



# Reacquaint with F5 Solutions

DAVIS LI

13 SEPTEMBER 2022



F5 TODAY:

“Deploy, Deliver, Defend, Distribute  
every app and API, anywhere”

# THE ‘Ds’ of



Web Server (NGINX),  
App Server (NGINX).  
Deploy virtual apps to  
fleets of remote Edges

Contextual traffic steering, GSLB,  
Zero Trust App Access, Protocol  
Enhancement, SSL Offloading,  
Ingress Controller, API Gateway,  
CDN, Service Mesh.

## DELIVER

L3-L7 OSI Layers



L3-L7, & “L8” OSI Layers

Web App and API Protection, L8 Bot  
and functionality-abuse Defense,  
Machine-Learning Based L3-L7 DoS  
Protection, SSL Orchestrator,  
Intelligence Feeds (IP, URLs, Device  
ID).

**Consistent App Delivery & Security Platform across Physical Data Centres, Private Clouds, Hybrid-Multi-Cloud, & Remote Edges**

# THE 'Ds' of



Complexity is the enemy of security and innovation  
Edges can be managed as 'fleets'  
Unify security policies for distributed apps from a SaaS control plane  
Bot Protection/WAAP as a Service



**F5 Today:** Visibility across app services from end user to backend apps

**F5 Today:** Differentiate between good & bad users/bots

**F5 Vision:** Deploy, Deliver, Defend, Distribute Apps based on data fed into your feedback-looped decision engines

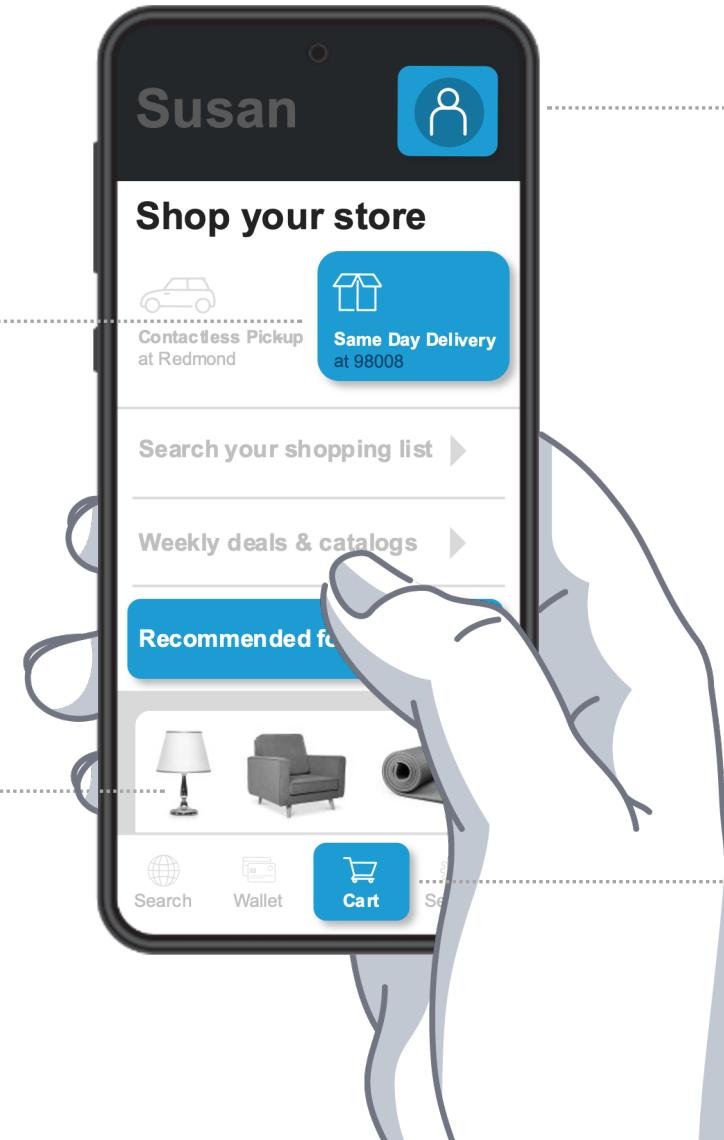
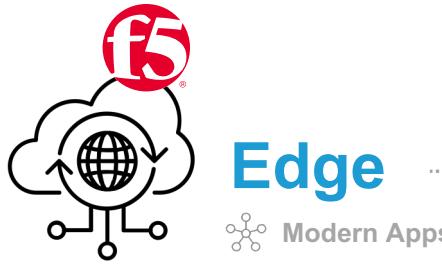
**F5 TODAY:**  
**“Deploy, Deliver, Defend, Distribute  
every app and API, anywhere”**



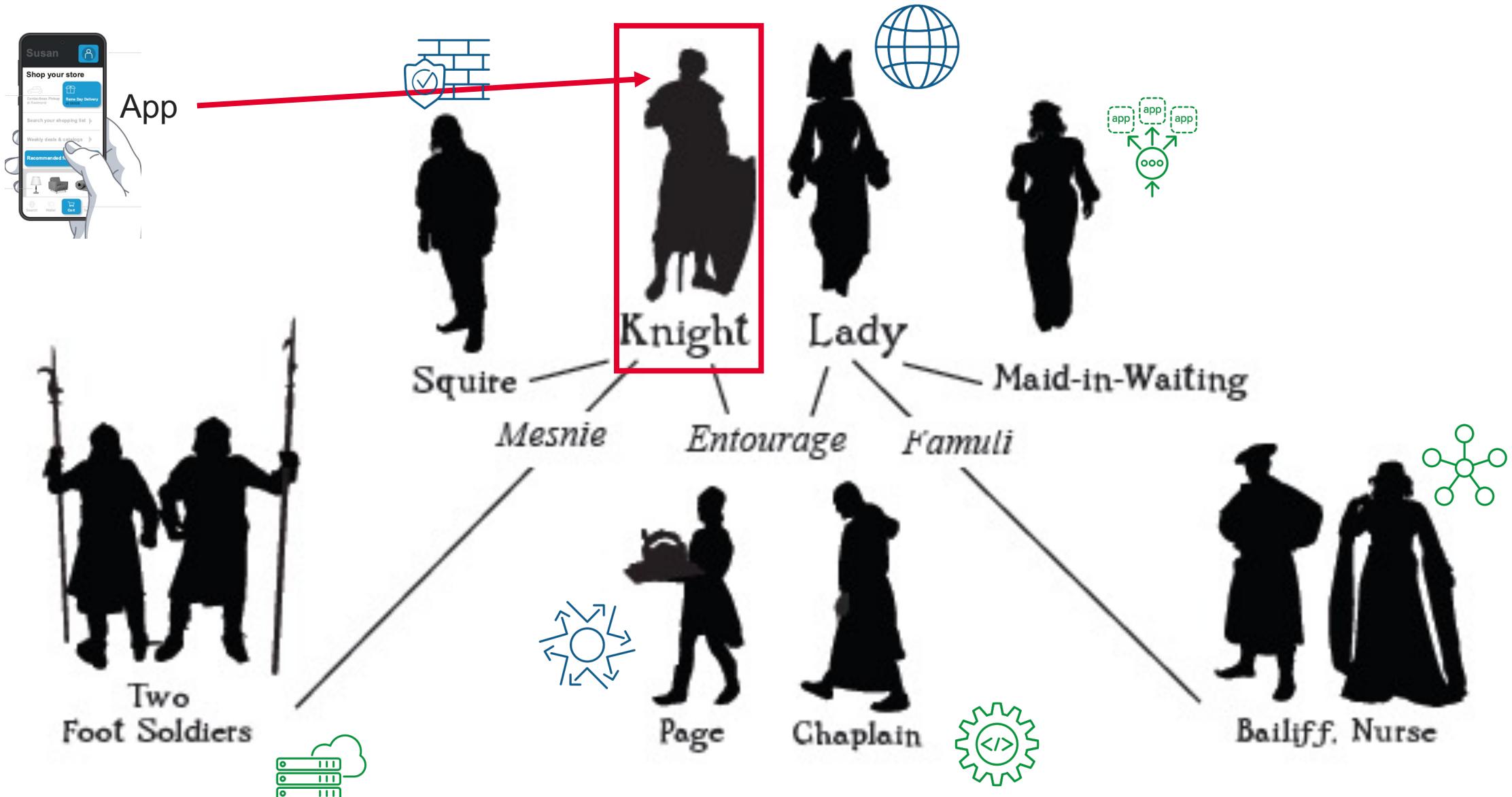


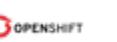
# F5 TODAY: “Comprehensive Services Surrounding Applications”

# Single App is comprised of **legacy** and **modern apps**, with **multiple app sources** spanning on-premise to edge



# Some claim Knight(s) can be supported by a group of 300 to 1100 people





## ECOSYSTEM INTEGRATIONS AND DECLARATIVE APIs

## Application Traffic Insight

### VISIBILITY & INSIGHTS

**NGINX Management Suite**

**BIG-IQ**

**F5 Distributed Cloud**

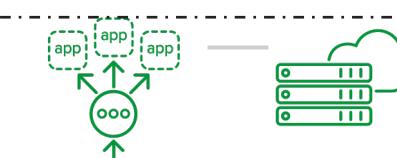
PLATFORM CONTROL PLANES

← Data Path →



**Code &  
App Clusters**

Service Mesh  
Automation  
Toolchains



East-West  
Security/LB  
Web Servers  
App Servers



K8s  
Ingress  
Controller

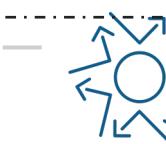
API  
Gateway &  
Management



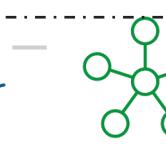
App Delivery  
Platform  
Performance &  
Contextual LB



Web App &  
API Security  
Zero-Trust  
Contextual  
Access Control



Global  
Service Load  
Balancing



L3-L7  
DDoS  
Bot  
Defence



**Customer/API**

Prevents Bot-Based  
Attacks:  
ATO, Credentials Stuffing,  
Fake Accounts, Inventory  
hoarding, Gift Cards  
Carding

## MULTI-CLOUD APPLICATION SERVICES

Security Managed  
Services

Security  
as a Service

Containers

Public/Private  
Cloud

Virtual  
machines

Commodity  
hardware

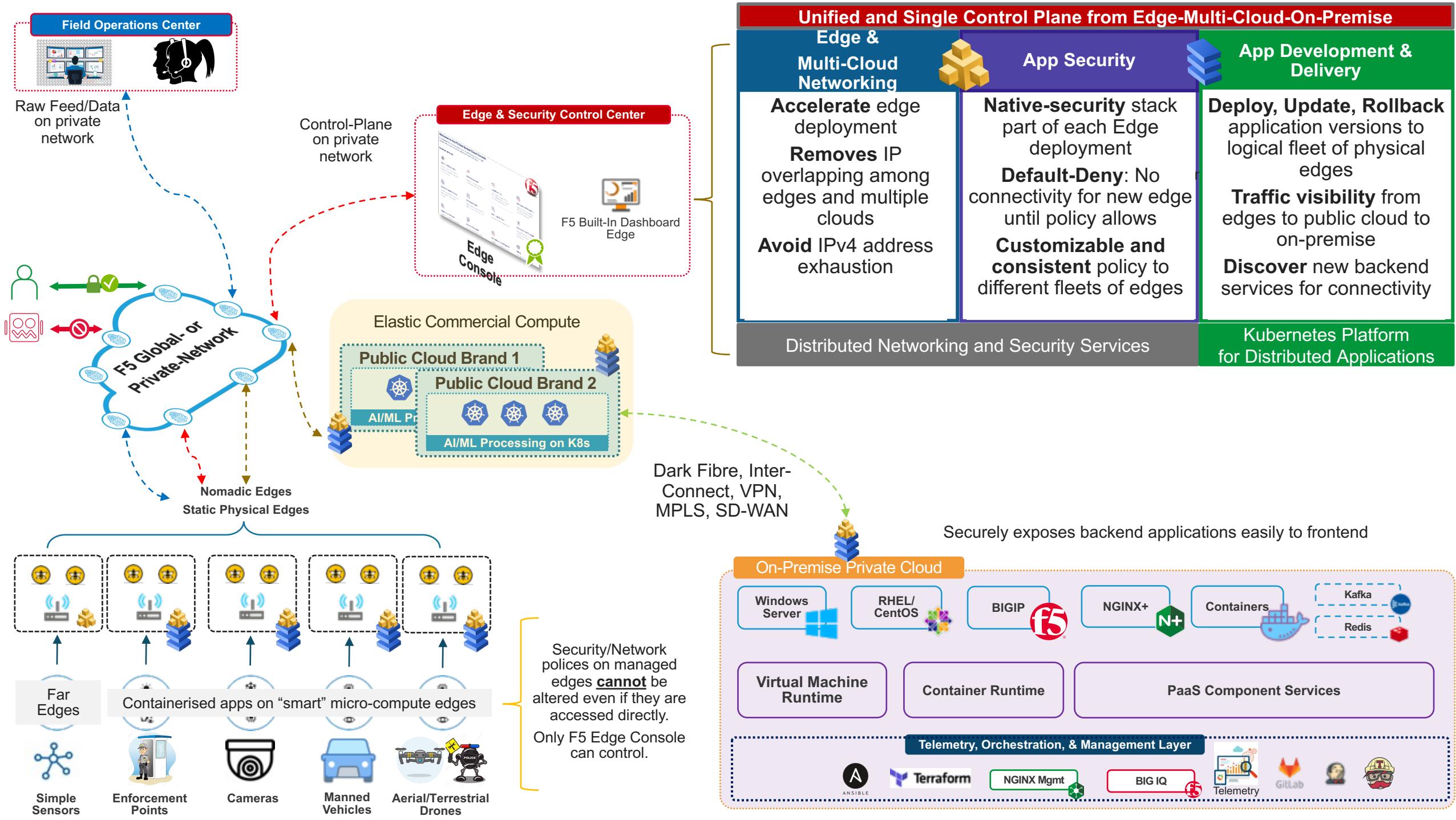
Purpose-built  
Hardware/  
Virtual appliance

Edge  
Deployment

Enforcement Points and Consumption Options from F5



# F5 TODAY: “Sample Reference Architectures”



# Use Case Categories

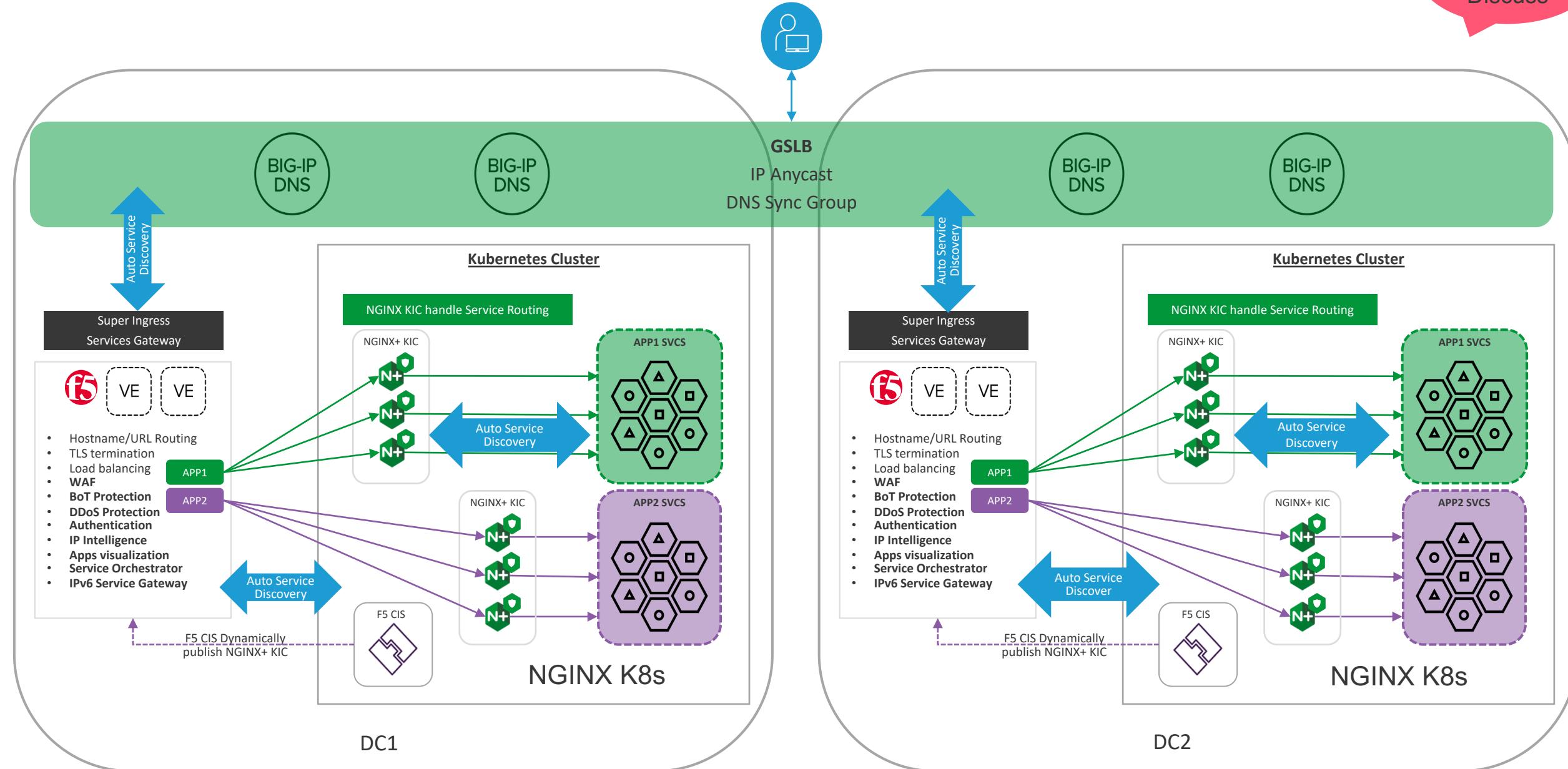
- Modern Infrastructure with BIG-IP
- Modern Applications with NGINX
- Edge Delivery Platform with F5  
Distributed Cloud

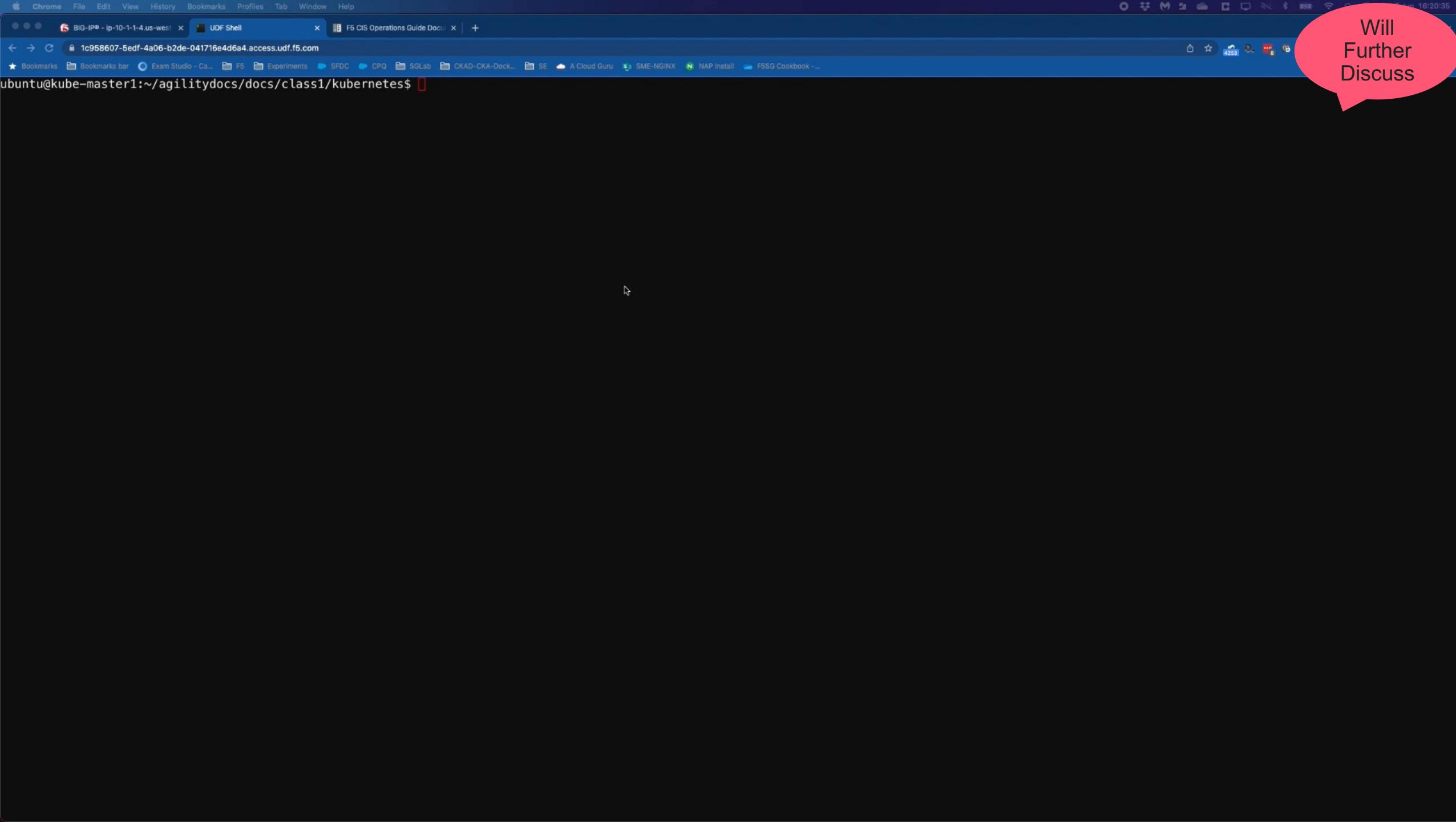
# Modern Infrastructure with BIG-IP

## Orchestrating BIG-IP's Data Center Services with Kubernetes Clusters

# Orchestrating BIG-IP's Data Center Services with Kubernetes Clusters

Will Further Discuss





Will  
Further  
Discuss



ONLINE (ACTIVE)  
Standalone

Main Help About

Statistics  
Dashboard  
Module Statistics  
Performance Reports  
  
IApps  
  
DNS  
Local Traffic  
Acceleration  
Device Management  
Shared Objects  
Security  
Network  
System

Statistics » Module Statistics : Local Traffic » Pools

Traffic Summary DNS Local Traffic Subscriber Management Network Memory System  
Display Options  
Statistics Type Pools  
Data Format Normalized  
Auto Refresh Disabled Refresh

/kubernetes/Shared/ingress\_default\_f5\_hello Search Reset Search

| Status                   | Pool   | Pool Member     | Partition / Path  | Bits |       | Packets |       | Connections |           |         | Requests |         | Request Queue |           | Message Routing Framework |           |            |            |             |  |  |  |
|--------------------------|--|-----------------|-------------------|------|-------|---------|-------|-------------|-----------|---------|----------|---------|---------------|-----------|---------------------------|-----------|------------|------------|-------------|--|--|--|
|                          |  |                 |                   | # In | # Out | # In    | # Out | # Current   | # Maximum | # Total | # Total  | # Depth | # Maximum Age | # Msg. In | # Msg. Out                | # Req. In | # Req. Out | # Resp. In | # Resp. Out |  |  |  |
| <input type="checkbox"/> | <input checked="" type="checkbox"/> ingress_default_f5_hello_world_web | 10.244.1.3:8080 | kubernetes/Shared | 0    | 0     | 0       | 0     | 0           | 0         | 0       | 0        | 0       | 0             | 0         | 0                         | 0         | 0          | 0          | 0           |  |  |  |
| <input type="checkbox"/> | <input checked="" type="checkbox"/> ingress_default_f5_hello_world_web | 10.244.1.4:8080 | kubernetes/Shared | 0    | 0     | 0       | 0     | 0           | 0         | 0       | 0        | 0       | 0             | 0         | 0                         | 0         | 0          | 0          | 0           |  |  |  |
| <input type="checkbox"/> | <input checked="" type="checkbox"/> ingress_default_f5_hello_world_web | 10.244.2.2:8080 | kubernetes/Shared | 0    | 0     | 0       | 0     | 0           | 0         | 0       | 0        | 0       | 0             | 0         | 0                         | 0         | 0          | 0          | 0           |  |  |  |
| <input type="checkbox"/> | <input checked="" type="checkbox"/> ingress_default_f5_hello_world_web | 10.244.2.3:8080 | kubernetes/Shared | 0    | 0     | 0       | 0     | 0           | 0         | 0       | 0        | 0       | 0             | 0         | 0                         | 0         | 0          | 0          | 0           |  |  |  |
| <input type="checkbox"/> | <input checked="" type="checkbox"/> ingress_default_f5_hello_world_web | 10.244.2.4:8080 | kubernetes/Shared | 0    | 0     | 0       | 0     | 0           | 0         | 0       | 0        | 0       | 0             | 0         | 0                         | 0         | 0          | 0          | 0           |  |  |  |

Reset

Visibility to each  
Pod is possible  
too

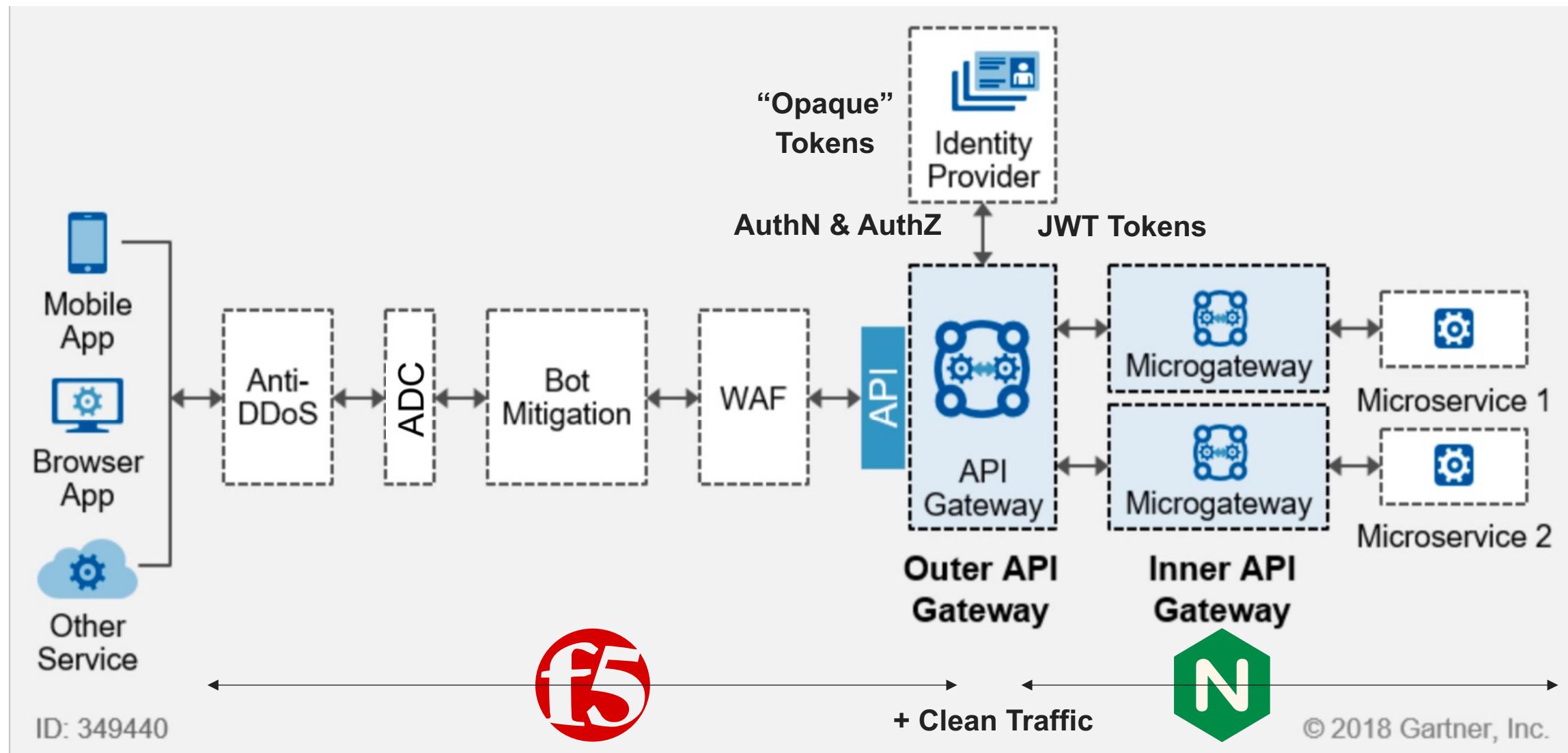
## Key Message:

“Current & Future Investments of **F5 BIG-IP’s Deploy, Deliver, Defend Solutions (ADC, DNS, Zero-Trust App Access, Web/API Security)** can be directly applied to both your microservices & traditional architectures”

# Modern Applications with NGINX

**Better Together with F5 BIG-IP**

# Modern Container Application Architecture From Gartner



Will  
Further  
Discuss

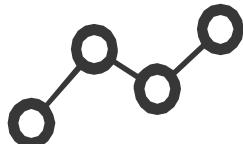


## N/S control

Control and secure the traffic to the container platform

## Ingress control

Expose, scale, and secure container-based apps to the world



## E/W control Microgateway

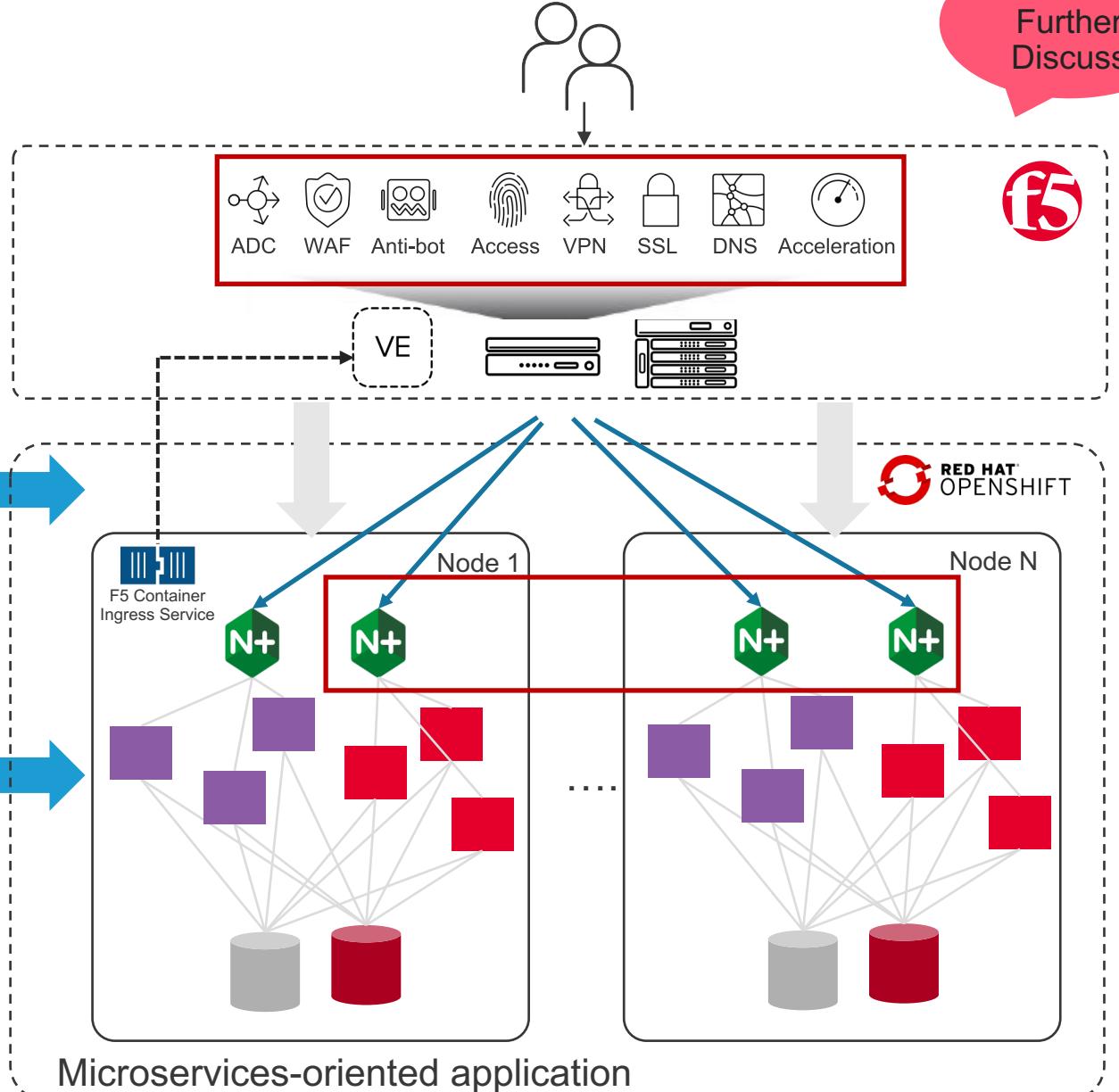
## NetOps/SecOps

Automating app services for N-S inbound traffic with F5 CIS<sup>#</sup>

## F5 Service Discovery

## DevOps/DevSecOps

Managing and securing microservices (E-W)



#F5 Container Ingress Service makes L4-L7 services available to users deploying microservices-based applications in a containerized infrastructure.

CIS allows you to expose a Kubernetes Service outside the cluster as a virtual server on a BIG-IP device entirely through the Kubernetes API.



# NGINX Technology Stack

*A suite of technologies to develop and deliver digital experiences that span from legacy, monolithic apps to modern, microservices apps.*

*NGINX runs on an OS, not a virtual appliance*

*NGINX performance is virtually unlimited. NGINX works at 1vCPU to 48vCPU and beyond*

*NGINX can run virtually anywhere*

## NGINX Management Suite



Self-Service  
(ADC/APIM)



Analytics



API-Mgmt



Discover



Config Management



API First

## NGINX Data Plane Solutions



Web Server/  
Content Cache/  
Video Stream



App Protect  
(WAF & DOS)



Load Balancer/  
Reverse Proxy



API Gateway



K8s Ingress  
Controller



Service Mesh

NGINX Plus



NGINX Unit (App Server)



Cloud



Virtual Machine



Container



Bare Metal

Multi-Cloud versatility

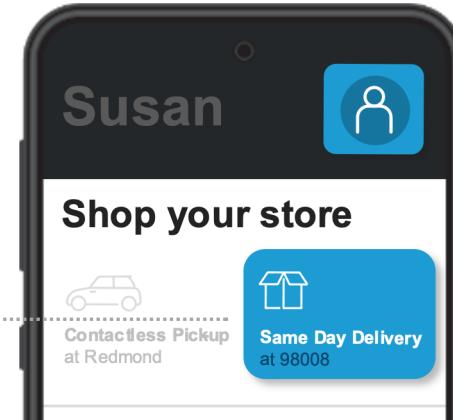
## Key Message:

“**NGINX** allows **DevOps/DevSecOps** to provide **Deploy, Deliver, Defend** services to scale according to applications’ needs and speed”

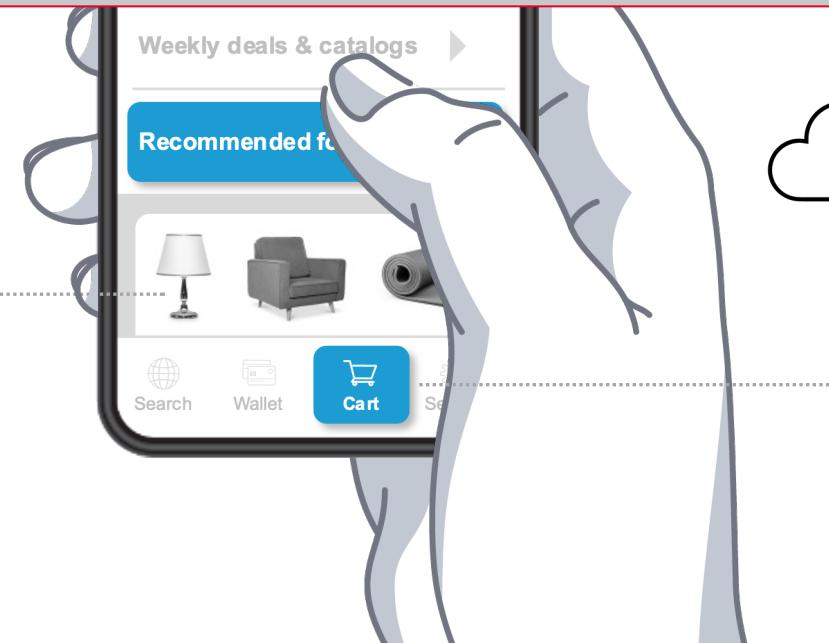
# F5 Distributed Cloud (Platform for your Edge)

## Application Delivery Network

**Single App is comprised of **legacy** and **modern apps**,  
with **multiple app sources** spanning on-premise to edge**



**REMEMBER THIS EXAMPLE?**



# F5 Distributed Cloud Platform (Abbreviated F5 XC)

What complexities we solve

'X' = Any App Services

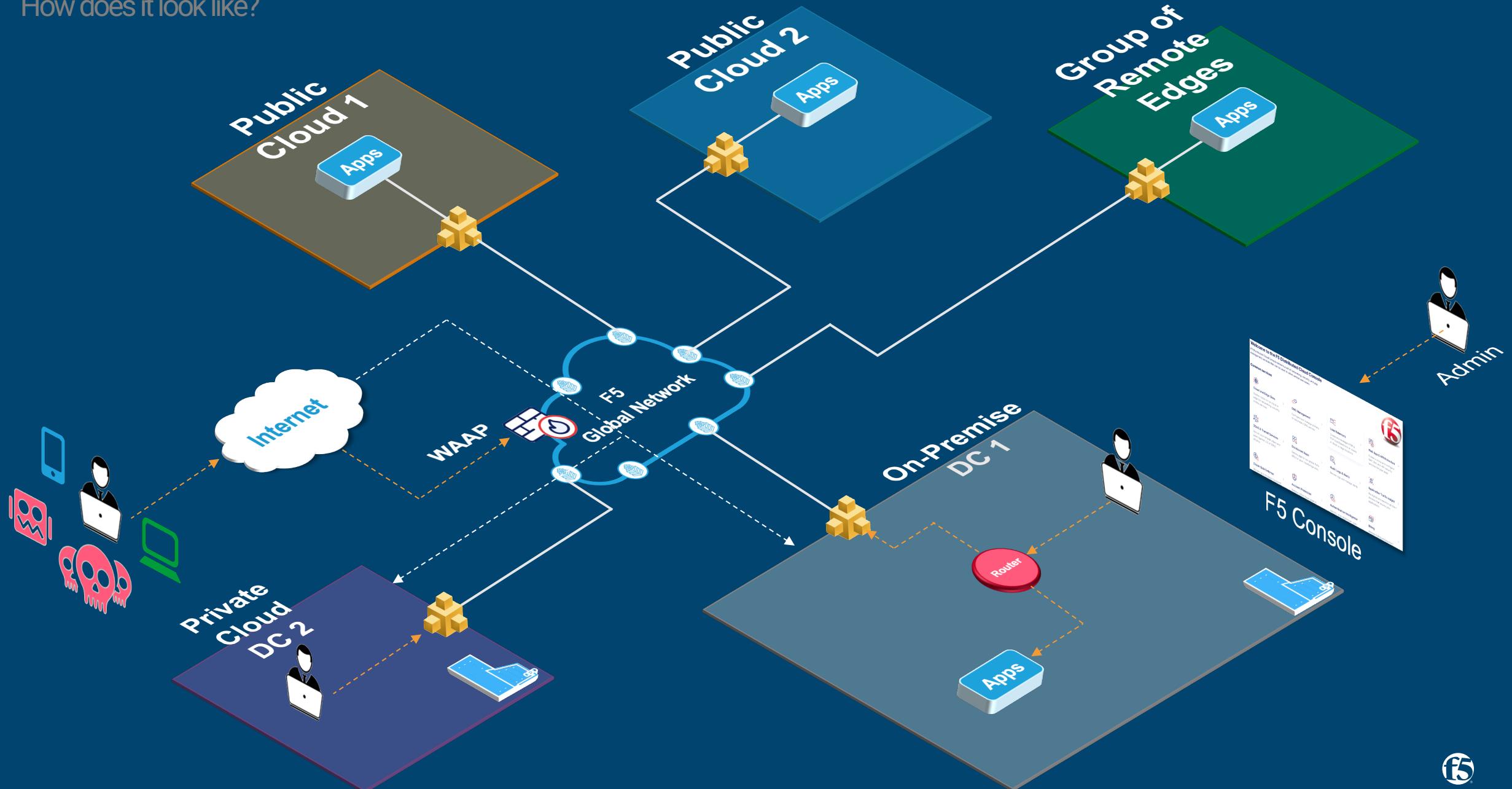
1. Simplify your network/application connectivity/stitching and reachability (MCN)  XC Mesh for stitching

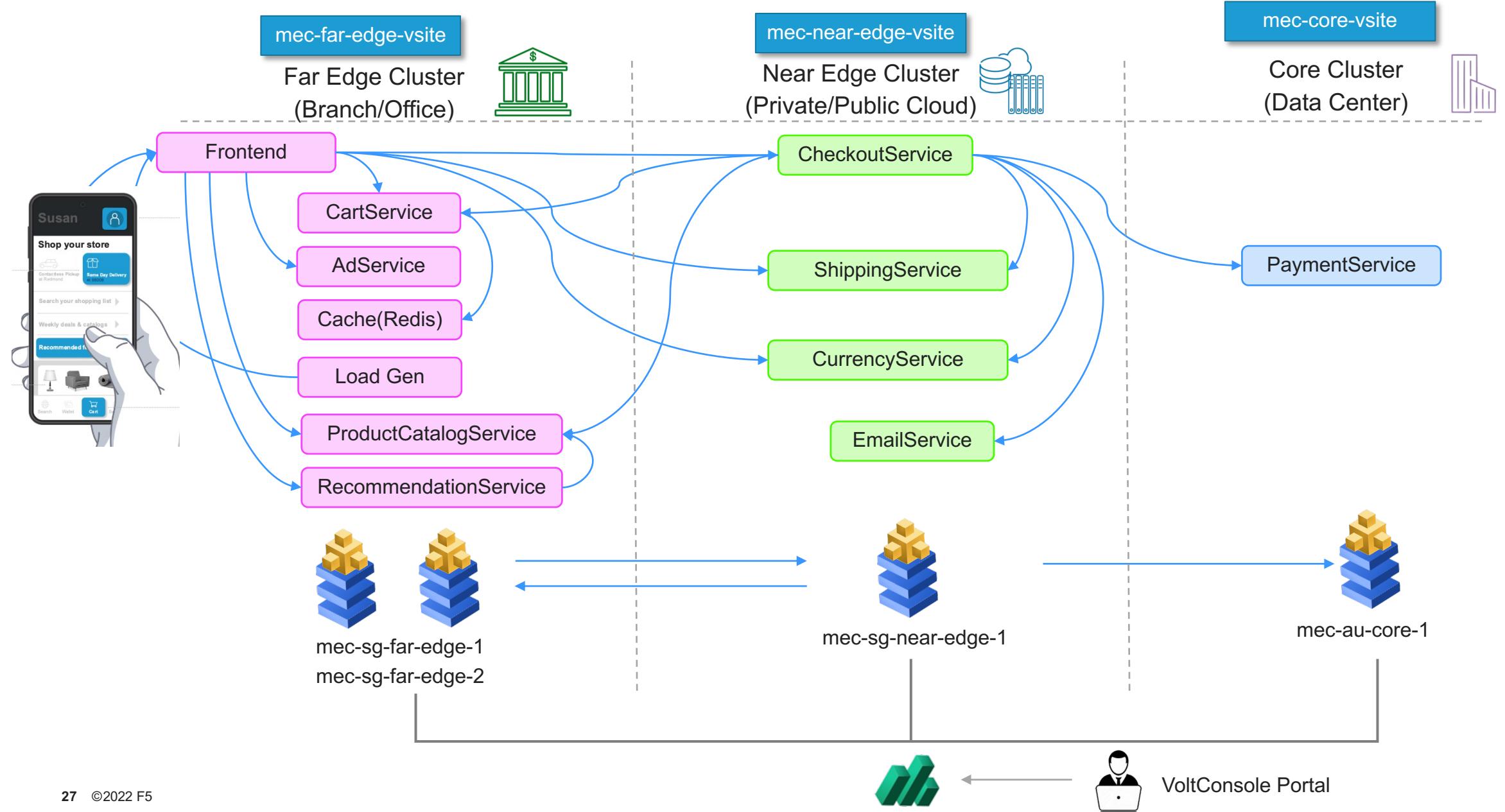
2. Simplify your modern application Deployment, Delivery and Lifecycle  XC K8s AppStack

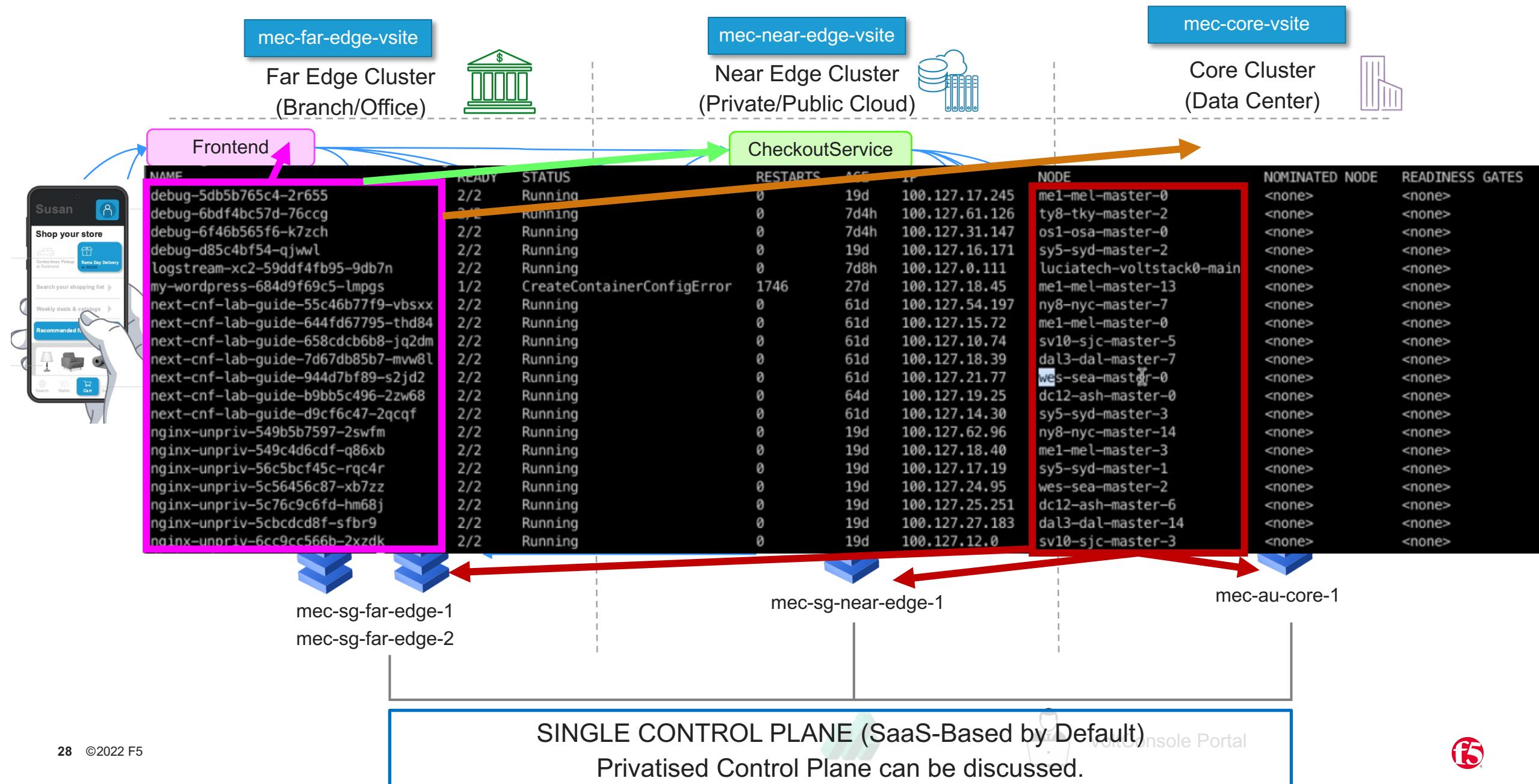
3. Simplify best of breed security consumptions (WAAP – WAF, API, Bot & DDoS)

# F5 Distributed Cloud Platform

How does it look like?



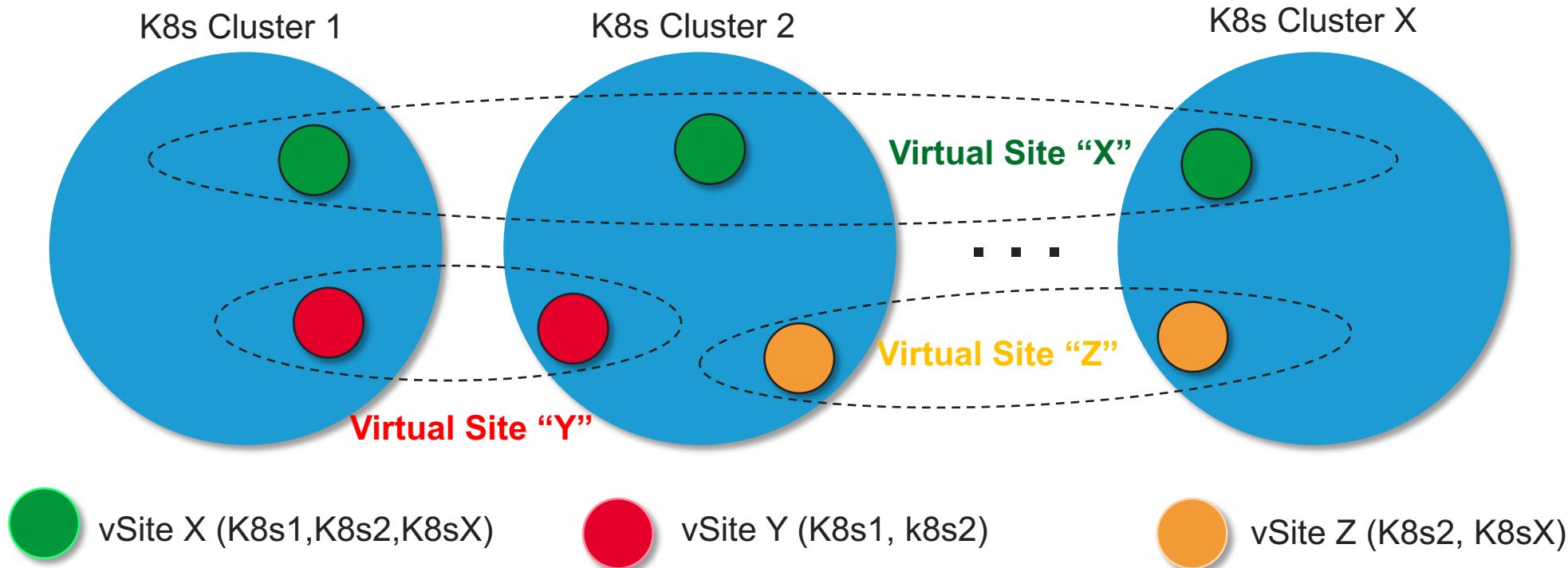




# Fleet Management with ‘Virtual Sites’

*F5 Distributed Cloud enables you to update and rollback application versions on numerous edges with literally “1-click”*

A Kubernetes compatible API for **centralized orchestration (intent-based orchestration)** of application across **a fleet of site**. It allow you to group your sites (tens to hundreds of thousands of site) into a **logical group** and manage them as a single entity.



*E.g “I want this payment apps on Virtual Site called “Z” which consists of K8s Cluster 2 and K8s Cluster X”*

“Key Message: Complexity is the enemy of security. Simply Deploy, Deliver, Defend, **DISTRIBUTE** your apps onto Edges with F5 Distributed Cloud Edge Platform”

