netskope

WHITE PAPER

The Network Is the Security: Perspectives on the design & implementation of security clouds

Mark Stuart Day
Chief Scientist, Netskope

Starting in the 1980s, the slogan "the network is the computer" was a trademarked phrase advocating two causes: both a new perspective on what a computer was, and buying that particular manufacturer's computers. At the time, a computer was usually understood to be a box. That box might be the size of a room, or only the size of a closet or table, but "a computer" was the natural unit for both end users and programmers. In contrast, modern computing is almost entirely based on the idea of multiple computers interacting over a network, with some computers serving as sophisticated user interfaces, others providing powerful computations, still others supporting massive storage with efficient indexing and queries.

Today, "the network is the security" is a new slogan with echoes of the older one, defining a transition from an old way of thinking to a new, more sophisticated approach. It likewise advocates for a dramatic rethinking, this time of what security is. Then, the box-focused "computer" dissolved into something more distributed and more intelligent. Similarly, today's box-focused security is dissolving into something more distributed and more intelligent. The end goal is to deliver superior security in a complex, ever-evolving world of threats and risks to the digital assets that are central to the value of most modern enterprises.

Until fairly recently, most enterprises hosted their data and applications in enterprise-managed data centers. Accordingly, the role of enterprise networking was primarily to support secure connectivity to the data center. The security services inspecting traffic and enforcing policies had certain features in common: They were protecting traffic flowing to and from the data center, and they were packaged as devices running in the data center.

Recent years have seen the data and applications of interest migrate out of the data center and into the cloud. Two examples are Microsoft Office 365 for office productivity tasks, and Salesforce.com for sales automation and customer relationship management. Accordingly, it's no longer realistic to treat the data center as the only concern for connectivity and security.

Instead, the primary focus becomes the user, and the user's varied requirements. The emphasis is now on an integrated platform of connectivity with embedded security services. [MacDonald+ 2019] From the user's perspective, this is just another kind of cloud: one that delivers various flavors of secure connectivity. Accordingly, we can refer to this access/security platform as a *security cloud*.

From the perspective of enterprise IT, a security cloud is appealing because it replaces a jumble of different connectivity and security technologies. Perhaps more importantly, the traditional approach is no longer workable: A collection of physical or virtual appliances daisy-chained in a data center simply won't work after the apps, users, and data have moved away.

The security cloud can become how *every* user reaches *every* service. It combines the best aspects of an enterprise WAN and the internet, with high-performance secure connectivity to a broad collection of destinations. It is a path for traffic to flow between users and application services; a set of inspection and policy-enforcement mechanisms to avoid security problems like malware or data leakage; and a means of isolating an organization's public services from direct internet connections, making them more difficult to find and attack, as well as mitigating the risk from lateral movement.

In this paper, we consider the design and implementation choices involved in realizing this appealing model of a security cloud. We focus primarily on what analysts refer to as the Secure Services Edge (SSE), but many of our points are also relevant to the broader category called Secure Access Service Edge (SASE). (These terms are intended to be helpful for those who are familiar with them, but can be safely disregarded; the rest of the paper does not depend on them.)

In simplest terms, the key requirements of a security cloud are:

- A distributed implementation with relatively intense computations performed at a relatively large number of locations, and
- Management and visibility tools sufficient to build an elastic but high-performance service

We consider three seemingly plausible ways to design a computing and network infrastructure to meet those requirements: using a public compute/storage cloud, using a global network provider, or using a CDN. We explain the drawbacks of each approach, and then describe our approach and how it overcomes those drawbacks.

THE REQUIREMENTS OF A SECURITY CLOUD

Broadly speaking, a security cloud must:

1. Support connections from enterprise users all over the world, on diverse devices. Part of that diversity includes both IT-managed devices and IT-unmanaged devices (the latter sometimes called “bring your own device,” or “BYOD”).
2. Control access to a variety of cloud services, including diverse SaaS services as well as enterprise app workloads that may be deployed in a multi-cloud arrangement.
3. Perform inspection and policy enforcement on enterprise traffic traveling between users and cloud services.
4. Maintain proper localization and proximity information for each user/service combination.
5. Be fast. More specifically, a security cloud needs to be fast enough so that user experience doesn’t motivate users to bypass or reject the security controls provided by the security cloud.

The key problems for good performance are distance (speed of light constraints) and crossing multiple networks (peering and market constraints). Both of these problems are well-known to network architects. A centralized implementation of a global security cloud cannot achieve high performance. Instead, the security services must be globally distributed: both for performance, and also for availability and resilience.

DETERMINING PLACEMENT OF PROCESSING

There are three potential sources of overhead to be minimized:

1. the network distance from the user to where traffic processing occurs;
2. the cost of security processing, both in terms of necessary resources but also the time constraints or latency of performing this processing; and
3. the network distance from security processing to service (for example, an organization's Office 365 or Salesforce instance). Network distance here refers to the effective distance, as measured by the time required to send data, rather than purely geographic distance.

Effectively, the security cloud is being "spliced" between the end user and the service, and the goal is to minimize the added network distance on both sides of the security cloud.

Even if we consider all of the service destinations to be "front doors" for a single popular cloud service like Microsoft Office 365, it's entirely reasonable to think that a user in the New York metro area would be accessing that particular cloud service somewhere relatively nearby. Likewise, a user in the Los Angeles metro area would be accessing that cloud service relatively nearby. There are lots of real-world user/service pairs that would experience substantial performance problems if their traffic needed to be diverted to somewhere relatively far away. (Such diversion + return topologies in a network are sometimes disparaged as traffic "hairpinning" or "tromboning.") Performance demands that we put security services near the users. Since the users and the application services of interest are spread around the world, there is a corresponding need to put security services all around the world.

We conclude that a security cloud must include a security processing system that is highly efficient at doing the necessary inspection and policy enforcement tasks. (It's worth noting here that most of this traffic is encrypted, and must be decrypted before it can be inspected.) That highly efficient processing must be placed at multiple locations around the world, in a way that minimizes the network distances to end users and the cloud services they access. The security cloud designer's slogan might be: "a lot of compute, in a lot of places."

MANAGEMENT AND VISIBILITY

Although placement of processing is a crucial issue for high performance, it's not the only architectural concern in realizing a security cloud. A crucial operational counterpart is visibility into the operating characteristics of the security cloud's network connectivity and its processing resources, as well as the management mechanisms to adjust the cloud implementation accordingly.

In particular, we need capacity management tools that would allow us to implement the security cloud's abstraction of elasticity. Those tools would allow us to determine levels of utilization and possibly rearrange physical computing infrastructure accordingly.

This requirement is very unlike the typical cloud workload, and indeed somewhat contradicts the cloud paradigm. In general, capacity management tools are not exposed to customers of a cloud, or they're only exposed in a simplified form. Customers of a cloud are effectively giving up that detailed level of visibility and control, in return for the cloud provider doing the work for them.

That simplified approach is an excellent tradeoff for lots of compute and storage tasks that may not be especially time critical, or may not have tricky interactions with networking behavior—but that's emphatically not the situation for implementing the security processing of a security cloud. Instead, a security cloud by its nature is concerned with efficiently intercepting, decrypting, inspecting, and re-encrypting a vast number of connections without degrading application/service performance for end users.

REALIZING A SECURITY CLOUD

Now we consider some ways that we could realize this design. There are three obvious possibilities that use existing clouds as the implementation technology: a public compute/storage cloud, a network cloud, or a content-distribution cloud. We take up each in turn so as to understand its limitations.

Each of these approaches will "work," in the sense that it is possible to build a security cloud using those facilities. However, each of the approaches comes with serious deficiencies that will be a concern for anyone trying to build and operate a large-scale, high-performance security cloud. This analysis is not only relevant to vendors building security clouds—understanding what's "under the hood" of a security cloud is critical for enterprise IT leaders to choose an appropriate one to meet their long-term goals.

Approach #1: Public compute/storage cloud

The first simple approach to building a security cloud uses the facilities of one of the existing major public compute/storage clouds (Amazon Web Services, Microsoft Azure, Google Cloud Platform, and others).

At a high level, this approach seems workable: A public compute/storage cloud has multiple locations for security processing, and those locations are accessible via multiple networks. As a bonus, a public compute/storage cloud offers the opportunity to offload some of the chores involved in provisioning and managing the security-processing computations we need to run. The public compute/storage cloud infrastructure will take care of the hardware resources, so we don't have to worry about those details—we just design our security-scanning and policy-enforcing computations so they scale with the traffic being sent to us by customers, and we're all set.

This high-level summary explains why some companies have picked this route to implement a security cloud. That choice is especially understandable where time-to-market is a driving factor, where financial backing is limited, and/or where key expertise is lacking or unavailable. But a closer analysis identifies some important problems. We really want to have lots of relatively small, but efficient processing locations in lots of different locations. A plausible granularity for these locations is about 2 Tb/s of connectivity and associated processing power. Depending on the volume of customer traffic to be handled, we might expect to deploy dozens of these locations around the world.

When we think about deployment in terms of that size "chunk" of computing and networking, it's not hard to get a suitable-size chunk of networking/computing/storage in one of the big public clouds. However, it's essentially impossible to get those chunks in all the places we might want to have them.

The problem is that a public compute/storage cloud provider builds its infrastructure to support its *total customer workload*, and in a way that is *profitable for the public cloud*. Let's assume, for example, that we have customers in Latin America. At this writing, the largest public compute/storage cloud has only a single site in South America—in Sao Paulo (Brazil). Will the public compute/storage cloud provider open a new data center in Bolivia if that's the best location to receive traffic from customers needing security services? That's not likely.

These data centers are enormous operations, and it's a big deal to create a new one. Real estate experts refer to Amazon presentations indicating that an AWS standard data center includes 50,000 to 80,000 servers, in 150,000 to 215,000 square feet. [Miller 2017]

So that's one problem with building a security cloud this way: It's at the mercy of what the public compute/storage cloud chooses to build. That market is driven by constraints that are unrelated to the design goals we've identified. The commercial reality is that even building a "big" security cloud for lots of enterprise customers is still pretty small when compared to the scale of the three main public compute/storage clouds. Revisiting our slogan of needing "a lot of compute in a lot of places," we find that with a public compute/storage cloud we can get "a lot of compute," but we can't get it "in a lot of places."

But let's suppose for a moment that we could solve that problem: Perhaps we can use a hybrid implementation technique like AWS Outposts [Amazon 2020] to put the security-cloud processing where we need it. With those changes, is a public compute/storage cloud a good choice?

Unfortunately, the answer is still probably "no." There are two structural problems, one about capacity management and one about network design.

1. *Capacity management*: Although a public compute/storage cloud has internal tools to manage its physical resources, those are not exposed to customers of the cloud. The lack of management tools is a key reason that an "Outposts"-like solution doesn't really work for this kind of use case: As a cloud customer, we don't really know enough about traffic and capacity to decide when or where to revise our supplemental deployment. Of course, this specific problem could be fixed. The larger issue is that there is an inherent conflict between the cloud service's wish to abstract away these kinds of concerns for customers and a cloud-builder's wish to see and control exactly these kinds of concerns.
2. *Network design*: A public compute/storage cloud is designed for the cloud provider's goals, which are only partially related to customer goals. We've noted that the public cloud's network can work well for traffic that's staying inside that particular cloud, but it generally doesn't work as well for traffic that needs to leave that particular cloud. As a business issue, every public compute/storage cloud wants to keep processing and storage "inside" its cloud—and designs its network and its pricing structure accordingly. These clouds typically have egress charges, imposing financial costs for data that leaves the cloud. In contrast, a security cloud by its very nature isn't about *capturing* or *keeping* traffic—instead, it's about *inspecting* traffic. Accordingly, the traffic is only crossing the security cloud, rather than terminating there. For similar reasons, a security cloud has to be concerned with latency in a way that a compute/storage cloud generally doesn't.

The problem isn't that public compute/storage cloud networks are designed by stupid people—quite the opposite. The networks of public compute/storage clouds are designed by smart people to solve a different problem. Public compute/storage clouds are *destinations*, not *pathways*. In contrast, a security cloud is not a destination: A security cloud is not attempting to hold or own any customer's computations or data. Instead, a security cloud is more like a passageway to an application service.

The economics and performance of a public compute/storage cloud are accordingly quite different from those of a security cloud. Although it's *possible* to build a security cloud within the constraints of a public cloud, these realities mean that any such implementation will always be limited and inadequate in some important ways. Although it can be useful to take advantage of the public cloud tactically, it's a strategic mistake to commit to the public cloud exclusively: A security cloud vendor that depends exclusively on the public cloud may find itself at the mercy of the public cloud provider(s).

Another way of thinking about the situation is that a public compute/storage cloud is structured to deliver a cloud service, not a cloud-building service. A public compute/storage cloud offers its customers elastic versions of computing and storage services, but is not particularly oriented toward exposing the tools and processes that underpin those elastic abstractions.

Approach #2: Public network cloud

The next likely design choice is to use a large public data network, perhaps a so-called "Tier 1 carrier" like Level 3. Such a network typically emphasizes the sheer number of locations connected and the sheer number of users or organizations reached by the network, and so seems to be in line with the goal of building a very broad-reaching security cloud.

The carrier is acting as a "one-stop shop" for transit and peering, but as a customer of that one carrier the security cloud doesn't see anything except traffic going in and out of the carrier's facilities. The carrier has traffic engineering and capacity management tools that are not exposed to their customers—in this respect, their limitation is quite similar to one that we saw in the previous section with public compute/storage clouds. Our would-be security cloud is again confronting the dark side of the cloud abstraction: Although most customers appreciate the simplicity of a "network cloud," it's not a good match for the requirements of building a high-performance security cloud.

The single large carrier can work well at moving data between its own network's locations, but that's not the crucial problem to solve when building a security cloud. The security cloud needs to solve a version of the "internet performance problem," and networking experts have known for a long time that isn't possible with only a single network operator.

Approach #3: CDN

Trying to get good performance from the internet is not a new problem. For more than 20years, some performance-oriented organizations have used a different architectural approach. [Zolfaghari+ 2020] A content distribution network (CDN) implements a performance-focused network by using the internet as the underlying transport. Rather than using the default, lowest-common-denominator interactions among networks, a CDN substitutes its own management and coordination mechanisms. A CDN is a "virtual network" that uses the networks of the internet selectively to achieve better results than the "ordinary" internet could do.

A CDN has a promising network architecture: The overlay network has a presence in many diverse networks. Most CDNs also offer some "edge computing" facilities that could in principle be used for security processing. Some even sell a kind of security service themselves—could they be a good platform for a security cloud?

In a 2020 earnings call, the CEO of Akamai noted that their network had "over 300,000 servers in over 4,000 locations." [SeekingAlpha 2020] That count suggests that the raw capacity of the Akamai network could readily meet our requirements, if we had exclusive access to it. However, those 75-ish servers per average location are being shared among all Akamai customers. Since Akamai also claims roughly 950 enterprise hardware and software customers and 825 retailer customers, it seems like those servers would be heavily utilized. There's enough capacity for modest extensions of CDN capabilities, but probably not enough for our purposes. In terms of our slogan, edge computing offers "a lot of locations," but doesn't appear to offer "a lot of compute" at each of those locations.

In addition, as with the public compute/storage clouds, there are problems with capacity management and network design. The capacity management problem is quite similar to what was described for the previous two approaches—the CDN operator has capacity management tools that are not available to a customer, but the security cloud needs that kind of insight to operate well.

The network design problem is quite different from what we encountered previously. We observe that CDNs are built to support content distribution and/or DDoS mitigation. Content distribution uses multiple distributed copies of a popular resource (such as a web page or a video) so as to make that popular resource available at larger scale and faster speeds than would be possible for a central server. Successful content distribution effectively absorbs requests at the edge so as to offload the origin server. DDoS mitigation is the opposite problem of content distribution: Instead of satisfying the requests, the distributed resources (mostly) discard them. In content distribution, a large number of legitimate requests are successfully answered by the distributed resources; in DDoS mitigation, a large number of attacks are successfully "scrubbed out" and discarded by the distributed resources. In both paradigms, the CDN is filtering or attenuating the original received workload, thus offloading traffic from the primary server. What passes through the filter to the origin server is either relatively unpopular content, or not part of an attack.

A security cloud's policies will likewise block certain requests by users, and block certain responses by services. However, in contrast to a typical CDN's behavior, those blockages are typically only a tiny fraction of the traffic that is processed. To a first approximation, a security cloud delivers all of the traffic that it receives, absorbing none—the very definition of failure for a CDN, whether being used for content distribution or DDoS mitigation.

A secondary concern relates to using a shared generic topology instead of a custom topology, and is somewhat similar to a problem of "not enough places" identified with approach #1. These kinds of concerns have prompted large streaming providers (such as Netflix) to build their own custom CDN: contrast the description of Open Connect [Florance 2016] with the description and measurement of Netflix's previous

multi-CDN strategy. [Adhikari+ 2012] The Netflix CDN is customized to support the Netflix workload, rather than having to take a slice out of a generic CDN that is being rented out piecemeal to many different customers.

As with the previous candidate architectures, CDNs do well what they were designed to do. The unsuitability of a CDN for a security cloud is not because CDNs were designed by foolish people—rather, CDNs were designed by smart people to solve a different kind of problem. Although there are aspects of CDNs that make them closer to our requirements than either public compute/storage clouds or public carrier networks, they're still not easily adaptable into security clouds. Instead, what we need is to apply the relevant good ideas from a CDN architecture, mix in the security cloud facilities that are missing from a CDN, and build the right architecture for a security cloud.

Approach #4: Netskope's better approach with a purpose-built private cloud

A security cloud needs to combine the diverse network presence of a CDN with the elasticity of a public compute/storage cloud, and add a focus on high-performance security services. The Netskope NewEdge global security private cloud has been built to meet these requirements. One perspective is that NewEdge is a little like the custom CDN that Netflix built—but instead of a focus on supporting the delivery of streaming video, it's designed to support the security processing of enterprise traffic. In line with our cloud designer's slogan, NewEdge has "a lot of compute," and crucially also has it in "a lot of places." As we've seen, that combination is not easily achieved with a public compute/storage cloud ("a lot of compute in a few places") or with edge computing ("a little compute in a lot of places"). At this writing, NewEdge already has more security-processing locations—at the edge, close to users—than even the largest public clouds.

NewEdge is also how we realize our initial motivational slogan, "The Network Is the Security." With NewEdge, we have an approach that meets our initial goals for a security cloud, with no compromises. However, this approach comes with unavoidable costs. In moving away from a single compute/storage cloud, a single data network, or a single CDN, the security cloud implementor must be prepared to manage interactions among multiple networks. Likewise, if it's no longer possible to rely on someone else's elastic compute/storage infrastructure, the security cloud implementor must be prepared to implement an elastic service. Such an implementation in turn requires considerable complexity and sophistication of operations. In short, going down this path is not for the faint of heart.

Fortunately, there are benefits to match the costs. By exercising control over routing, peering, and traffic engineering, Netskope's custom network gives the multi-network performance virtues of a CDN, while avoiding the limitations of a CDN's focus on filtering traffic. At the same time, our custom computing infrastructure (and the associated management practices) give the capability of placing security processing close to end users and scaling the service up or down with variations in demand, like a public compute cloud. In contrast to any large public compute cloud, Netskope NewEdge supplies these elastic capabilities purely for the transient security processing of enterprise network traffic.

Even though Netskope operates the NewEdge network, we can still take advantage of public compute/storage clouds for security processing whenever it's the right tool for the specific task, either temporarily or over long periods of time. In contrast, a security cloud provider that is dependent on a public compute/storage cloud does not typically have the skills or facilities to do something like NewEdge for themselves. With a base infrastructure like NewEdge, Netskope is equipped to "burst" to the public cloud; any competitive security cloud provider that's based in the public cloud has no corresponding ability to "burst" to a private cloud.

By count of diverse metro compute locations, Netskope NewEdge is the largest security private cloud in the world. By count of network adjacencies, Netskope NewEdge is also one of the best-connected security clouds in the world. [HE 2021] In contrast to a public-cloud approach, it delivers more-distributed computing. In contrast to a single-carrier approach, it has more effective task-focused peering and routing. In contrast to a CDN, it is built to deliver substantial computing at the edge, and is built as a pathway rather than as a filter.

These architectural choices allow Netskope to deliver industry-leading service level agreements (SLAs) for customers. [Netskope 2021] Netskope is the first security cloud provider to offer any SLA for encrypted traffic, despite the reality that today's enterprise traffic is mostly encrypted. In addition, Netskope's SLA for unencrypted traffic is more than 10x faster than the matching SLAs of key competitors.

Although much depends on the choice of sites, traffic, and policy, a common customer experience with Netskope NewEdge is that the full "security cloud" configuration is roughly comparable in end-to-end latency to the "no security cloud" configuration. Effectively, the design choices in NewEdge gain enough improvement in network latency to pay for the security processing time. Enterprise customers, some with hundreds of thousands of users, have reported this advantage translates to a noticeable user and application experience improvement. Examples include anecdotal reports of "improved application throughput of up to 50%" in one case, and "6x improvement for specific SaaS apps" in another. In another striking example, a top-5 global bank asked to run their traffic over NewEdge just for the network performance gains they experienced, independent of their specific security requirements.

CONCLUSION

At a high level, a security cloud appears to "only" require the application of computing to traffic. That requirement is relatively easy to meet by using simple approaches. However, a more careful analysis shows that a security cloud requires substantial computing near users, excellent visibility into capacity, well-managed peering, and an architecture that supports the low-overhead transit of traffic. Seen in that light, a security cloud requires thoughtful attention to network design. Not all security clouds are alike! It's critical for enterprise IT buyers to understand those design differences—especially for any organization reconsidering their cloud strategy, seeking new capabilities for better security and data protection, or addressing gaps in their existing cloud-based solution.

Netskope NewEdge is a purpose-built global security private cloud that addresses these requirements more effectively than alternative approaches. To learn more about Netskope, the world-class security and data protection capabilities of the Netskope Security Cloud, and the underlying NewEdge network infrastructure that powers it all, please visit: https://www.netskope.com/platform/newedge.

References

[Adhikari+ 2012] Vijay Kumar Adhikari, Yang Guo, Fang Hao, Matteo Varvello, Volker Hilt, Moritz Steiner and Zhi-Li Zhang. “Unreeling Netflix: Understanding and improving multi-CDN movie delivery.” Proceedings of INFOCOM 2012.

[Amazon 2020] “AWS Outposts.” https://aws.amazon.com/outposts/

[Florance 2016] Ken Florance. “How Netflix Works With ISPs Around the Globe to Deliver a Great Viewing Experience.” https://about.netflix.com/en/news/how-netflix-works-with-isps-around-the-globe-to-deliver-a-great-viewing-experience

[MacDonald+ 2019] Neil MacDonald, Lawrence Orans, and Joe Skorupa. “The Future of Network Security is in the Cloud.” Gartner, 2019.

[HE 2021] Hurricane Electric peering report, https://bgp.he.net/country/US

[Miller 2017] Rich Miller. “Amazon plans epic data center expansion in northern Virginia.” Data Center Frontier, 6 November 2017. https://datacenterfrontier.com/amazon-plans-epic-data-center-expansion-in-northern-virginia/

[Netskope 2021] “Netskope Sets New Industry Benchmarks for Cloud Security Performance; Announces Industry-First SLA to Address Encrypted Traffic Processing.” Netskope press release, September 2021. https://www.netskope.com/press-releases/netskope-sets-new-industry-benchmarks-for-cloud-security-performance

[SeekingAlpha 2020] “Akamai Technologies' (AKAM) CEO Tom Leighton on Q2 2020 Results - Earnings Call Transcript” Seeking Alpha, 28 July 2020. https://seekingalpha.com/article/4361639-akamai-technologies-akam-ceo-tom-leighton-on-q2-2020-results-earnings-call-transcript

[Zolfaghari+ 2020] Behrouz Zolfaghari, Gautam Srivastava, Swapnoneel Roy, Hamid R. Nemati, Fatemeh Afghah, Takeshi Koshiba, Abolfazl Razi, Khodakhast Bibak, Pinaki Mitra, and Brijesh Kumar Rai. “Content Delivery Networks: State of the Art, Trends, and Future Roadmap.” ACM Computing Surveys, Vol. 53, No. 2, Article 34, April 2020.