Moving from a Traditional Branch to a Virtualized Branch

This material includes confidential and proprietary information of Versa Networks. It may not be replicated or distributed without the written consent of Versa Networks. This document may include forward looking roadmap and product strategy information from Versa Networks. It is intended for informational purposes only and should not be interpreted as a commitment on the part of Versa Networks. Versa Networks makes no warranties, expressed, or implied on future functionality and timelines in this document.

Moving from a traditional enterprise branch boundary to a virtualized branch boundary involves consolidating network, computing, storage, security, and application capabilities. To effectively realize a virtualized branch boundary, these unified stacks must be delivered with continuously verified security and as close as possible to users, devices, and things. A combination of "Versa SASE + MEC" serves as one of the effective delivery mechanisms to facilitate this transition to a fully virtualized, software-defined boundary.

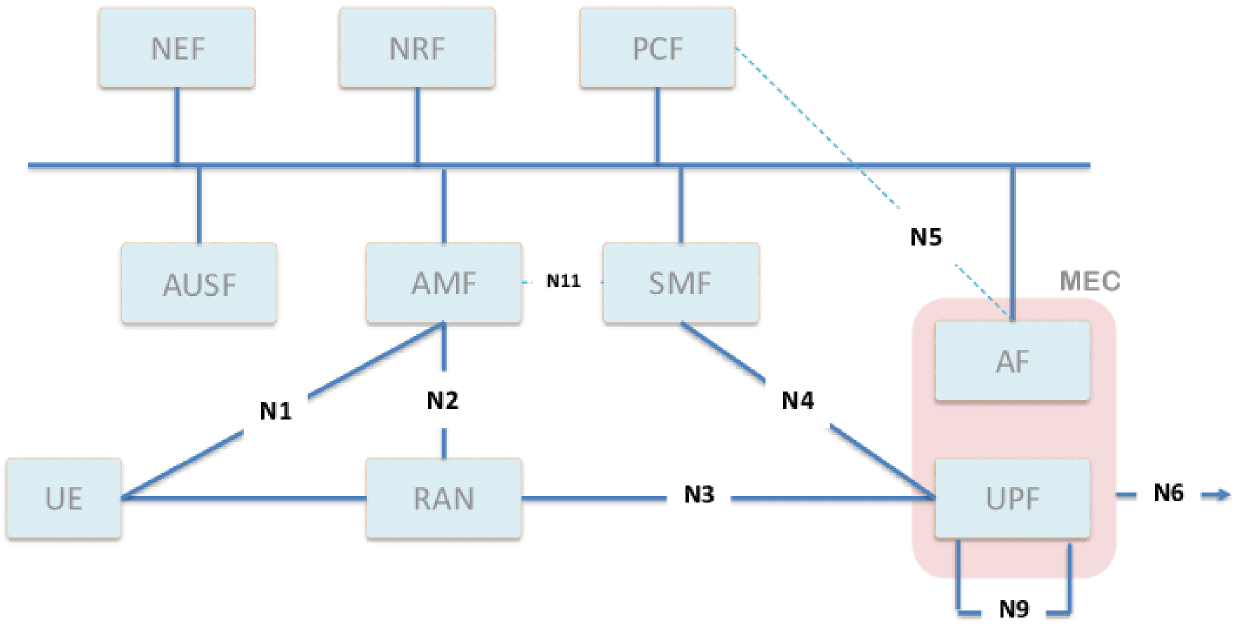
The following are key tenets of Versa to facilitate this transition:

Software-based Hardware Neutral

Ideal architecture should be software- and hardware-neutral, making it consumable on any COTS server. This allows end-customers to flexibly change as and when they need to, since the solution is not tied to any vendor-specific proprietary hardware architecture. There is no vendor lock-in.

Platform Agnostic Full Stack Security

A key function of a good MEC solution is to deliver SASE security blended with robust routing and SD-WAN. This includes IPS, IDP, DNS protection, DLP, URL Filtering, SSL proxy, SWG, and Zero Trust Network Solutions to protect against unauthorized access, malware, and ransomware. This should hold true irrespective of communication between MEC and core-cloud (N4), Internet egresses (N6), switching contexts between MECs (N6, N9), multi-clouds, or the enterprise domain.



Single Pane of Glass Management

An ideal architecture should provide centralized management and orchestration capabilities. The resulting single-pane-of-glass platform should be able to catalog, define, deploy, and manage policies and service chains of different VNFs hosted on the MEC platform. It should also deliver a unified visibility plane into traffic going through VNF and VNF service chains.

Carrier-class Routing

To meet demands for scalability, adaptability, and fast failure convergence, a MEC architecture should fully support common and advanced routing protocols, which are crucial for WAN and LAN network agility. These include Static, OSPF, BGP, MP-BGP (MPLS-based L3VPN, MPLS-based EVPN, VXLAN-based EVPN), RIP, multicast (IGMP, PIM), VRRP, PBR (policy-based routing). It should also support IPv6 on both the LAN and WAN interfaces and all permutations of dual stack, along with Carrier Grade NAT (CG-NAT)

Application-aware Intelligent Steering

This can be achieved through application-based traffic steering and per-tenant QoS/App-QOS. Application-based traffic steering should be able to auto-identify applications and then make intelligent traffic routing decisions based on the SLA required for that application. This ensures that users always experience the best application performance.
Hierarchical QOS ensures optimal resource allocation, QoS, adaptive shaping, and per-tenant-based resource allocation.
Typical Versa Hierarchical QoS use cases:

- Per CN operator level and forwarding class
- Time of day QoS policy (well suited for IoT apps)
- Hierarchical shaping to allow traffic to be scheduled and shaped at a traffic-class level, logical interface level, encrypted tunnel-level, and/or physical port level.

Support for Software Defined LAN Services

Programmable LAN fabric is essential to enforce granular application segmentation at the L2 domain, allow for layer 2 multi-pathing, eliminate legacy layer-2 challenges, and provide easy service insertion for MEC components.

Bring Your Own VNF with Robust Fault Management

The MEC solution should be flexible to support and manage a plethora of 3rd party VNFs. It should provide native capability to monitor different VNFs through health checks and provide fast failure convergence if any of the VNFs become unhealthy (by pass on fail).

In-Line Encryption/Decryption that Scales

Encryption/decryption are compute-intensive and may impair application performance. SASE solutions for MEC include options for software and hardware-accelerated encryption and decryption that provide faster processing and tamper-resistant key storage.

Environment Agnostic Multi-Tenancy

The solution should be an enabler to overcome native multi-tenancy limitations by delivering complete segregation of control-plane, data-plane, and management-plane for each of the tenants. Each tenant should have multi-level RBAC (role- based access control) to manage the network based on the roles and responsibilities. This reduces the footprint and costs for MVNOs and expedites service rollout.

Tight Integration with Network Slicing

The solution should be able to segregate different slices into different SD-WAN overlays. This ensures full isolation of control and data traffic. Furthermore, a mature MEC solution should support different contextual security mechanisms within each SD-WAN overlay. The Security and routing policy for each slice should be SASE driven and based on user, device, and context. The solution should deliver powerful Role Based Access Control (RBAC) for slice management.

Flexible Third-Party Integration

The MEC solution should be able to flexibly integrate with the Mobile Service Orchestrator or any other 3rd party OSS/BSS system. The solution should publish 100% of its Restful APIs for its Management Plane (alarms, monitoring, configuration, analytics) to provide such integration capabilities. It should also support flexible, global Integration with all flavors of MEC Zones from HCPs (Azure, AWS, Google, etc.) to support a wide variety of business use cases.

Zero-Touch Cloud Instantiation

To ensure faster and more seamless deployment, the complete orchestration of the MEC hosting components should be done with true zero-touch provisioning. This methodology significantly cuts down cloud instantiation times and operational involvement.