Emerging Approaches in a Cloud-Connected Enterprise: Containers, Microservices and Cloud Security

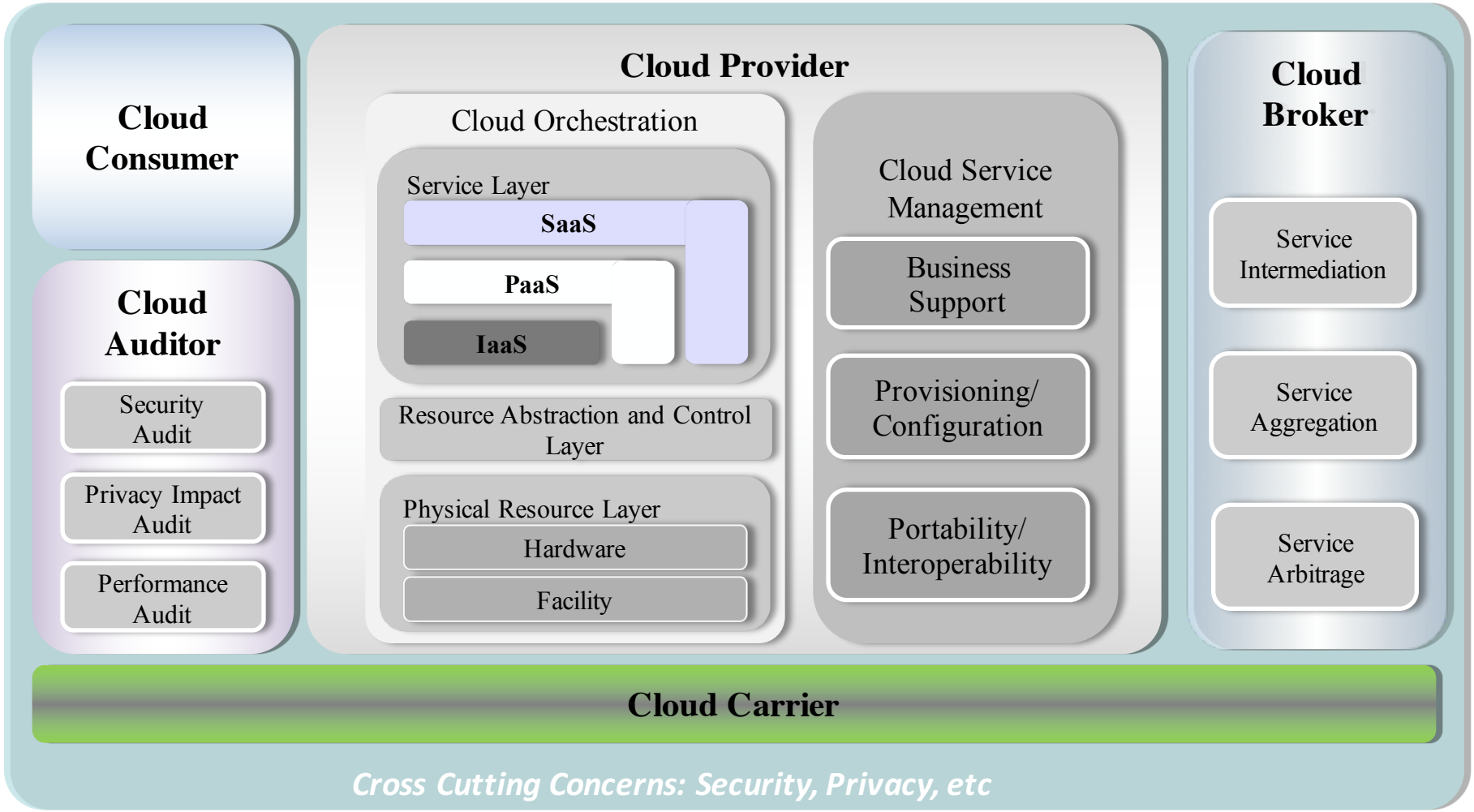
Anil Karmel
Co-Chair, NIST Cloud Security Working Group
Co-Founder and CEO, C2 Labs
akarmel@c2labs.com
@anilkarmel

C2 LABS
TAKE BACK CONTROL

Emerging Cloud Technologies and Trends
Cloud is Our Reality

- Evolving Cloud Models
  - Private Cloud (IaaS)
  - Public Cloud (SaaS, PaaS, IaaS)
  - Hybrid Cloud is becoming the defacto norm
- What About Security?
  - OPM Breach

NIST Cloud Computing Reference Architecture
SP500-292

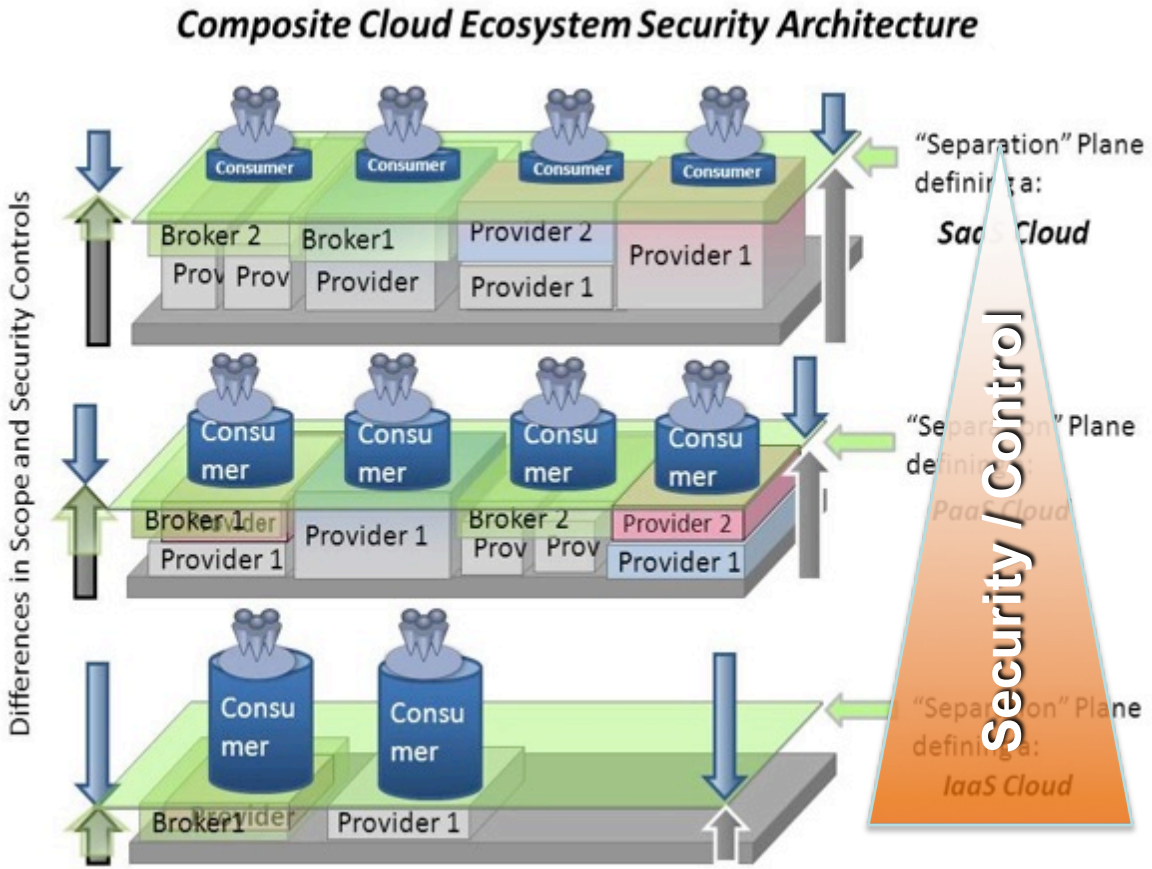


Cloud Demystified
What is a Cloud Ecosystem?

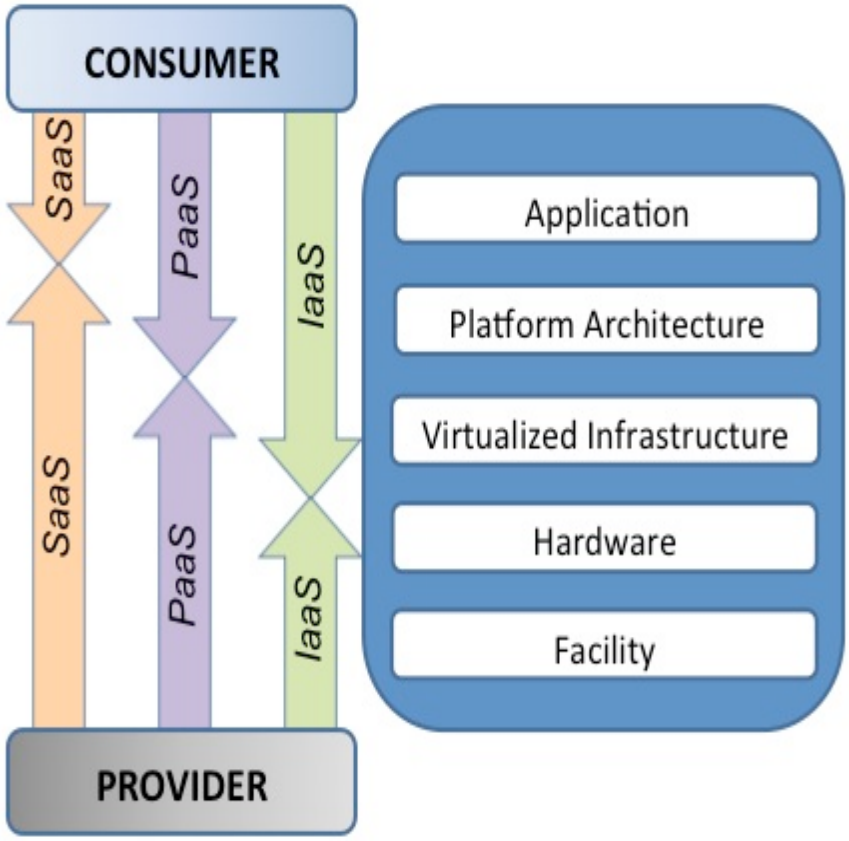
Software as a Service

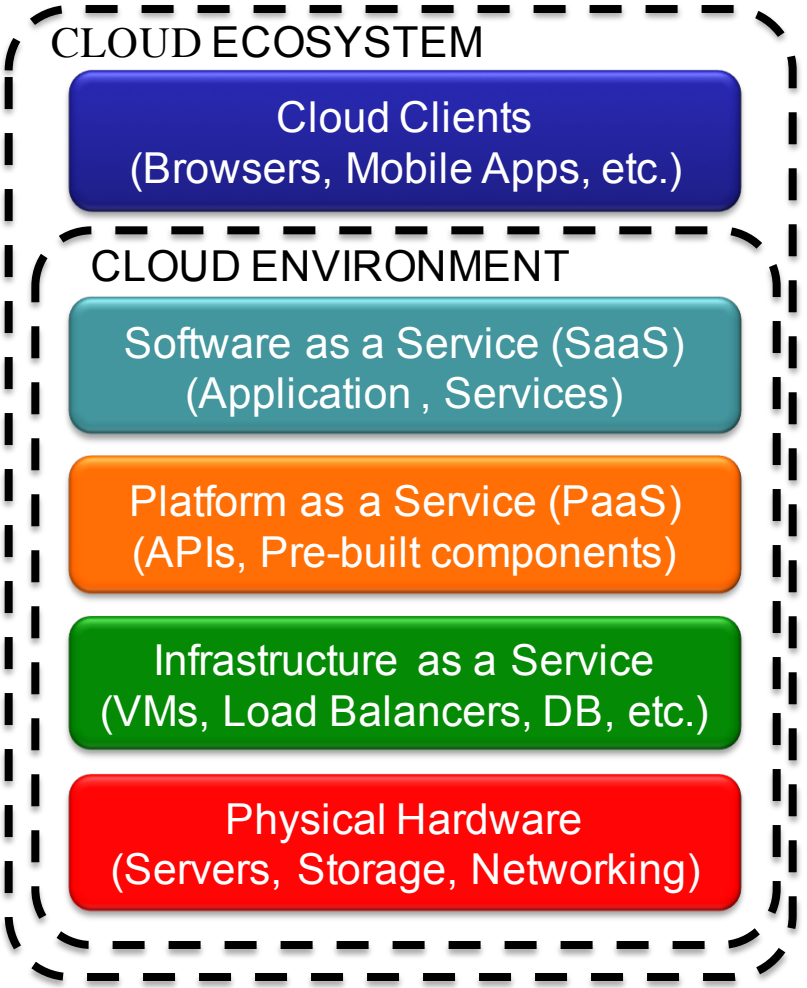
Platform as a Service

Infrastructure as a Service

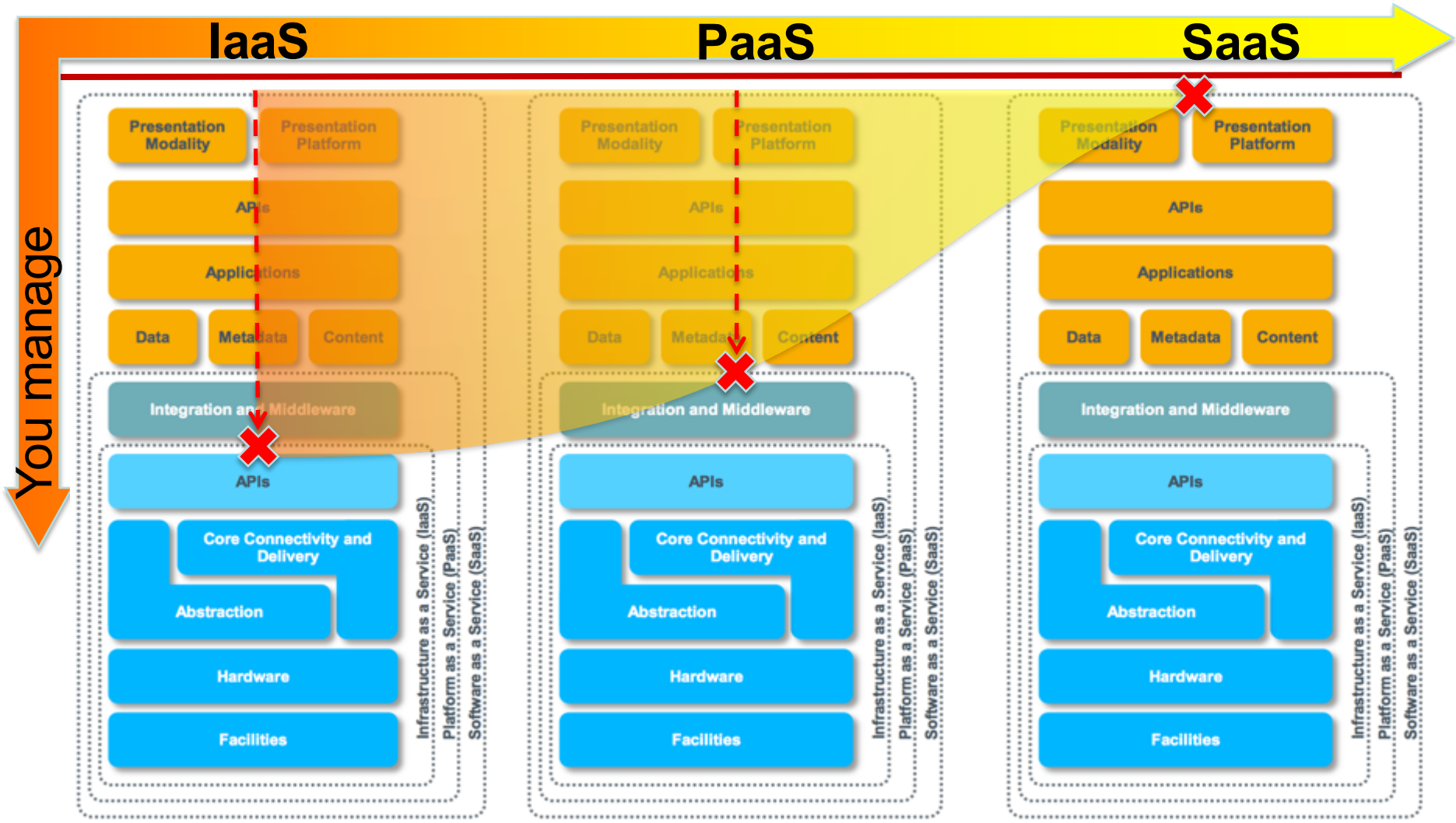


Distributed Architecture = Split Control / Responsibilities





What you can manage…



Stack image source: Cloud Security Alliance specification, 2009

Organizational Challenges
Modernizing IT

- Agility
  - Organizations are struggling to deliver more in a fiscally and resource constrained environment
- Flexibility
  - Existing IT investments are typically problematic to reconfigure or scale to meet new application demands
- Transparancy
  - Difficult to quantify the cost of optimizing legacy infrastructure to support new applications

Organizational Challenges
Modernizing IT – Cloud, Mobile, Social, Big Data

- Cloud
  - Powerful ROI story with real security challenges
- Mobile
  - BYOD with Mobile Application Management result in security and privacy concerns
- Social
  - Agency data inadvertently ends up on public social networks via geotagging
- Big Data
  - Unstructured data unveils actionable intelligence but what about the Mosaic effect?

How does you balance time to market, cost concerns, security, manageability and risk in the move to a cloud-connected enterprise?

How do we revolutionize our investments?
Software-Defined IT

- **REDEFINE CONTEXT**
  - *Who* is the user?
  - *What* data are they trying to access?
  - *Where* is the user and the data?
  - *How* are they accessing the information?

Context Aware IT

Level of assurance of the data defines the required level of trust

Context Aware IT
Data Centric Approach

- Understand your Data
  - Identify and understand the value of the data in your organization
- Decompose Your Data
  - Break down applications and data into building blocks
- Monitor Your Data
  - Understand Risk to your Data using the Risk Management Framework for Cloud
  - Employ Continuous Monitoring of your Systems to identify and limit the damage an adversary has to your data

Emerging Cloud Technologies and Trends
Microservices and Containers

- Microservices
  - Decompose Complex Applications into Small, Independent Processes communicating with each other using language-agnostic API's
  - Highly Decoupled and Modular with services organized around capabilities (e.g. User Interface, Billing)
  - Allows for Continuous Integration

- Containers
  - Much like Virtualization abstracts the Operating System from Hardware, Containers abstracts to Applications from the Operating System
  - Applications are isolated from other Applications on the same Operating System
  - Allows for Cloud Portability and Scale Up/Out
  - Security issues need to be evaluated and addressed in native container deployments

Container and Microservices Definition NIST SP800-180 (DRAFT)

NIST Special Publication 800-180 (DRAFT)

NIST Definition of Microservices, Application Containers and System Virtual Machines

Anil Karmel
C2 Labs, Inc.
Reston, VA

Ramaswamy Chandramouli
Michaela Iorga.
Computer Security Division
Information Technology Laboratory

This publication is available free of charge

February 2016

http://csrc.nist.gov/publications/drafts/800-180/sp800-180_draft.pdf

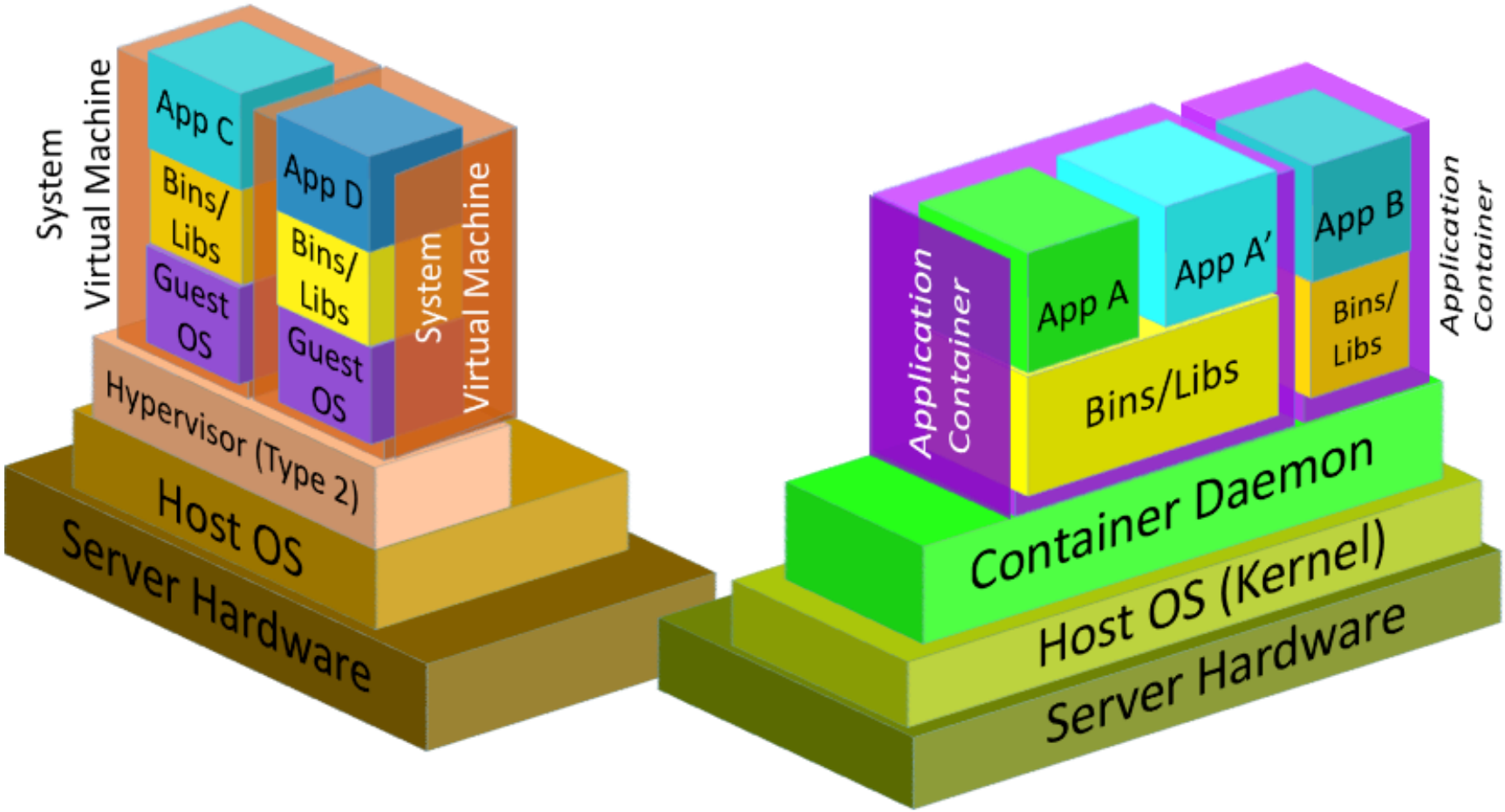
Definition of Microservices
NIST SP800-180 (DRAFT)

- **Microservices:** *A microservice is a basic element that results from the architectural decomposition of an application's components into loosely coupled patterns consisting of self-contained services that communicate with each other using a standard communications protocol and a set of well-defined APIs, independent of any vendor, product or technology.*
- Microservices are built around capabilities as opposed to services, builds on SOA and is implemented using Agile techniques. Microservices are typically deployed inside Application Containers.

Definition of Application Containers
NIST SP800-180 (DRAFT)

- **Application Containers:** *An Application Container is a construct designed to package and run an application or its' components running on a shared Operating System.*
- Application Containers are isolated from other Application Containers and share the resources of the underlying Operating System, allowing for efficient restart, scale-up or scale-out of applications across clouds. Application Containers typically contain Microservices.

Emerging Cloud Technologies and Trends
Virtual Machines vs Containers



Source: NIST SP800-180 (DRAFT)

Microservices and Containers Use Cases
Google

- "EVERYTHING at Google runs in a container"
  - Starts over 2 Billion Containers per week as of 2014

- http://www.theregister.co.uk/2014/05/23/google_containerization_two_billion/

Microservices and Containers Use Cases
NetFlix

- Best Practices for Designing a Microservices Architecture
  - Create a Separate Data Store for Each Microservice
  - Keep Code at a Similar Level of Maturity
  - Do a Separate Build for Each Microservice
  - Deploy in Containers
  - Treat Servers as Stateless

- https://www.nginx.com/blog/microservices-at-netflix-architectural-best-practices/

NIST and CSA Partnership

Best Practices for Application Containers and Microservices

- NIST and CSA have joined forces to define best practices for Application Containers and Microservices (ACM)
  - CSA ACM Members have joined the NIST ACM Cloud Security Working Group
  - NIST artifacts will serve as the foundation for CSA ACM work
    - NIST SP 800-180: NIST Definition of Microservices, Application Containers and System Virtual Machines
    - Additional artifacts from NIST ACM Working Group

NIST Application Container and Microservices (ACM) Charter

- NIST ACM Working Group Charter: http://collaborate.nist.gov/twiki-cloud-computing/bin/view/CloudComputing/ApplicationContainersAndMicroservices
- Objectives
  - Aggregate and document application containers and microservices use cases;
  - Research and document the challenges of implementing and managing application containers and microservices
  - Identify process-based and end-product based threats to container deployment and container stacks respectively;
  - Provide security recommendations for adopting state of the art practices for mitigating the identified threats.

NIST Application Container and Microservices (ACM) Charter

- NIST ACM Working Group Charter: http://collaborate.nist.gov/twiki-cloud-computing/bin/view/CloudComputing/ApplicationContainersAndMicroservices
- Deliverables
  - Document the challenges of implementing and managing application containers, with a particular focus on deployment and run-time security threats to application containers and microservices
  - Document the security recommendations for mitigating identified deployment and run-time security threats to application containers and microservices

NIST Application Container and Microservices (ACM) Progress to Date

- NIST Progress to Date
  - Documented Challenges per a Use Case Template
  - Created Methodology to Score Challenges
  - Currently Scoring Challenges to determine which challenges have the highest impact

- NIST Path Forward
  - Finalize Challenges and publish document shortly
  - Begin work on Best Practices document mapped to Challenges document

CSA Application Container and Microservices (ACM) Charter

- CSA ACM Working Group Charter:
- https://docs.google.com/document/d/1k_82U2BFgvA9j06MaI96VZAoMIYFmAg8HoAFA2GEA1Y/edit
- Objectives – Q1 2017
  - Create an Application Container Implementation Guidance document that includes:
    - Overview of the Application Container threat landscape
    - Unique security issues/concerns introduced by Application Containers
    - Application Container host hardening and security recommendations
    - Application Container hardening and security recommendations
    - Security considerations for application containers in a DevOps environment
    - Define Microservices secure development standards and governance

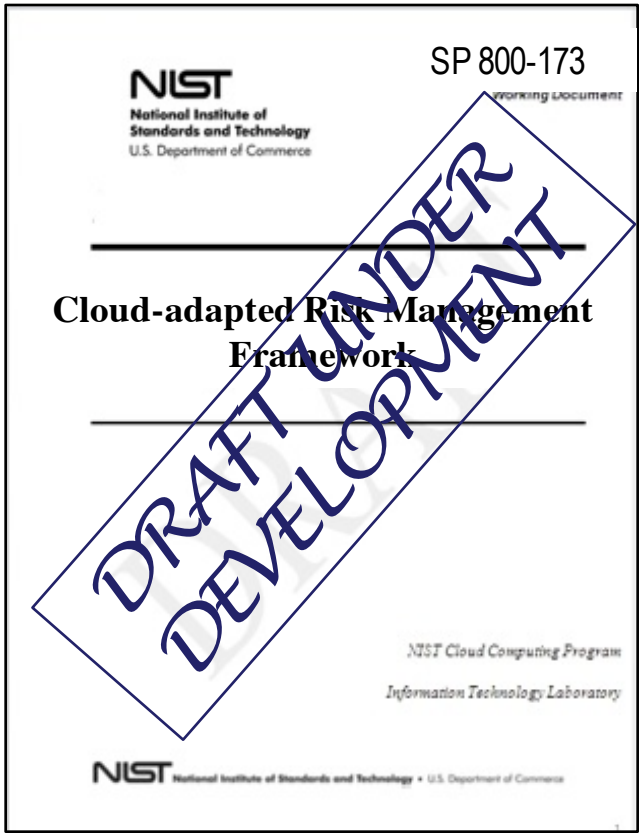
CSA Application Container and Microservices (ACM) Charter

- CSA ACM Working Group Charter:
- https://docs.google.com/document/d/1k_82U2BFgvA9j06MaI96VZAoMIYFmAg8HoAFA2GEA1Y/edit
- Objectives – Q2 2017
  - Create a Microservices Implementation Guidance document that includes:
    - Similarities and Differences between a Services Oriented Architecture (SOA) and a Microservices Architecture
    - Best Practices for implementing a Microservices Architecture for Cloud-native applications
    - Best Practices for decomposing monolithic applications into Microservices

NIST and CSA ACM Working Group
Call for Volunteers

- Email us and we'll get you connected.
  - Anil Karmel, Co-Chair, NIST Cloud Security Working Group, Co-Founder and CEO, C2 Labs
    - akarmel@c2labs.com
  - Andrew Wild, CISO, QTS Data Centers
    - andrew.wild@qtsdatacenters.com

NIST's Current Work
Risk Management Framework for Cloud Ecosystems



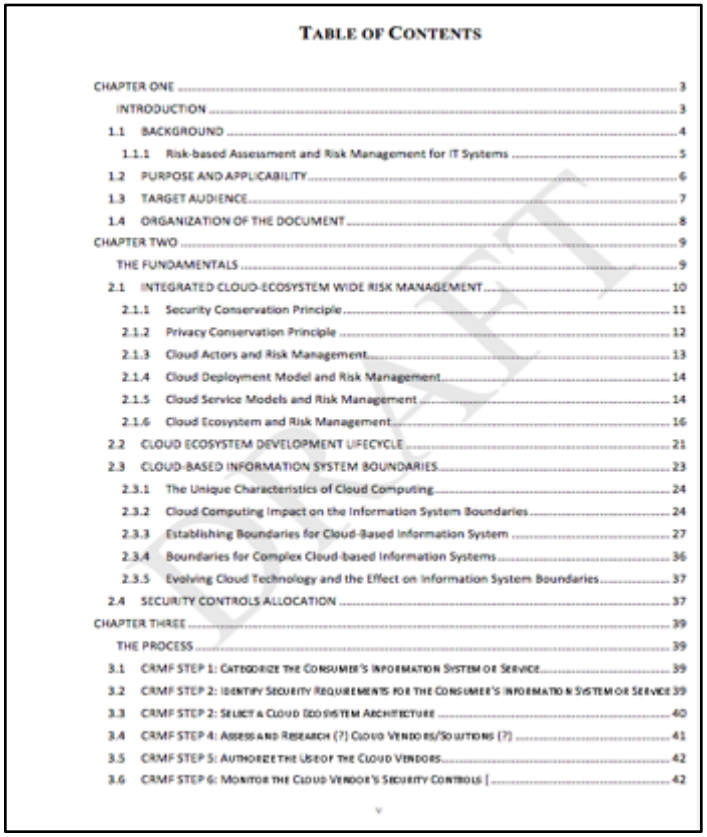
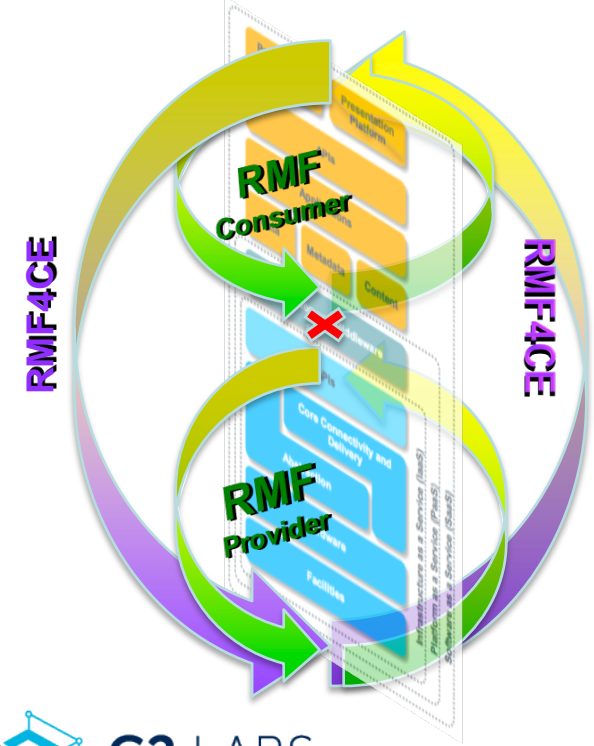
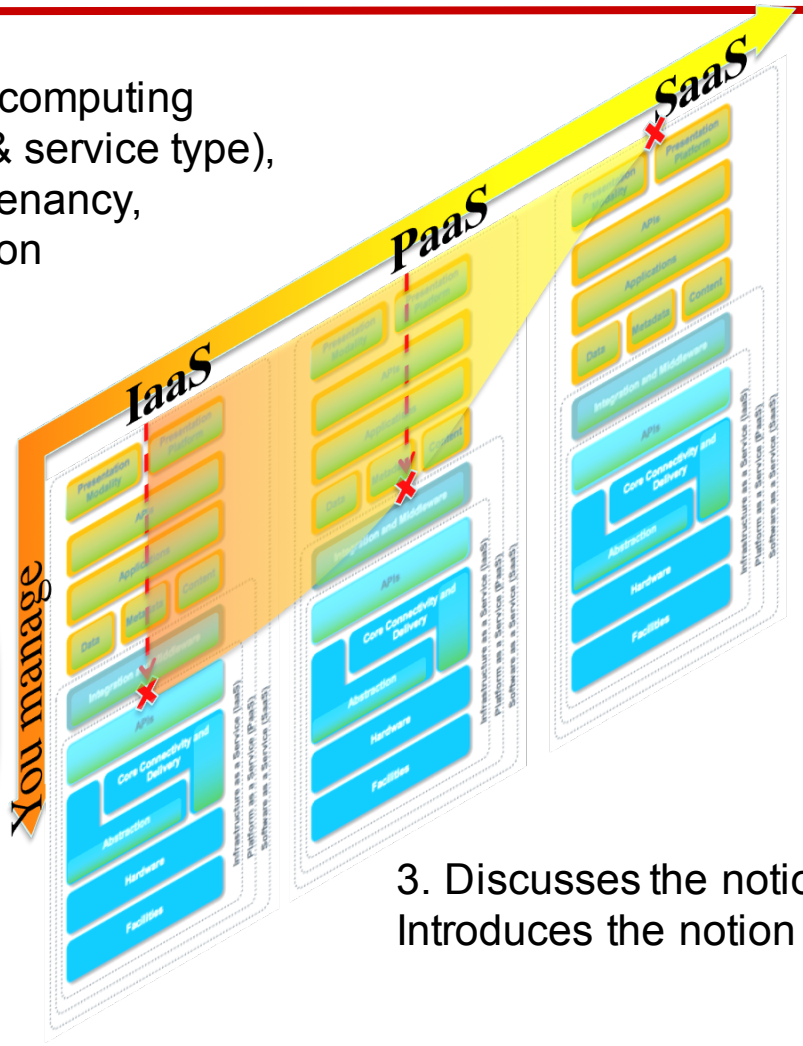


TABLE OF CONTENTS

Risk Management Framework for Cloud Ecosystems (RMF4CE): SP800-173

1. Follows NIST RMF structure
2. Discusses the impact of cloud computing architecture (deployment model & service type), and cloud characteristics (multi-tenancy, resource-pooling, elasticity, etc.) on "Information System Boundary".





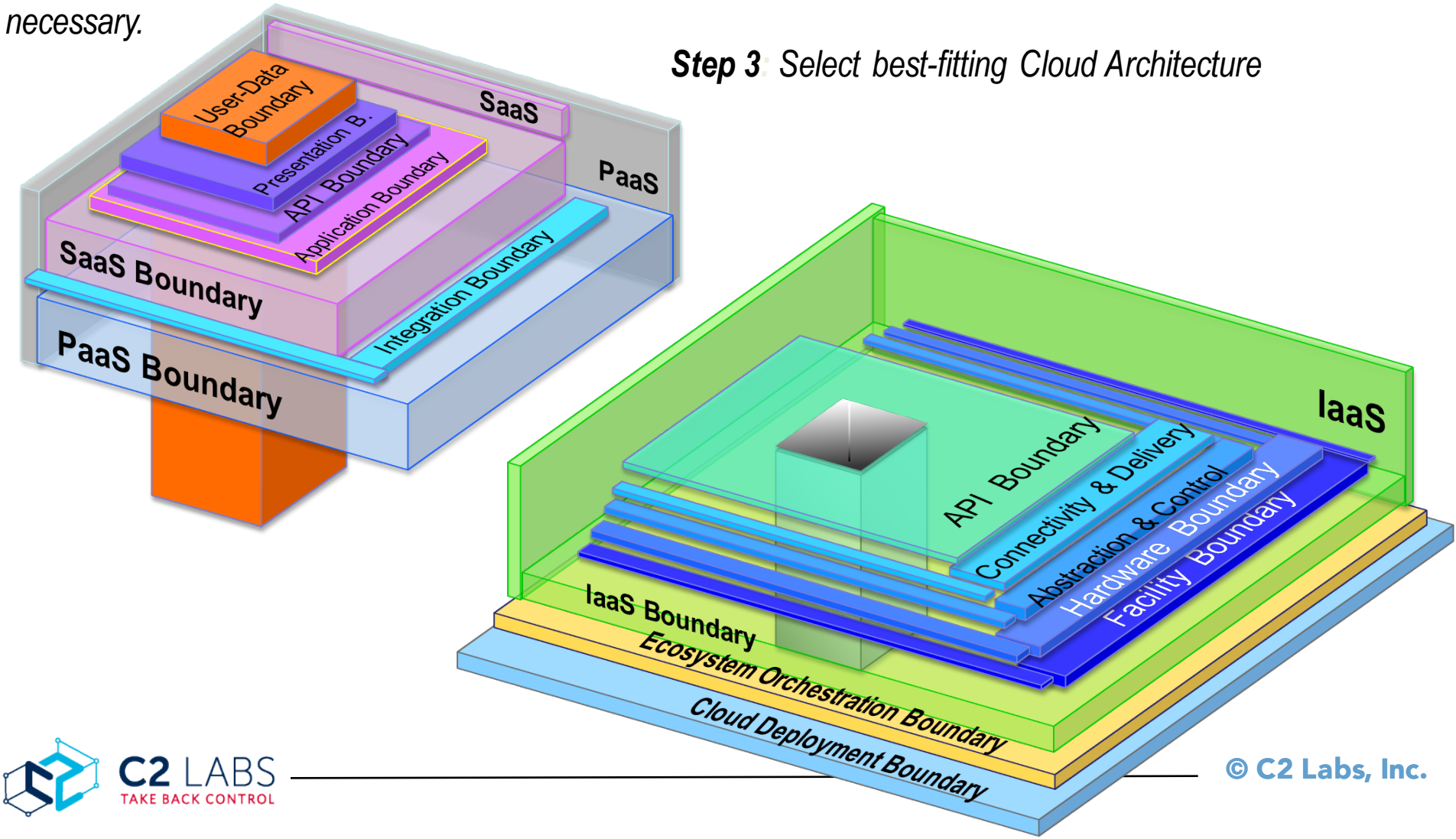
3. Discusses the notion of TRUST, and Introduces the notion of TRUST BOUNDARY

Risk Management Framework for Cloud Ecosystems (RMF4CE): SP800-173

Step 1: *Categorize Federal Information System*

Step 2*: Identify Security Requirements, perform a Risk Assessment & select Security Controls deemed necessary.*

Step 3*: Select best-fitting Cloud Architecture*

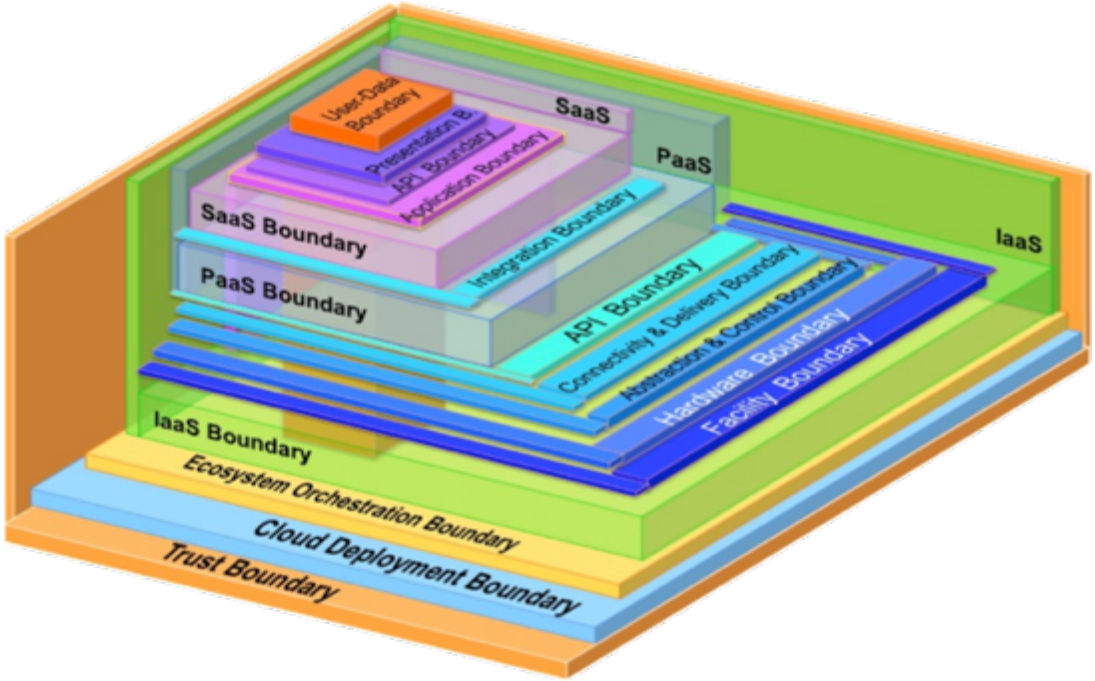


C2 LABS
TAKE BACK CONTROL

Risk Management Framework for Cloud Ecosystems (RMF4CE): SP800-173

***Step 4**: Assess Service Provider(s) & Broker (if applicable) → leverage FedRAMP P-ATOs or Agency-ATOs, or assess the controls → build necessary TRUST that the residual risk is acceptable*

***Step 5**: Authorize Use of Service → negotiate SLAs & Security SLA*



Step 6: Monitor Service Provider(s) (on-going, near- real- time); Repeat process as necessary

Thank you!

Anil Karmel
akarmel@c2labs.com
@anilkarmel

C2 LABS
TAKE BACK CONTROL