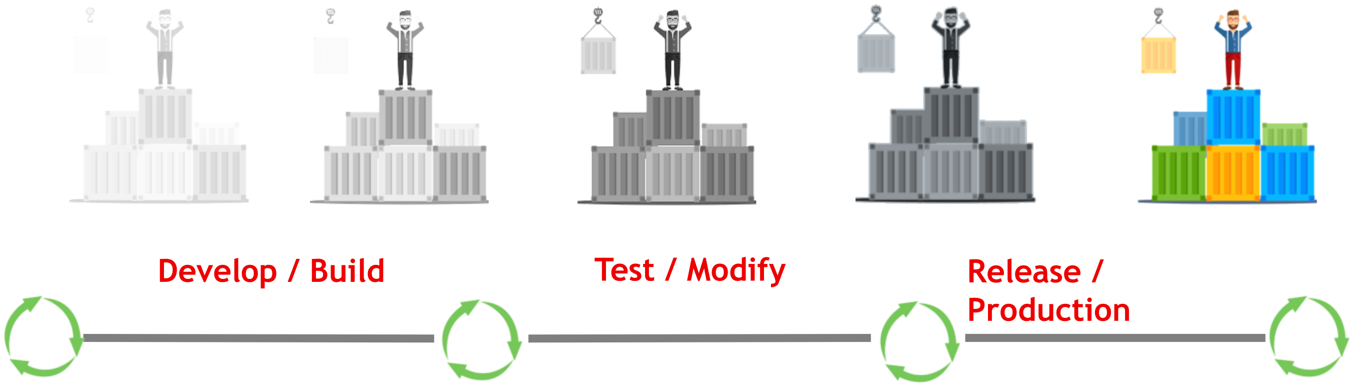
- Audit logging
- Event logging

Image adapted from Qualys materials

[illegible] mary

Adapted from: Ten Basic Steps To Secure Software Containers, Instructions For Safely Developing And Deploying Software In Containers, by Amy DeMartine and Dave Bartoletti, April 14, 2017

mary

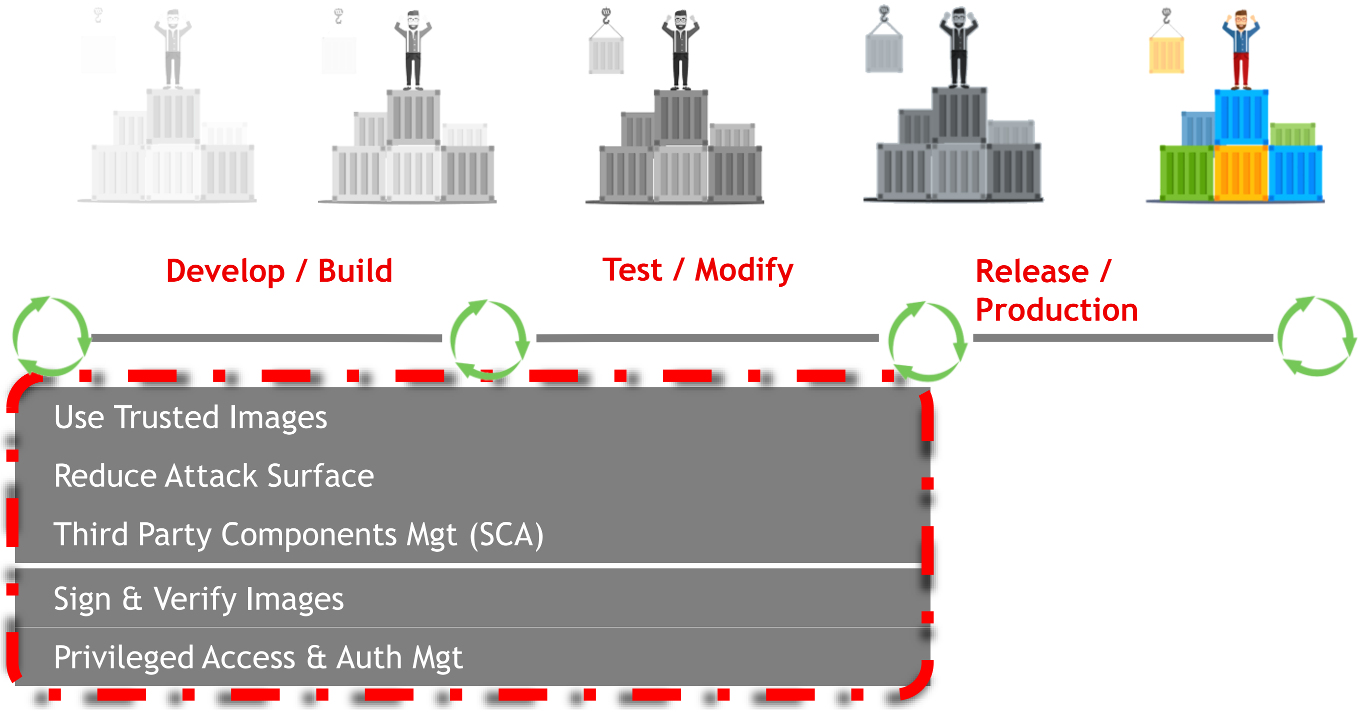


Develop / Build

Test / Modify

Release / Production

Adapted from: Ten Basic Steps To Secure Software Containers, Instructions For Safely Developing And Deploying Software In Containers
by Amy DeMartine and Dave Bartoletti, April 14, 2017



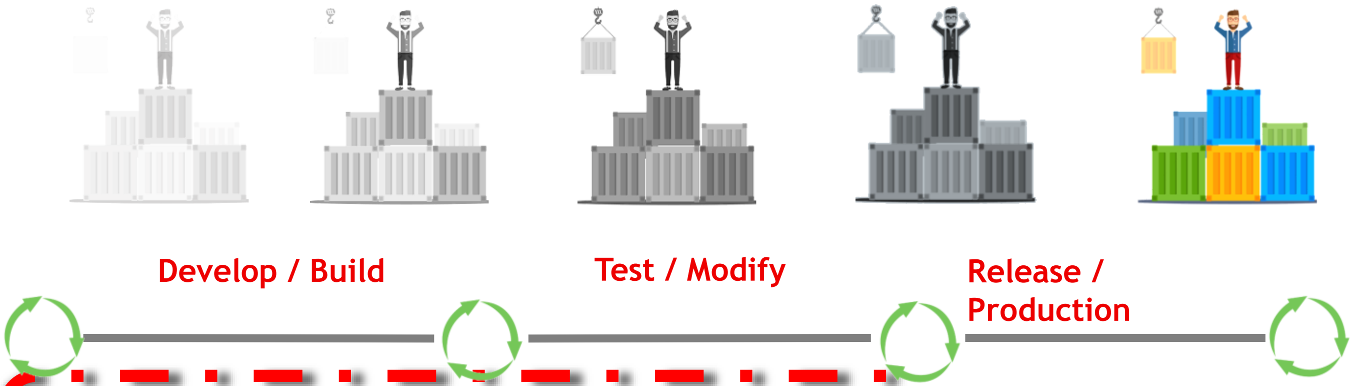
Develop / Build

Test / Modify

Release / Production

- Use Trusted Images
- Reduce Attack Surface
- Third Party Components Mgt (SCA)
- Sign & Verify Images
- Privileged Access & Auth Mgt

Adapted from: Ten Basic Steps To Secure Software Containers, Instructions For Safely Developing And Deploying Software In Containers by Amy DeMartine and Dave Bartoletti, April 14, 2017

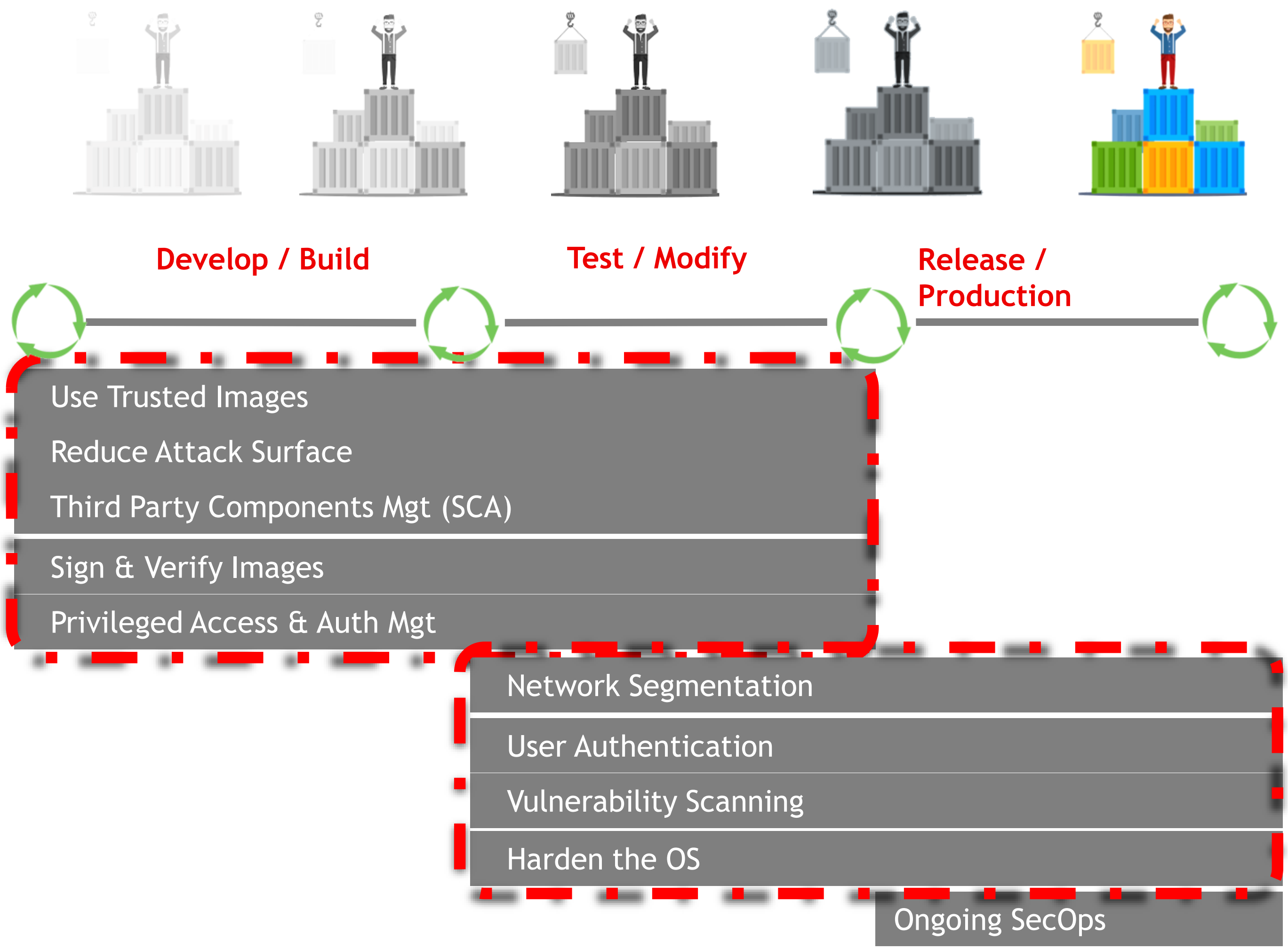


- Use Trusted Images
- Reduce Attack Surface
- Third Party Components Mgt (SCA)
- Sign & Verify Images
- Privileged Access & Auth Mgt

- Network Segmentation
- User Authentication
- Vulnerability Scanning
- Harden the OS

Adapted from: Ten Basic Steps To Secure Software Containers, Instructions For Safely Developing And Deploying Software In Containers, by Amy DeMartine and Dave Bartoletti, April 14, 2017

mary



Develop / Build

Test / Modify

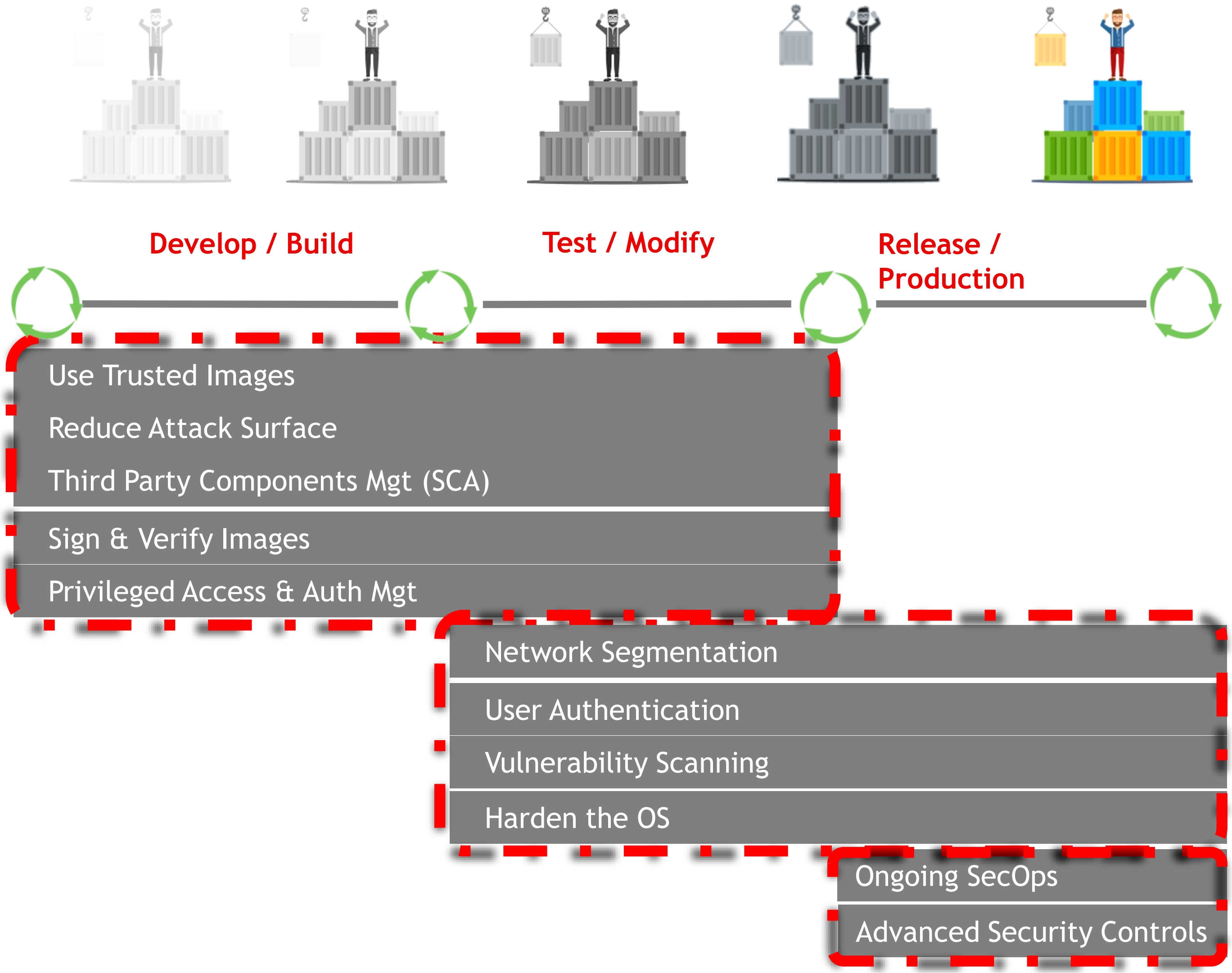
Release / Production

- Use Trusted Images
- Reduce Attack Surface
- Third Party Components Mgt (SCA)
- Sign & Verify Images
- Privileged Access & Auth Mgt

- Network Segmentation
- User Authentication
- Vulnerability Scanning
- Harden the OS

Ongoing SecOps

Adapted from: Ten Basic Steps To Secure Software Containers, Instructions For Safely Developing And Deploying Software In Containers by Amy DeMartine and Dave Bartoletti, April 14, 2017



Develop / Build

Test / Modify

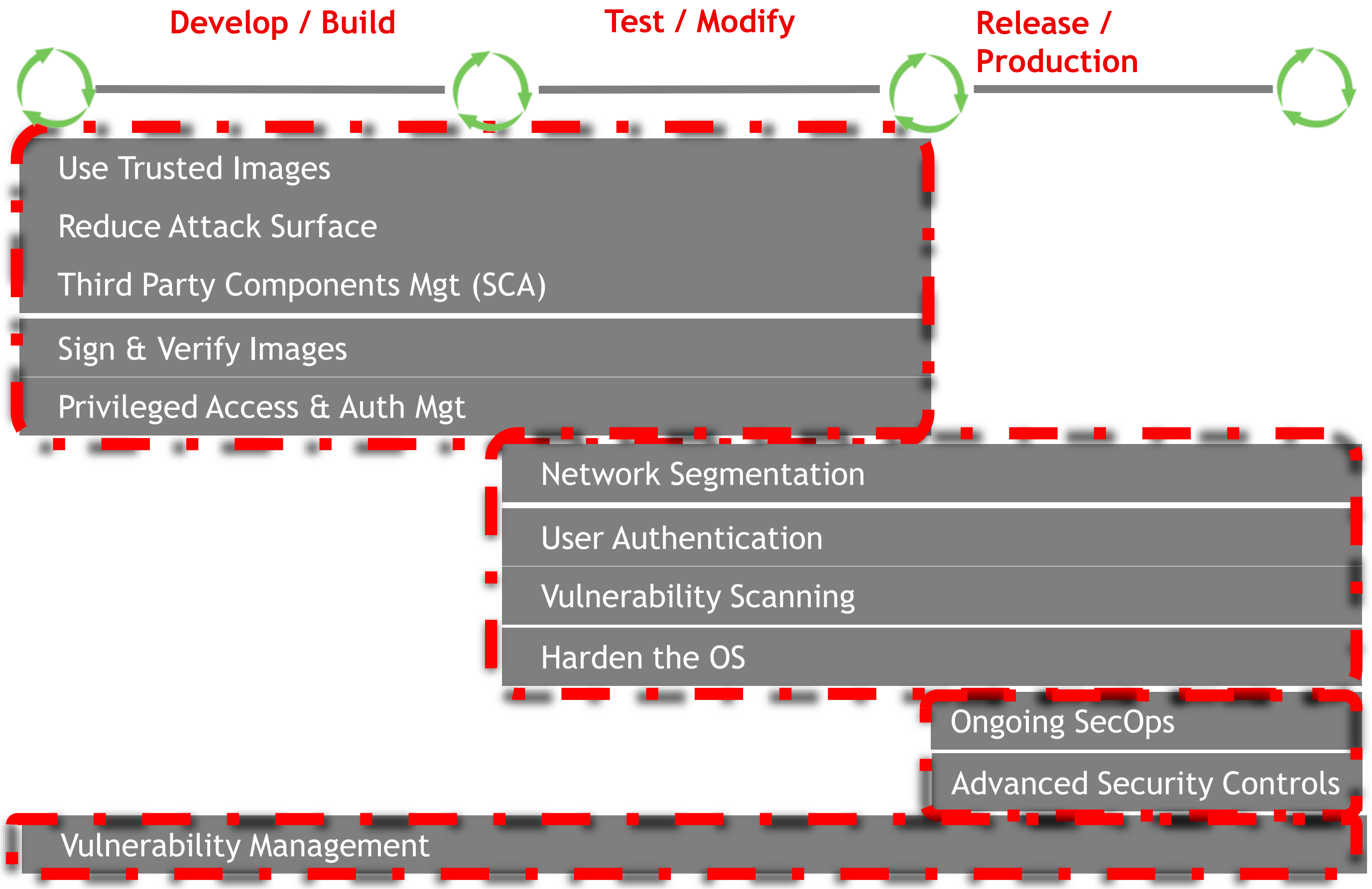
Release / Production

- Use Trusted Images
- Reduce Attack Surface
- Third Party Components Mgt (SCA)
- Sign & Verify Images
- Privileged Access & Auth Mgt

- Network Segmentation
- User Authentication
- Vulnerability Scanning
- Harden the OS

- Ongoing SecOps
- Advanced Security Controls

Adapted from: Ten Basic Steps To Secure Software Containers, Instructions For Safely Developing And Deploying Software In Containers, by Amy DeMartine and Dave Bartoletti, April 14, 2017



Develop / Build

Test / Modify

Release / Production

Use Trusted Images

Reduce Attack Surface

Third Party Components Mgt (SCA)

Sign & Verify Images

Privileged Access & Auth Mgt

Network Segmentation

User Authentication

Vulnerability Scanning

Harden the OS

Ongoing SecOps

Advanced Security Controls

Vulnerability Management

Adapted from: Ten Basic Steps To Secure Software Containers, Instructions For Safely Developing And Deploying Software In Containers, by Amy DeMartine and Dave Bartoletti, April 14, 2017

Recap

1	**Serverless, Microservices and Container Security**	4	**CI/CD Integration for Automated Security**
2	Key Implications for Penetration Testing Programs		End to End Vulnerability Management
3	Key Security features for Container Deployments		Continuous Monitoring, Governance & Compliance Reporting

Sense of Security

Thank You!

murrayg@senseofecurity.com.au

Security, it's all we do. Knowledge, Experience & Trust.

Sense of Security Pty Ltd
ABN 14 098 237 908

Sydney
Level 8, 59 Goulburn Street
Sydney NSW 2000

Melbourne
Level 15, 401 Docklands Drive
Docklands VIC 3008

Tel. 1300 922 923
Intl. +61 2 9290 4444
www.senseofsecurity.com.au

@ITSecurityAU