“The increased speeds and lower latency of 5G networks are beginning to impact nearly every facet of life for consumers and enterprises. Fortunately, security has been the built into 5G right from its inception and has been required throughout its development, planning and deployment.”

Chris Pearson, President, 5G Americas

"5G will allow operators to evolve toward new business models. For 5G to achieve its potential, organizations must embrace multi-layered security that goes far beyond 3GPP specifications by using a pragmatic, multi-layered approach. End-to-End Security should cater to RAN, SDN, MEC, and hybrid, multi-cloud deployments based on a cloud native architecture, secure CI/CD, and zero trust security for 5G."

Pramod Nair
Technical Solutions Architect – Security
Cisco

"5G continues to integrate with other key technology enablers. In the cloud's multi-stakeholder environment, cloud-native function software vendors, platform vendors, mobile network operators, hyperscale cloud providers, and system integrators must collaborate to clearly define requirements, roles and responsibilities for implementing security architecture and controls."

Scott Poretsky

Director of Security, North America, Network Product Solutions at Ericsson

5G Deployment Models

On-premises

5G CNFs deployed only on Private Cloud, programmable routers, low footprint servers

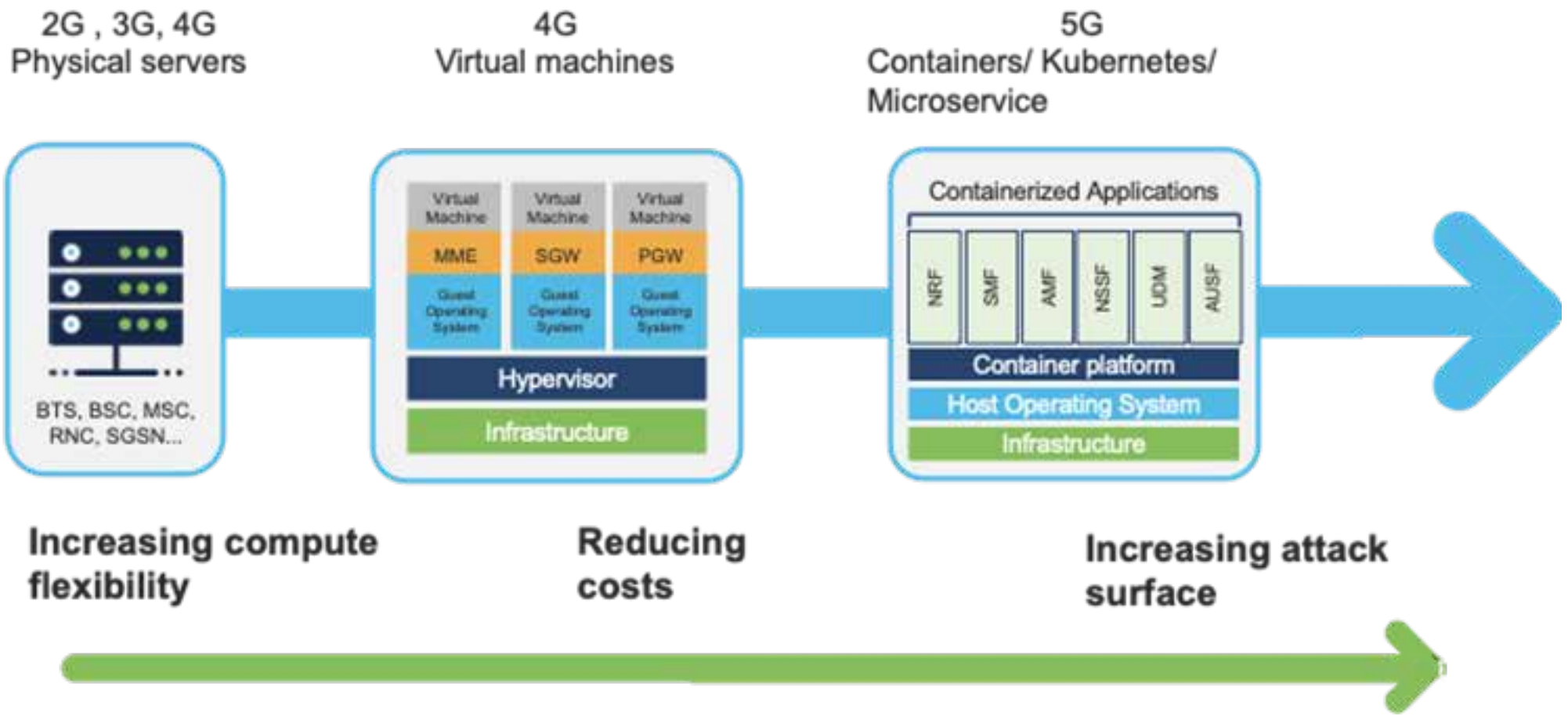
Multi-stack Public Cloud

5G CNFs deployed only on Public Cloud providers (GCP, AWS, Azure etc) & Public Cloud provided by 5G Equipment Vendors

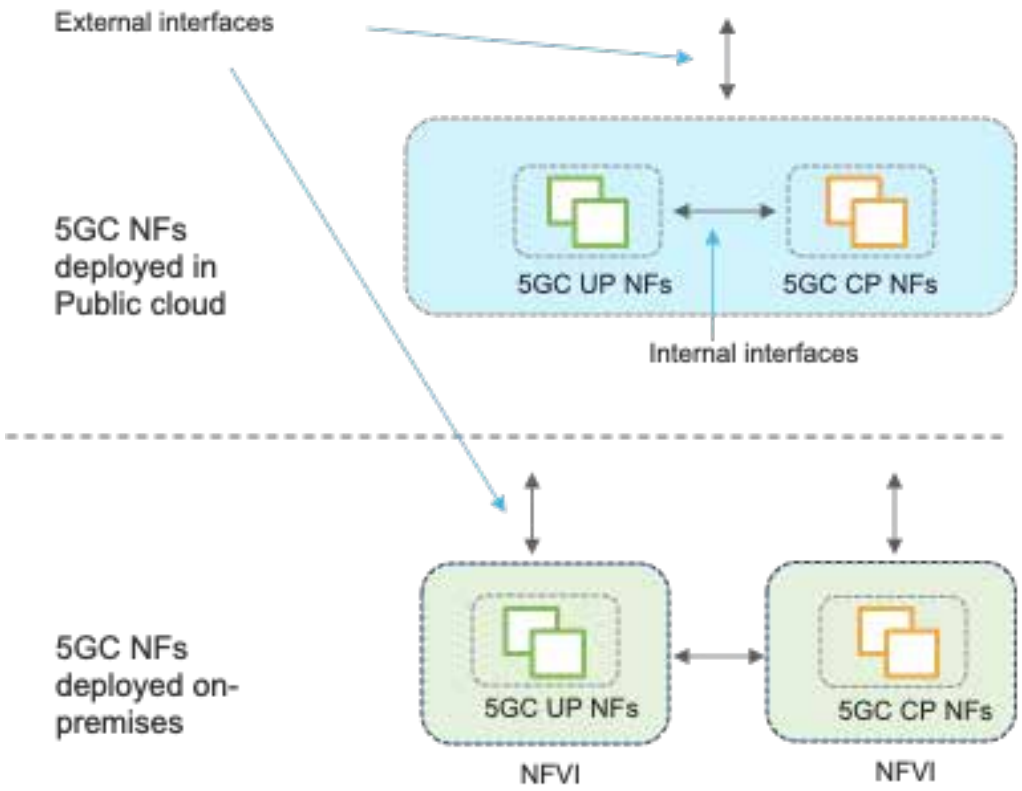
Hybrid Cloud

5G CNFs deployed on mix of on-premises & Multi Stack Public Cloud

Evolution to 5G CNFs

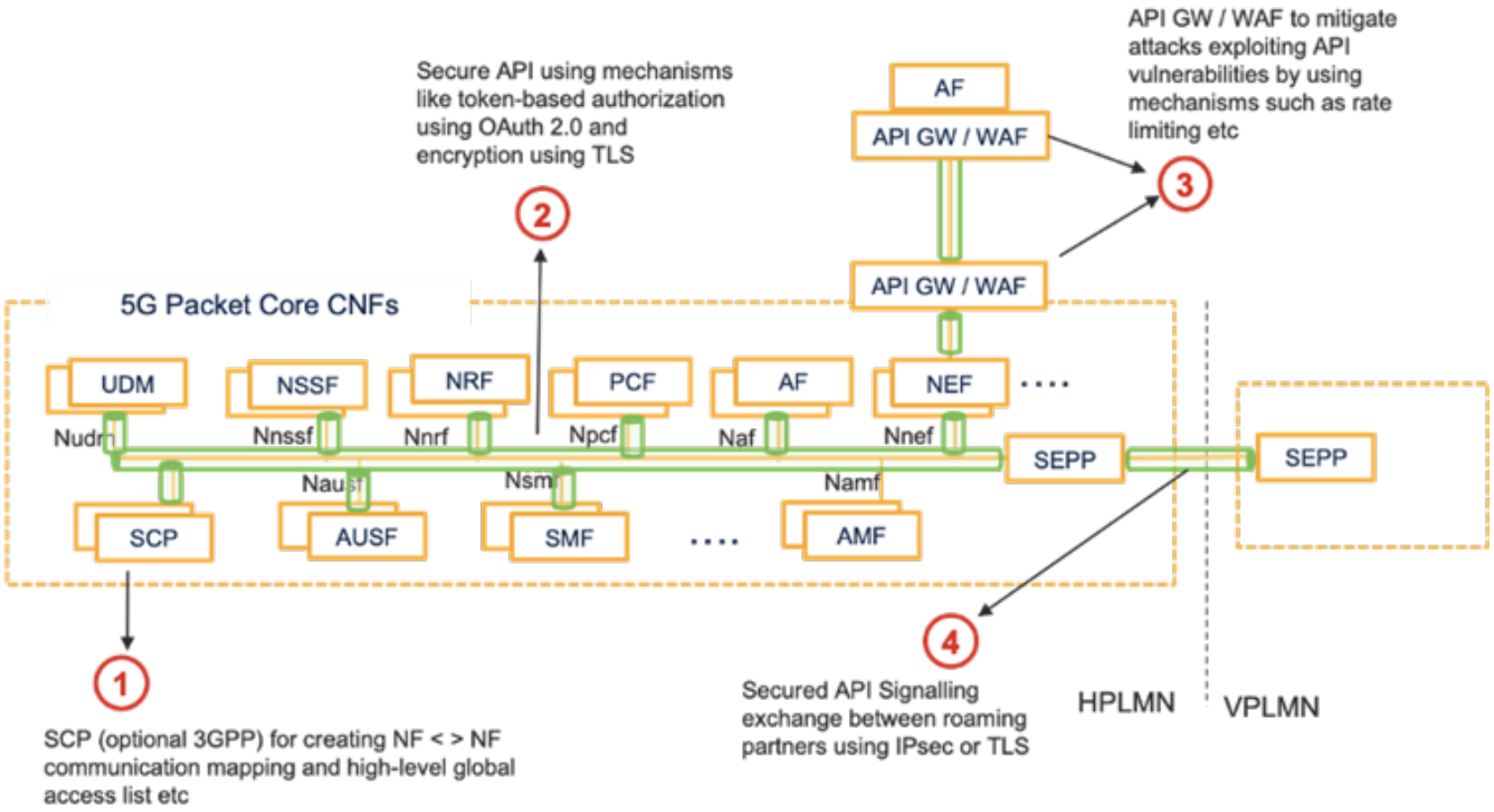


Security risks within the 5G CNF deployments

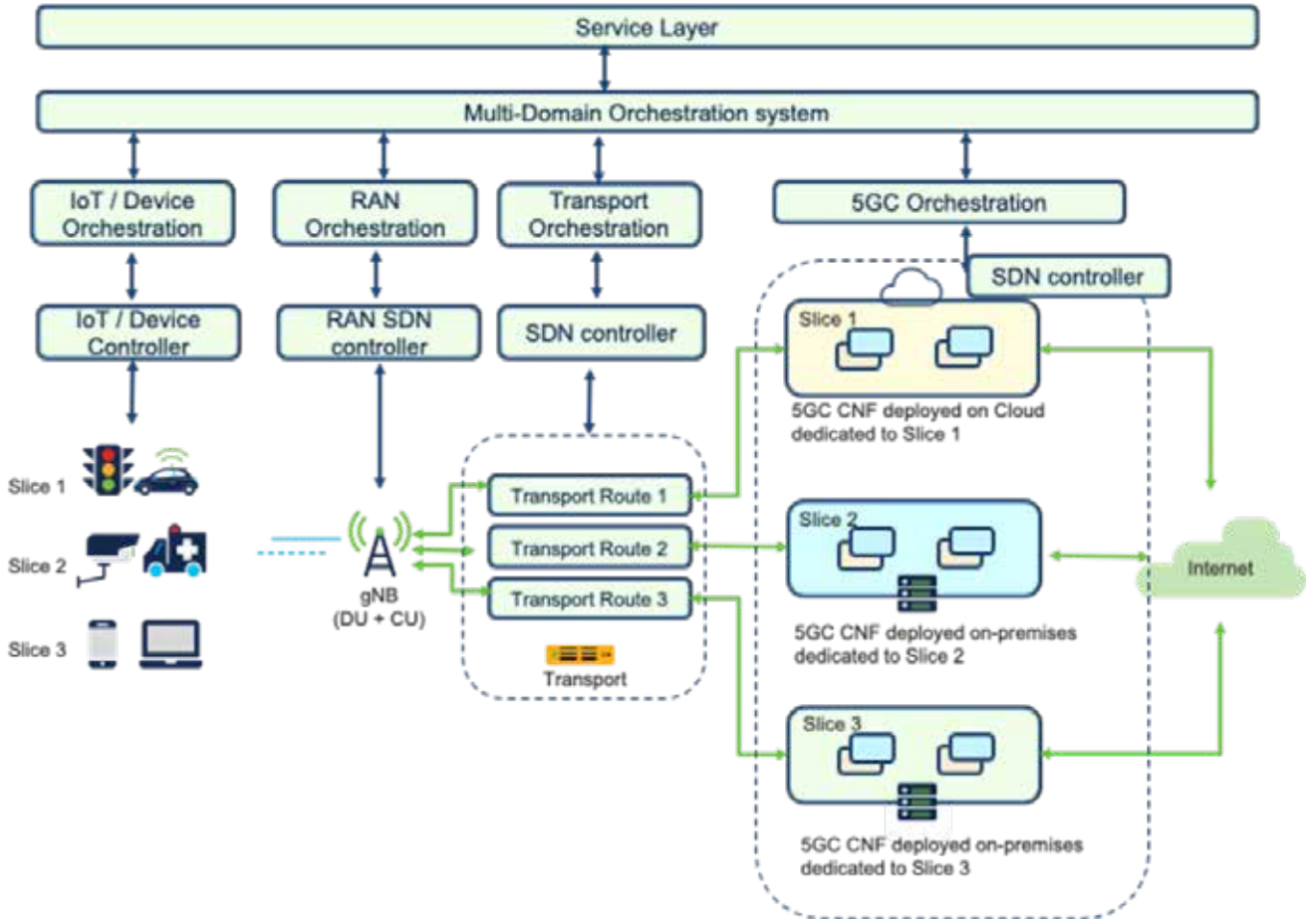


- 5GC Container vulnerabilities
  - Insecure container build
  - Container runtime vulnerabilities
  - Insecure container host
  - Malicious Container network traffic
  - Malicious 5GC CNFs
  - Insecure container management and orchestration
  - Improper access control
  - Insufficient isolation
- Internal Interfaces
  - API vulnerabilities within SBI communication
  - Insecure network leading to eavesdropping
- External Interfaces
  - Improper Isolation & Segmentation with enterprise network
  - API vulnerabilities
  - Vulnerabilities relates to non-3GPP MEC NFs
  - Insecure roaming interface & misconfigurations
- Hardware
  - NFVi Hardware & Software vulnerabilities
  - Improper Access control

API Security controls in the 5G Service Based Architecture (SBA)



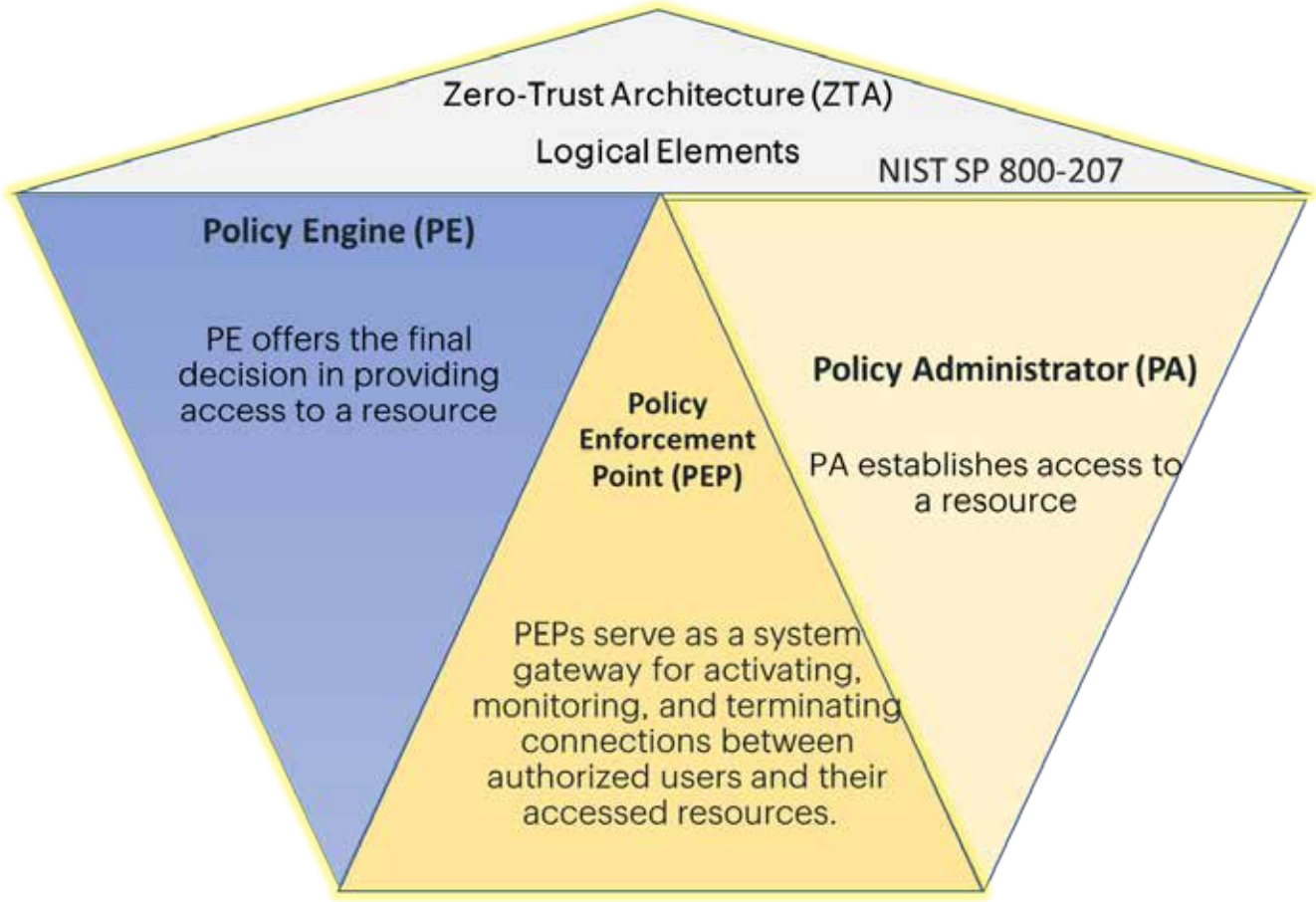
Service, Orchestration, and Automation layers in a 5G network



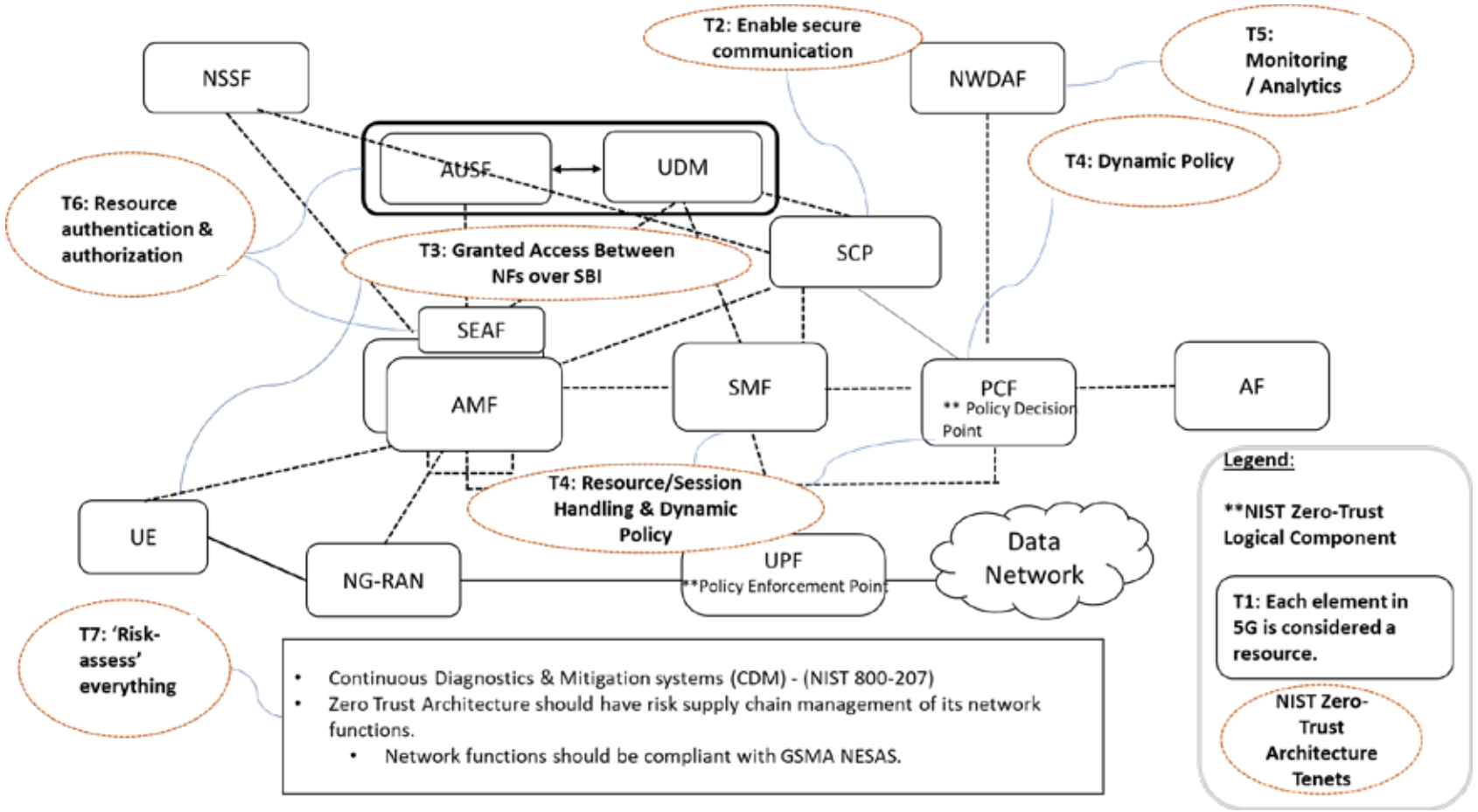
Orchestration	Securing Orchestration management & interfaces, Securing Policy Enforcement and enhancing visibility within Orchestration and between Orchestration and network components
User	Segmentation, User Access based on Zero Trust principles, DNS protection
Network	Segmentation, Policy enforcement, Securing Network interfaces, Securing Cloud integrations and workloads, Securing Peering & Roaming interface
Applications	Securing 3rd Party application interfaces, DDoS protection, Application security, DevSecOps practises, Segmentation, Cloud application policy sync and enforcement, securing API
VNFs and CNFs	Securing VNF / CNF, securing Software Lifecycle, Isolation between VNF's / CNFs, detecting malicious virtual functions and vulnerabilities
Infrastructure	Hardening of NFVI, perimeter security, DDoS protection, securing – E-W traffic

Layered 5G security controls

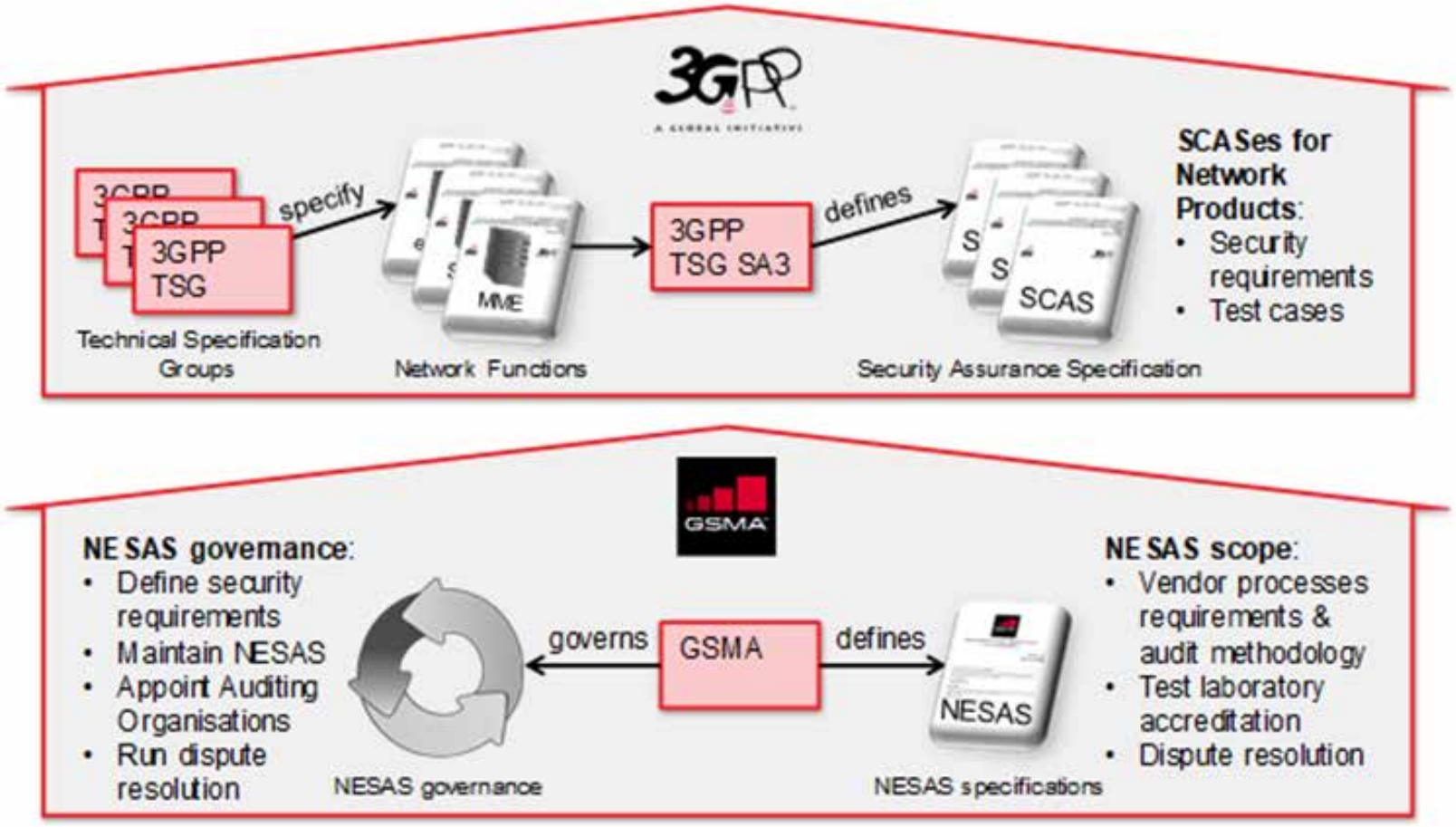
Zero-Trust Architecture Logical Elements



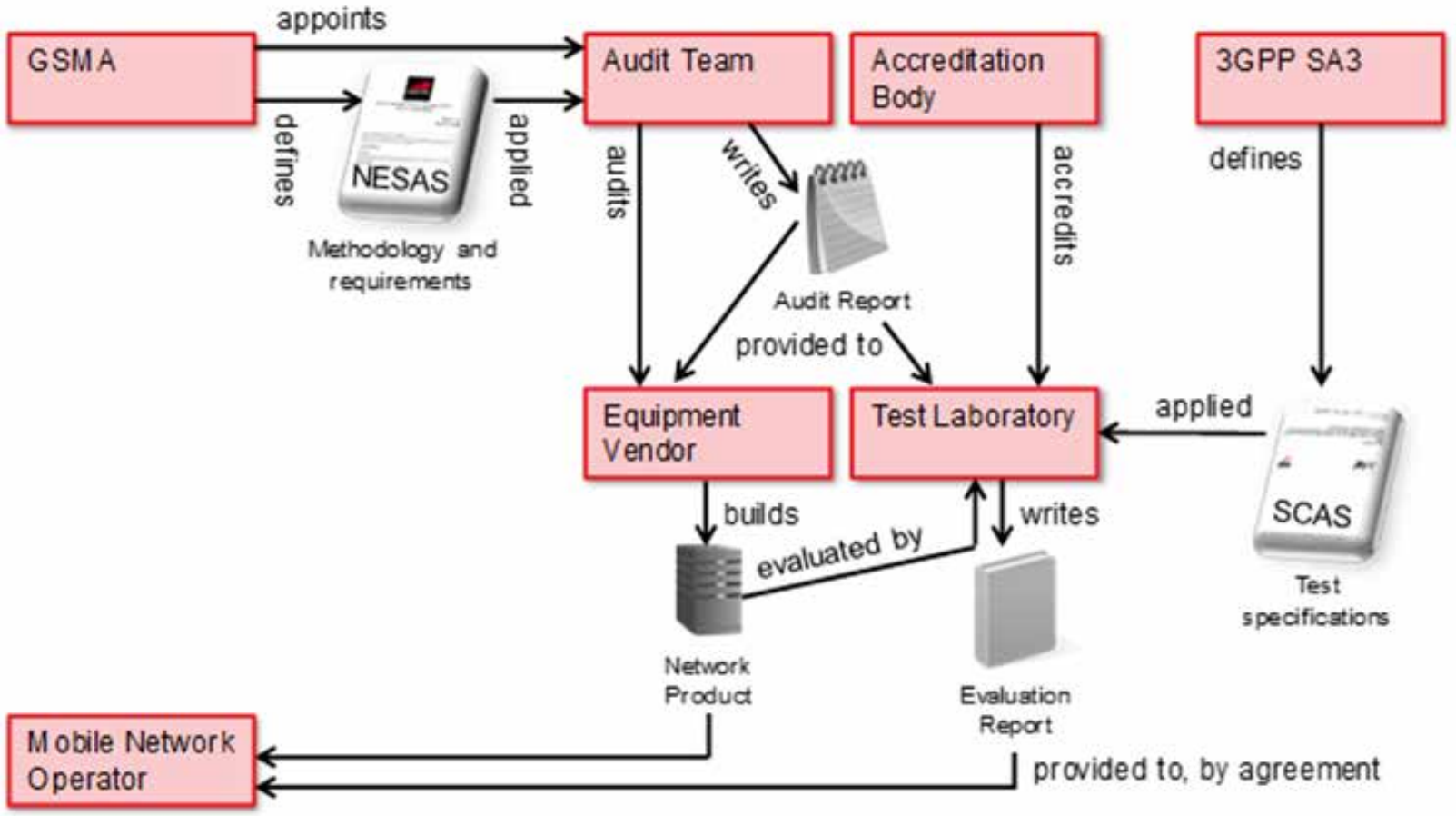
Overlay of NIST ZTA with 3GPP 5G Architecture



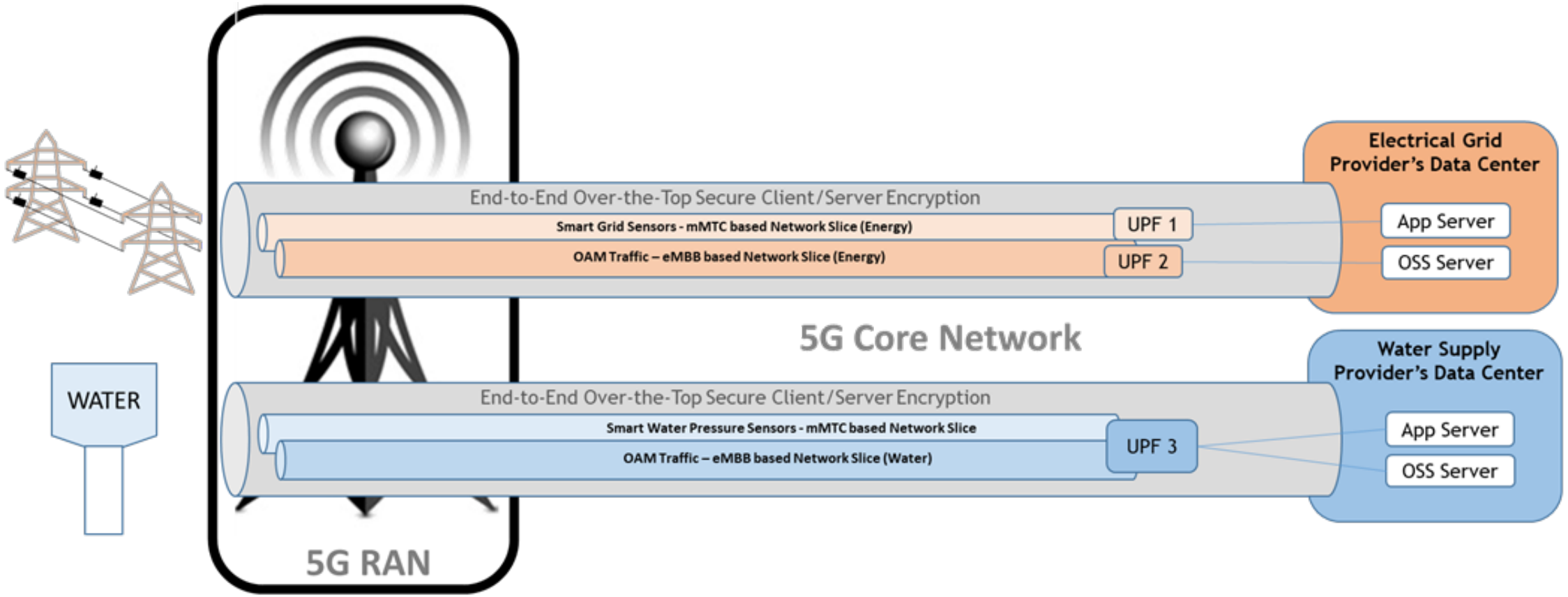
Roles of 3GPP and GSMA in NESAS



NESAS High Level Overview



Network Slicing and Secure Tunnel



Open-source software security benefits and risks

Benefits		Risks
Developers behave as “good citizens” in which consumers also contribute, provide useful feedback, and share fixes.	⟷	Intentional backdoors can be inserted by malicious developers.
Transparency of code. Many expert eyeballs reduces software complexity and the number of bugs. This crowdsourcing approach effectively produces quality software at low cost.	⟷	Attackers can review code to identify vulnerabilities.
Open source provides a platform for talented coders to openly collaborate and build software.	⟷	Developers do not spend sufficient time on security. Vulnerabilities can propagate through reuse.
Open source also reduces fragmentation and increases interoperability among different products by producing components and protocols that become the de facto standard.	⟷	‘Trees of dependencies’ make it difficult to ensure all uses of the code are patched.