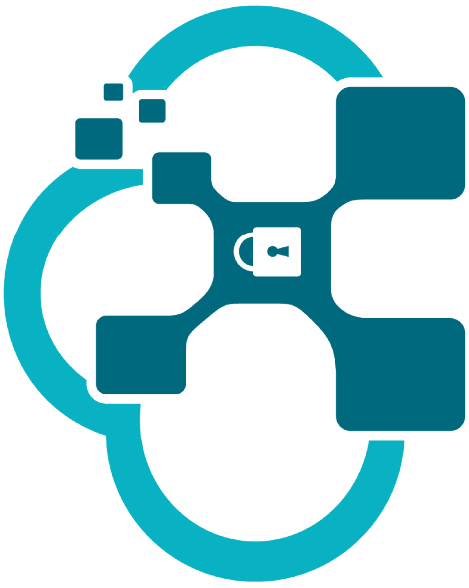
Packetmen



SD-WAN: Best of Breed or Add-On?

By Erik Fritzler – www.packetmen.com

Everyone in the networking world has, by now, been exposed to their vendor-of-choice's SD-WAN solution. I've seen WAN optimization companies, router and switch companies, and virtualization software companies all add some form of SD-WAN capability to their products. Some of these companies designed the product themselves, and others acquired the technology through acquisition. This leaves the question; Do I buy best of breed, or all-in-one from my existing vendors? And, what exactly *is* best of breed?

Best of breed solutions take innovation to a new level by building their product from not only through the lens of better technology, but better business impact. They answer the critical question of how to deliver a business result as opposed to how to deliver a technology.

I wrote, in a previous article, about the difference between application classification and packet classification. Routers, firewalls, and most of the SD-WAN devices on the market rely on classification by source addresses, destination addresses, port numbers, and some rudimentary deep packet inspection. They function by inspecting each packet and classifying them according to static rules. This quickly becomes unmanageable even within a small enterprise due to the sheer number of applications present. The everchanging landscape of business SaaS solutions adds even more complexity to the mix. With dynamic IP addresses, changing ports, and hostnames that are used to provide more than one service with varying behaviors (think bulk data transfer vs transactional vs collaboration). Packet-centric devices simply can't accurately identify what application is in use, and worse yet, the rules are defined using low-level parameters, rather than simply saying "I want an Office 365 policy".

Best of breed SD-WAN solutions have the application detection capabilities of packet-based SD-WAN solutions but view the world through the

lens of application sessions rather than singular packets. Applications are identified by both *static* characteristics (such as the identification done by packet-centric systems) as well as *dynamic* characteristics, including cross-flow correlation, signature examination, and heuristics, allowing the administrator to say "I want to identify Office 365". The application centric view provides a much simpler and more robust method of defining and managing policy by making the application the central anchor of policy definition. I would much rather setup a policy for Google Drive, Dropbox, or Office365 by clicking a box rather than creating an access list of several dozen, if not hundreds, of lines to classify the traffic; and manage it continually as services and IP addresses change. Most packet-centric SD-WAN solutions are relegated to using "Intranet HTTP" and "Internet HTTP" which leaves little room for granularity. To make matters worse, many SaaS services offer applications with varying characteristics behind the same hostname, making it impossible to define an appropriate performance and QoS policy for both bulk transfer applications and voice collaboration services that reside behind the endpoint.

All-in-one vendors that claim to offer everything and the kitchen sink generally have a series of loosely-integrated "good enough" products. These products meet the main checkboxes required by businesses that are making purchasing decisions, and having such a wide product line puts the vendor in the position to do pricing and bundling gymnastics to make their inferior "good enough" products either price competitive or free. These tactics are commonly followed by "no one was ever fired for buying from XYZ".

Modern IT is very different than the IT of ten years ago. Tightly-integrated systems from a single vendor had tremendous value, but we now live in the age of open systems with well documented APIs, value-added integration, and developer operations (DevOps). IT is able to extract far more value out of an API-centric product – even if it's from a new vendor – than they are from a product that doesn't push the envelope from a business or technology perspective from a large established vendor with creative pricing and bundling. Put simply, "good enough" doesn't cut it when I/T's ambition is to reduce cost and complexity, and the only way to achieve that goal is through automation, which demands a robust DevOps ecosystem for the products they choose to deploy.

Good-enough all-in-one vendors tend to focus on meeting the "minimum viable product" needs – the bare minimum - when bringing a new product to market. Market consolidation usually happens first with large vendors that feel their install base is threatened and their acquisition target usually becomes the startup with first mover advantage, and those startups tend to have the highest revenue given their age in their respective nascent market. However, these early entrants didn't have the benefit of taking a moment to allow a market to mature prior to building their fundamental architecture.

That is, at the core, why every first-generation SD-WAN system available from large vendors lacks the architecture required to meet today's demanding DevOps needs. And, why they fail *painfully* when customers try to integrate these products using APIs and automation.

Having a robust set of APIs, developer documentation, software development kits, scripts, and open source code samples form the basis for a DevOps-friendly platform and helps provide radical efficiency for I/T organizations that want to automate. The recent push for network to code validates this viewpoint for the entire lifecycle of I/T products, including installation, configuration, provisioning, management, monitoring, and troubleshooting. These tasks are time consuming and cumbersome for traditional network infrastructure *and* first generation "good-enough" SD-WAN solutions, and they are also error prone and inefficient on many different levels. You often have clunky CLI interfaces that provide poorly integrated management of the solution.

Finally, beyond application-centricity and automation, I/T organizations demand a system that is self-healing. With application-centricity, one can define a policy that says "I care about Office 365 and it should receive a gold level of service". An application-centric system will not only be able to accurately identify the application, but also understand the performance and health characteristics of the application transactions themselves. Not just bandwidth, latency, loss, and jitter, which are the staples of "good-enough" SD-WAN from "all-in-one" large vendors, but the actual characteristics of the application. An application-centric system will use machine learning to understand how an application will behave under a given set of current network conditions to control and forward an application on the link that will provide the best possible user experience – unlike "good-enough" solutions which will just look for the best available WAN link, with no consideration for how the application will perform.

To summarize, don't fall into the trap of "good enough" SD-WAN. Vendors that provide this "all-in-one" type of solution may look good on paper – meeting a large number of checkboxes. Look deeper into what the business value is and whether or not these solutions are congruent with your goals of aligning WAN policy and management with actual business goals, whether or not you can automate the infrastructure programmatically, and if the WAN will respond and make decisions based on what's actually going to be best for your users rather than using low-level metrics that don't consider the user experience.