Better data. Better decisions.

Operationalizing Zero Trust

Change how you think, architect for success, and integrate a broad tool set.

By John Burke, CTO

Nemertes

February 2022

Table of Contents

The Problem: Putting Zero Trust Into Practice is Not Easy **- 3 -**

Step One: The Radical Rethink—Start with a Zero Trust Architecture **- 3 -**

Standing on the Shoulders of Giants: Begin with the NIST Architecture *- 4 -*

Infrastructure, Applications, Data, Devices *- 5 -*

Step Two: Rearchitecting Networks and Security **- 5 -**

Not One Tool, Many Tools Working as One **- 6 -**

Step 3: Starting Points for Operationalizing Zero Trust **- 7 -**

DIY vs Managed Services *- 7 -*

Conclusion and Recommendations **- 8 -**

Zero-Trust Checklist **- 9 -**

The Problem: Putting Zero Trust Into Practice is Not Easy

In a zero-trust environment, no person, no machine, no part of a network is assumed to be trustworthy. All trust relationships have to be explicitly stated, are conditional on good behavior, and therefore are temporary.

Zero trust (ZT) is a bad name for a crucial idea: rooting out *implicit* trust. As such, it centers on three major shifts in the enterprise security environment:

- Contracting security boundaries so every entity has its own "perimeter of 1"
- Reconfirming identity and authorization on every action
- Watching behavior after admission to an environment to spot bad actors

These represent major changes from current practice for most organizations, and especially for network and security teams. Most teams still think mainly of admission control—verifying identity when something or someone joins the network—not of active, ongoing monitoring of behavior with good behavior a prerequisite of continued access. Likewise, they (and most other parts of IT) are used to thinking of different parts of the network as corresponding to trust zones, some of which are more trustworthy than others, and sometimes everything within is assumed to trust each other. Similarly, ZT requires letting go of the idea that the organization can implicitly trust any device, entity, or service based on who owns it.

Compounding the difficulty, there is no "Zero Trust in a box" product out there. In addition to requiring a shift in thinking, policy, and practice, implementing zero trust requires solutions in all IT domains: infrastructure, applications, data, devices, and identity. For most organizations, some existing solutions will carry through, others will need to be replaced, and gaps filled with new solutions.

With such deep-seated attitudes working against it, with security and network teams already overstretched but no simple "buy and deploy" solution available, and with all of IT as the field of action, it is no wonder that accomplishing a shift to zero trust is difficult. The problem is less what to do than where to start.

Step One: The Radical Rethink—Start with a Zero Trust Architecture

Nemertes' *Secure Cloud Access and Policy Enforcement 2020-2021* research study assessed security success based on the ratio of significant security incidents experienced per security incident (SIPI). Based on the SIPI metric, the most successful organizations are 137% more likely to have adopted a zero-trust approach to security than everyone else. Universally, these organizations devote the first phase of their ZT initiative to developing a zero-trust architecture. By contrast, the other organizations mix all levels of architecture together—ZT, network, security, etc.—and at the same time begin implementation work such as selecting tools and rewriting applications.

The organizations most successful in cybersecurity are 137% more likely to have adopted a zero-trust approach to security than everyone else.

Standing on the Shoulders of Giants: Begin with the NIST Architecture

The National Institute of Standards and Technology—NIST—creates and maintains several gold-standard security- and compliance-related frameworks that range from high-level organization of cybersecurity efforts to highly granular recommendations for specific controls. NIST supplies a now-definitive zero trust architecture; NIST standards are at the base of the federal government's push to have all agencies implement ZT architectures by September 2024.

In 2021, NIST finalized a zero-trust architecture, which should serve as the starting point for every enterprise new to the space. At the highest level, it specifies that for each domain of security—infrastructure, application, data, and device—there be four kinds of tools in place to manage security in every interaction in the environment:

- A policy engine to define and store access policies
- One or more policy decision points to decide in light of those policies whether any given interaction in the environment is allowed
- One or more policy enforcement points that act on the decisions made
- One or more behavioral monitoring tools that provide information on current behavior in the environment to the policy engine

It also requires there be an identity and access management system (IAMS) in place so there can be consistent means of verifying the identity of an entity in the environment, or trying to get into it.

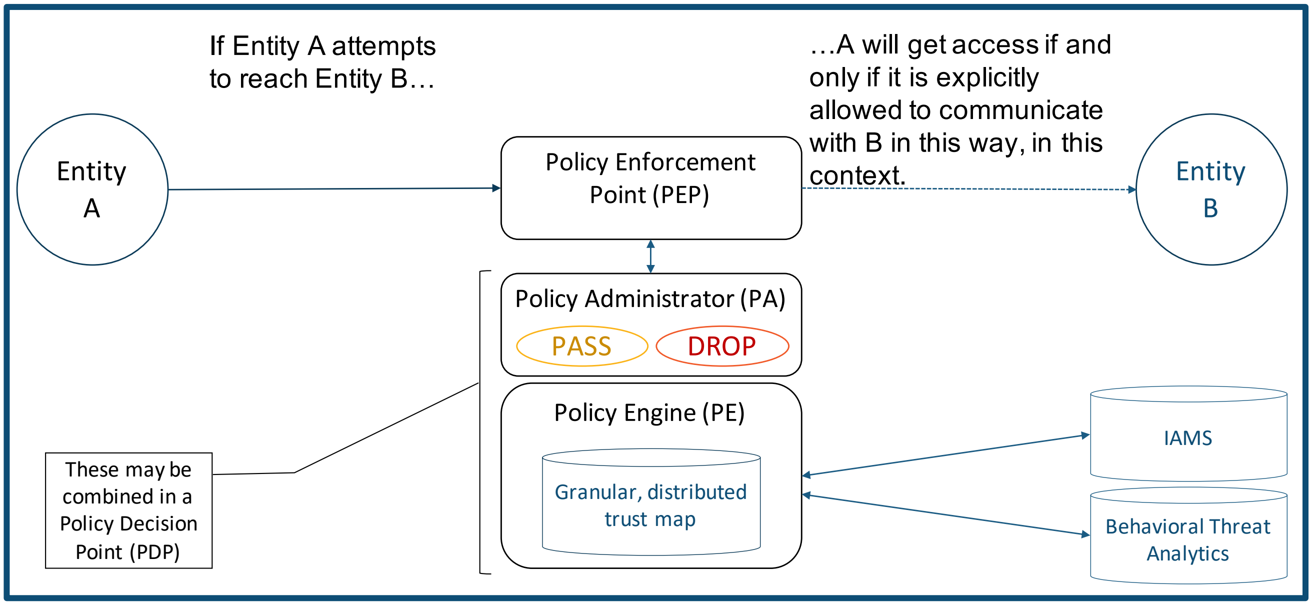
If Entity A attempts to reach Entity B…
…A will get access if and only if it is explicitly allowed to communicate with B in this way, in this context.
Entity A
Policy Enforcement Point (PEP)
Entity B
Policy Administrator (PA)
PASS
DROP
Policy Engine (PE)
Granular, distributed trust map
IAMS
Behavioral Threat Analytics
These may be combined in a Policy Decision Point (PDP)

Figure 1: Zero Trust Architecture, High Level

There is no expectation that the same tools will decide and enforce policy in multiple domains, nor even that there be a single policy engine or policy map. Having a single policy engine would be ideal, though, as it would minimize the number of integrations the enterprise would have to create and maintain, or the amount of redundant work the enterprise would have to perform, to keep multiple policy engines in sync, and eliminate any related latency in achieving that synchronization.

Infrastructure, Applications, Data, Devices

ZT is a fundamental rethinking of access to all enterprise resources. As such, it is critical to recognize that ZT thinking must come to pervade all aspects of IT operations, architecture, and technology. While there is only one comprehensive ZT architecture, people find it easier to comprehend and act on slices of it specific to each domain of interest. Also, working with different facets of the architecture as distinct areas of focus allows easier management of the different paces of implementation in each.

> ***Working with different facets of the architecture as distinct areas of focus makes it easier to manage the different paces of implementation in each.***

Therefore, most enterprises will develop intersecting/overlapping sub-architectures for

- ZT in the physical and virtual infrastructure—initially the network, and especially the network security infrastructure, for most organizations
- ZT in the systems and applications running and communicating on top of that *or other entities'* infrastructure (e.g. on some cloud service providers' infrastructure)
- ZT for the enterprise data resident in all such systems
- ZT for devices trying to gain access to or through any such resources, and for access to the devices as well (i.e. not only can something on device A access service B, but can service B send data to device A)

Step Two: Rearchitecting Networks and Security

Shifting to zero trust network access is one of the most profound transformations facing many organizations because it runs counter to so many habits of mind and requires modifying or reversing decades of practice. By way of contrast, most organizations are already used to thinking about what access rights a user profile in a given enterprise application will have, to a high degree of granularity, but they have no similar habits of thought regarding what a given network node should be able to see, talk to, or listen to. But, rearchitecting their networks and network security also has enormous benefits in terms of risk mitigation, both in the long term—by reducing overall vulnerability to breaches—and in the short term—by improving the methods available to stop or limit the reach of a breach in progress.

> ***Most organizations are***
>
> ***used to thinking about what application access a user profile should have, but not what any given network node should be able to see or talk to.***

So, once they have a high-level ZT architecture, most organizations focus on rearchitecting their network and security infrastructures to fit into it. They can pursue one or more options, ranging from client-based software-defined perimeters (SDPs) to distributed firewalls and access gateways to microsegmentation within a software-defined network. Pursuing multiple paths cannot be allowed to distract from the need to unify policy, though, or the goal of having a single policy engine and trust map guiding however many separate enforcement systems are intended.

Not One Tool, Many Tools Working as One

There is no silver bullet product or service, nor even a coherent, integrated suite of products or services, that delivers “zero trust in a box.” ZT touches all the main domains of enterprise IT activity—cybersecurity, identity, endpoints, data, applications, and infrastructure—and spans all environments, whether on-premises and all clouds. Consequently, IT will require many tools drawn from across the ever-expanding alphabet soup of security product spaces to achieve the ZT vision. (Please see the Zero Trust capabilities checklist at the end of the paper.)

No single tool or suite of tools will meet all zero-trust needs, but IT can pursue a strategy of buying integrated suites or service portfolios that cover large portions of the puzzle.

Automation, orchestration, and centralization of policy and logging are key enablers for zero-trust, and therefore goals to pursue with every tool and service acquisition. With that in mind, the enterprise will need to consider (among other things) how best to

- **Control access to the network**, both in terms of initial admission (*a la* traditional network access control or NAC) and in terms of limiting ongoing access to well-behaved nodes, based on behavioral threat analytics (BTA, aka UEBA, aka XDR)
- **Secure access to cloud resources**, as with a CASB, and access among cloud resources, which might also involve firewalls, access gateways, or other technologies
- **Secure access to and among on-premises resources**, as with an SDP or microsegmentation solution, or some combination of firewalls and access gateways, or other technologies
- **Secure endpoints** through some combination of endpoint protection products, which should must protect the endpoint as such (prevent malware compromises, for example), feed data into the behavioral threat analytics/XDR solution; an ideal tool might also, EDR-style, be able to serve as a broader policy enforcement point (e.g. by blocking a user on the endpoint from connecting to services for which they are not authorized).
- **Secure access to applications based on identity**, using traditional IAM and in-application access management, CASBs, and possibly combined network-level enforcement of application access policies via SDP or other techniques

Though no single tool or suite is going to meet all ZT needs, IT can pursue a strategy of buying integrated suites or service portfolios that cover large portions of the puzzle, and then work to integrate those suites with each other—to aim for a dozen integration points rather than a hundred, say. All such integrations among suites and point solutions should be built around whatever standards exist in the space, and any solution IT considers should have a list of other solutions for which integration is ready to roll.

With any such suite, IT should evaluate how integrated it really is: Does it offer a unified management experience, as it should, or are there different management tools for different functional areas? As importantly, is management in each functional area based on common entity definitions, and run from common policy bases? Dangerous misunderstandings can arise if “source entity” means one thing in a

A suite should be truly integrated from the UI down.

cloud-based firewall policy but something subtly different in secure web gateway policy within the same suite. And is the integrated whole built on a modern base, such as a software defined network or a microservices architecture? Since most security suites grow by acquisition, even those that manage a common vocabulary, policy environment, and management interface are often, behind the scenes, built on a variety of legacy code bases. Wherever that is true, it should be the provider's job, not IT's, to deal with that complexity. The provider should deliver simplicity to IT, whatever the underlying complexities.

Step 3: Starting Points for Operationalizing Zero Trust

Thanks to the pandemic and its legacy of increased hybrid and remote work, connecting remote users and devices to enterprise services is a high profile, high reward starting point for Zero Trust. Consequently, most enterprises begin implementation with an SDP or a cloud-based Zero Trust Network Access (ZTNA) service to control access to on-premises resources, in conjunction with a CASB to provider finer-grain control of access to cloud services (and sometimes on-premise ones).

Most organizations will carry forward some solutions they have, replace others, and add new ones to fill gaps. Their goals in deciding which to replace or add should be to achieve maximum advancement towards ZT with minimum impact on the cybersecurity and network teams, especially in the SOC and NOC, and maximum ability to integrate with each other and with the tools retained. Every new tool or service should be able to integrate with the incident and event management infrastructure and BTA/XDR systems.

However, it is crucial to remember that ZT is not just a bunch of security products—it is an approach to using them. Consequently, operationalizing zero trust also requires acquiring ZT-conversant staff and/or retraining existing staff, as well as rewriting policy and re-engineering processes. Whichever technology space the implementation starts with, relevant policy and process revisions must proceed with it, and staff "upskilling" or hiring should focus there.

DIY vs Managed Services

That need for new staff or staff with newer skill sets can be a major challenge. The existing and increasing complexity of the security environment is running up against a serious shortage in the marketplace of available, skilled staff. Consequently, costs are going up.

This kind of situation always increases the attraction of using a managed security services provider instead of hiring in-house. Offloading the problems of hiring and retaining skilled staff can free up enterprise staff to focus on matters of architecture and policy instead.

Should IT commit to training up a bunch of folks to be aircraft maintenance staff and then replace all the parts of the airplane while flying in it, or hire an aircraft company to keep it flying?

This logic is even stronger when the MSSP is providing not just staffing but also major pieces of a security portfolio. In a variation on the old adage, IT needs to decide if it wants to train up a bunch of folks to be aircraft maintenance staff and then replace all the parts of the airplane while flying in it, or hire the plane manufacturer to keep it flying. Making the people providing some significant chunk of the enterprise's security portfolio responsible for integrating all the pieces and also integrating it with other adjacent tools or suites takes a huge load off (probably already deeply overloaded) internal staff, and can bring IT closer to the goal of unified policy environments with less effort. The benefit runs even deeper for organizations not yet managing a SOC of their own, or straining to accommodate the costs and difficulties of staffing it properly. Engaging SOC as a service in support of Zero Trust pushes that set of challenges off to the provider as well.

Conclusion and Recommendations

To move their organization to ZT, cybersecurity professionals should:

- Develop a high-level zero-trust architecture building on existing standards such as NIST's
- Drill in on architectures in the pillars of ZT: identity, endpoint, infrastructure, applications, and data
- Select a starting point for implementation, with remote network access as a solid default, and the understanding that it is just a starting point—other efforts should pick up in parallel to grapple with
    - Secure access to cloud resources
    - Secure access to and among on-premises resources
    - Securing endpoints
    - Secure access to applications
- Evaluate what to add or change in the current environment to achieve ZT network architecture goals
- Assess which solutions already in place can stay, which must be replaced, and what must be added; prioritize the investments
- Acquire new solutions based on their ability to integrate with retained tools and other tools to be acquired, as well as with SOC systems e.g. for logging and ticketing systems
- Prioritize tools that cover multiple functional spaces, where they are themselves built on modern bases (cloud-friendly and containerized) and embody true integration from management down to infrastructure, and where they can reduce costs and complexity
- Evaluate managed solutions able to accelerate progress, or reduce risks and costs

About Nemertes: Nemertes is a global research-based advisory and consulting firm that analyzes the business value of emerging technologies. Since 2002, we have provided strategic recommendations based on data-backed operational and business metrics to help enterprise organizations deliver successful technology transformation to employees and customers. Simply put: Nemertes' better data helps clients make better decisions.

Zero-Trust Checklist

A checklist of points to consider in evaluating ZT solutions, not in priority order as each organization's priorities will depend on its current state, existing tool set and capabilities, and expected future needs.

- **ZTNA**: Zero-trust network access applies the principles of ZT to the task of connecting entities in the network to each other. Most organizations first focus on connecting users and devices to on-premises or cloud services, to replace a legacy VPN solution
- **SDP**: A software-defined perimeter solution uses a combination of endpoint clients and access gateways to enforce a fine-grain access policy at the network level. If A is not supposed to talk to B, A will not be able to send packets to B. SDP addresses ZTNA at all levels, including deep segmentation as in a data center or IaaS environment.
- **Network segmentation**: Beyond access to specific services, IT needs the ability to segment network traffic with a high degree of granularity, from the data center core and the LAN to the WAN to the virtual networks in cloud environments
- **CASB**: Cloud access security brokers secure and monitor use of cloud-based solutions, usually SaaS applications. They either block or allow user connections to cloud services, and can provide some level of visibility into that use. CASBs can sit in-line between users and applications, or use APIs to integrate into the solutions and control and monitor access.
- **SASE: "**Secure access service edge" is a catch-all classification for solutions or suites of products that address some or all of the functions of a ZTNA, a CASB, an SD-WAN, an SWG, and a cloud-based firewall.
- **BTA/XDR**: Behavioral threat analytics or extended detection and response solutions are essential to a ZT environment because they close the dynamic trust loop—they provide allow adjustments to the ZT trust map based on actual network behavior after initial admission to the network. Sometimes known as UEBAs (for user and entity behavioral analysis).
- **Identity and access management**: IAM is foundational to ZT. Robust IAM with multifactor authentication, either fully consolidated (one source of identity for all users in all contexts) or deeply integrated at a process level, is crucial to making ZT actually work.
- **Granular trust maps**: IT, application owners, and cybersecurity professionals need a mechanism by which to provide and maintain the detailed and specific application access rights of authorized entities (e.g. users) reaching for other authorized entities (e.g. applications running in IaaS).
- **Application use tracking**. Ideally, the solution should provide for more detailed application usage information tracking than network-level monitoring, for both on-premise and cloud systems. This may be part of a CASB, for some applications.
- **Integration with ancillary security solutions**, especially for outbound and endpoint security. For example, a ZT architecture should include/provide, or integrate at policy and logging levels, with
  - **SWG:** Secure web gateways provide protection from malware for out-bound web sessions, and ensure enforcement of corporate web access policies. Adjacent to ZT but not definitional.
  - **DLP:** Data loss prevention tools control and monitor the transfer of enterprise data, an essential aspect of securing the enterprise but not part of ZT itself.
  - **EDR/EPP**: Endpoint detection and response tools expand the ability of endpoints to contribute behavior data to a BTA/XDR and can be policy enforcement points as well. EDR tools should also be, or integrate with, traditional endpoint protection products (anti-malware, etc).
- **NGFW**: Next generation firewalls combine traditional stateful firewall functionality with deep packet inspection, application firewalls, intrusion prevention and detection systems (IDS/IPS) and other capabilities. A complete and seamless ZT environment can in theory get by without one, but they are essential during the transition and useful long-term as "pre-filters" to reduce the amount of traffic at other enforcement points.