TRY-BUY-ADAPT

26 Sep 2019

Program Executive Office for Simulation, Training, and Instrumentation (PEO STRI)

Amit Kapadia
Rick Osborne

Problem Statement

- In the US Department of Defense (DoD), following a waterfall process can produce high quality products but often the products are **no longer relevant** by the time they reach their users.
- The commercial sector is developing key technologies and applications that have the potential for **cost-effective adaptation** for defense use cases such as mission rehearsal and training.
- The exploitation of **commercial technologies** and use of **open architectures** can improve efficiencies, enabling the military modeling and simulation community to effectively respond to dynamic environments with rapidly changing threats.

Purpose

- Discuss how PEO STRI utilized **commercial technologies** and **open architecture** to release an initial cyber training prototype as well as a subsequent release within one year of receiving funding.
  - PEO STRI leveraged a **Try-Buy-Adapt** Acquisition Strategy and **Microservice Architecture**
- Provide challenges and lessons learned regarding **Try-Buy-Adapt** Acquisition Strategy and **Microservice Architecture** adoption
- **Vision:** Continued microservice architecture adoption will increase release frequency to quarterly and perhaps monthly.

PCTE Vision

FOR Cyber Mission Forces (CMF)

WHO Need a High Fidelity, Persistent Training Environment to Conduct the Spectrum of Cyberspace Operations

THE PCTE provides a holistic, on-demand platform

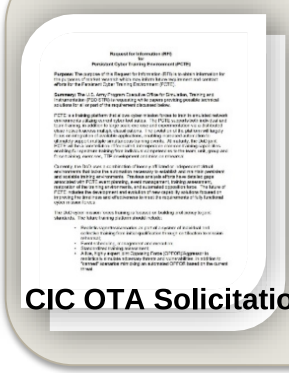
THAT Enables end-to-end planning, preparation, execution & assessment across the training continuum

UN-LIKE Stove-piped, manpower & time-intensive training environments

Provides a standardized platform with ecosystem of capabilities to shape, execute & re-use multiple training environments/scenarios

TRY-BUY-ADAPT

Try: Cyber Innovation Challenge (CIC) Assessment



CIC OTA Solicitation

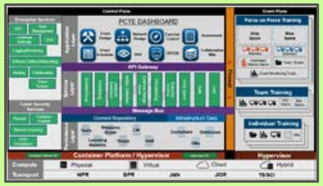
- Identify large capability gaps
- Write CIC solicitation
- Down-select Industry COTS best-of-breed prototypes
- Conduct demo of candidate solutions for CMF evaluation
- Assess PCTE integration readiness of candidate solutions

Buy: License Products and Procure Services

Utilize Other Transaction Authority (OTA) Vehicles

- License COTS products for adaptation
- Procure services to integrate COTS products, add new features, and conduct pilots

Adapt: Integrate Products using Open Architecture



PCTE Reference Architecture

- Leverage open architectures and standards to facilitate integration
- Integrate COTS products and add new features
- Leverage COTS enterprise middleware to integrate products
- Deliver though monthly agile scrum process

Try-Buy-Adapt Lessons Learned

Challenge	Lesson Learned
How do I identify the right scope for CICs to ensure that capabilities can be assessed?	• **Problem:** During the CIC-1, vendors could demonstrate all cyber training capabilities which made it difficult to create evaluation criteria and subsequently assess the products. • **Solution:** In CIC-2 and CIC-3, we identified specific cyber training capabilities which made it easier to develop evaluation criteria.
How do I train the CIC evaluators to ensure the data collected is meaningful?	• **Problem:** During CIC-1, the evaluators did not receive training so the first few demonstrations were essentially practice. • **Solution:** In CIC-2 and CIC-3, training was provided on the process, tools, and evaluation criteria so the evaluators could perform meaningful assessments.
How do I ensure the solutions chosen by the evaluators can be integrated into PCTE?	• **Problem:** One solution chosen by the evaluators in CIC-1 could not be easily integrated into PCTE. • **Solution:** In CIC-3, a software architect was included in the selection process to ensure the capabilities selected by the evaluators can be integrated into PCTE.

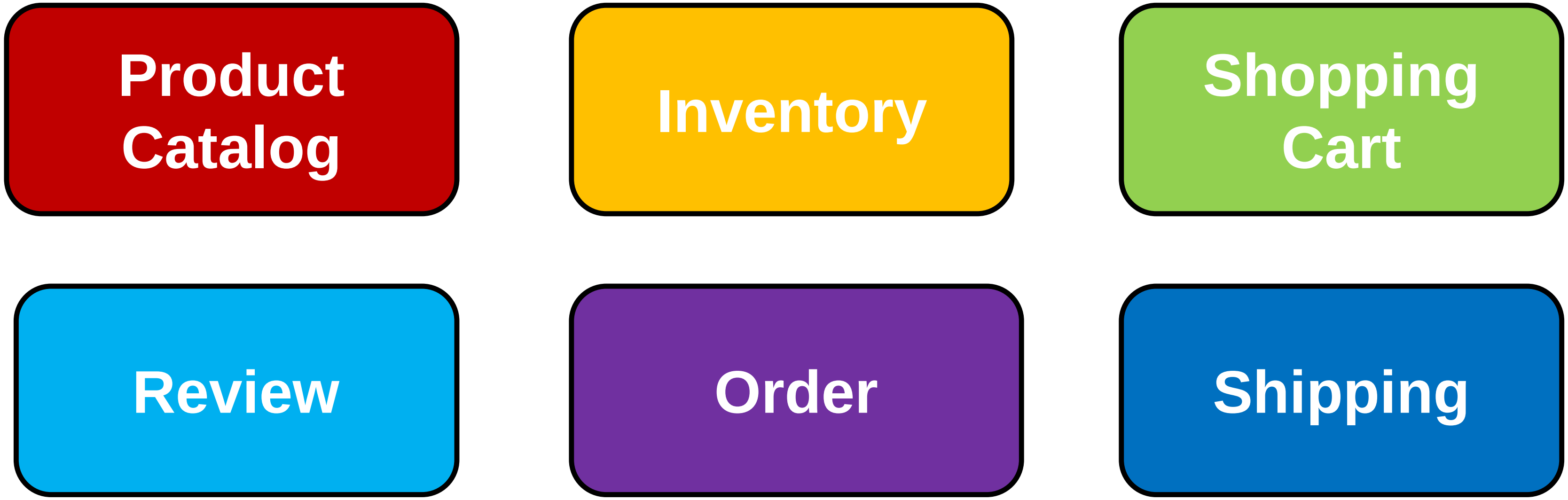
Microservice Architecture

The microservices architecture is a design approach that enables **rapid releasing of software** by developing an application from a **collection of loosely coupled services.** Each service provides a **single business capability**.

Microservice Pioneers and Adopters

Notional Microservice Architecture

For example, a notional online store may consist of the following services. Each service provides **a single business capability**:



Product Catalog

Inventory

Shopping Cart

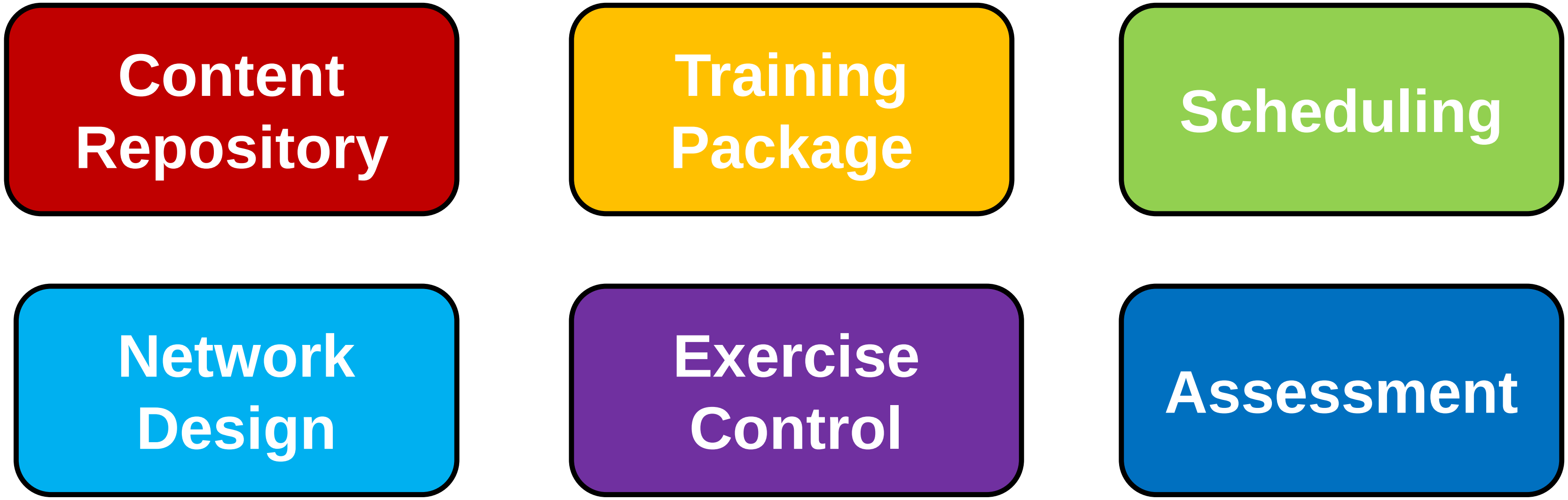
Review

Order

Shipping

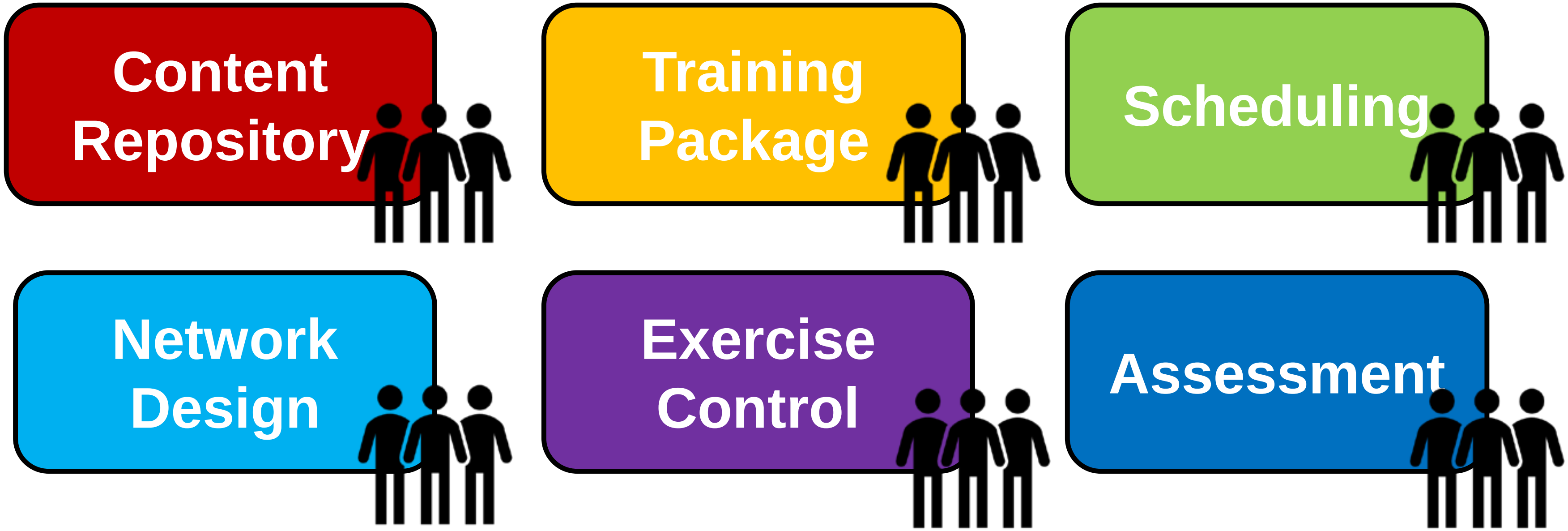
PCTE Microservice Architecture

PCTE consists of but is not limited to the following services. Each service provides **a single business capability**:



Rapid Software Releases

Microservice Architecture enables **rapid software releases** by allowing agile teams to **independently** develop, test, and deploy each service.



Content Repository

Training Package

Scheduling

Network Design

Exercise Control

Assessment

How do agile teams independently develop, test, and deploy each service?

Independent Development

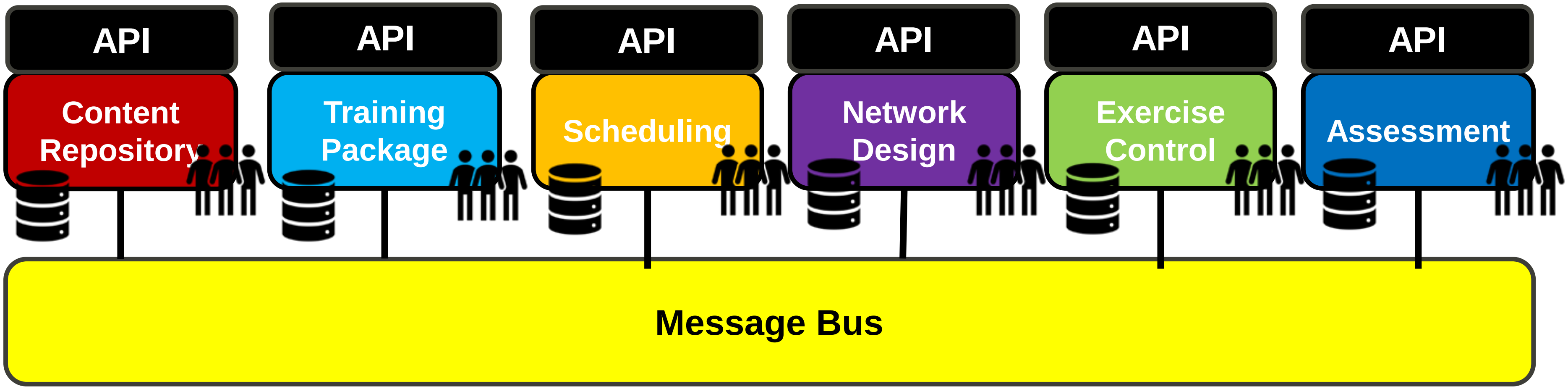
How do agile teams **independently develop**, test, and deploy each service?

Limit dependencies between teams by following these best practices:

- No shared software libraries
- No shared databases
- Loose coupling via well-defined APIs and inter-service communication through a message bus using a publish-subscribe model.

PCTE Team has no visibility of COTS internals

Ongoing PCTE effort

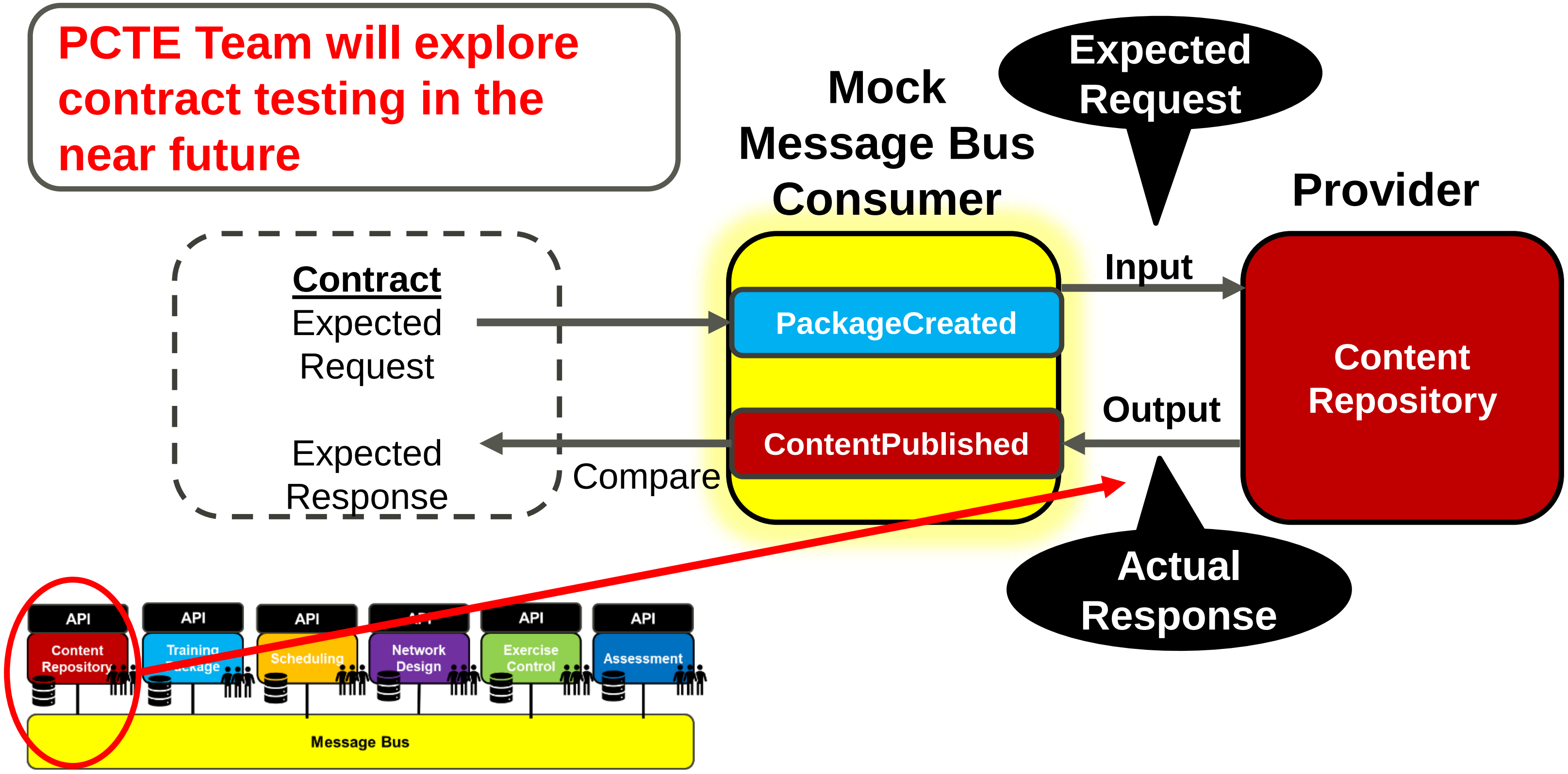


Independent Testing

How do agile teams **independently** develop, **test**, and deploy each service?

Implement **Contract testing** which is a technique to ensure that microservices will work together **prior to conducting an expensive integration test.**

PCTE Team will explore contract testing in the near future



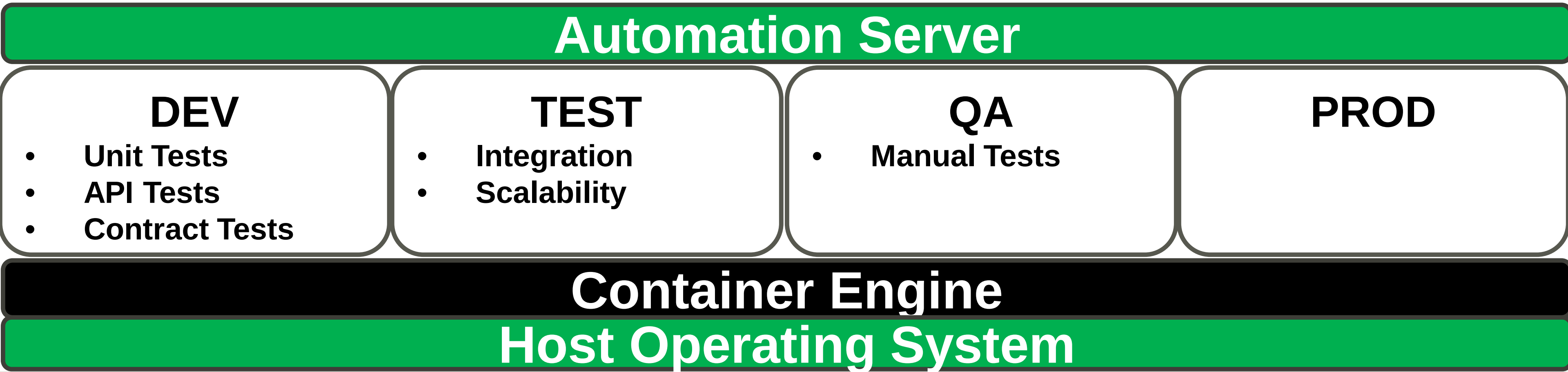
Independent Deployment

How do agile teams **independently** develop , test , and **deploy** each service?

Completed

- Adopt containers which **deploy in seconds** and support rapid testing.
- Construct **Continuous Integration / Continuous Delivery (CI/CD) pipelines** with automation servers to automate the testing and promotion of microservices from DEV->PROD
- **Independently promote** microservices between environments

Future PCTE Effort



Automation Server

DEV
- Unit Tests
- API Tests
- Contract Tests

TEST
- Integration
- Scalability

QA
- Manual Tests

PROD

Container Engine

Host Operating System

Summary

- The military modeling and simulation community cannot afford to continue the waterfall government-based development that **takes years to field** a capability to the warfighter.
- Acquiring **commercial technologies** and leveraging **open architectures** allows the acquisition community to field capabilities in significantly less time.
- Continued use of CICs (**Try-Buy-Adapt**), coupled with **microservice architecture adoption** throughout PCTE's life cycle, will enable the Army to **rapidly deliver new features** to the CMF, and evolve the PCTE platform based on changes to technology, threat, and tactics, techniques, and procedures (TTPs).

Question & Answers