The Case for Zero Trust in Operational Technology (OT) Networks

OT Evolution

Operational technology (OT) is a critical component of industrial networks that involves hardware and software used for monitoring and controlling industrial equipment, processes, and events. It is commonly found in industries such as manufacturing, transportation, and energy where recent changes have outpaced traditional networking and security methods.

Historically, OT systems were managed by a single entity and isolated from other networks. However, the trend towards decentralization and digitalization now requires the use of IoT sensors and edge devices that communicate with cloud, corporate, and third-party applications to enable advanced capabilities like machine learning. While this shift is driven by cost efficiencies and the need to stay competitive, it can also put a strain on networking and cybersecurity staff.

Humans are not machines

Network administrators often try to use IT solutions, such as Software-Defined Networking (SDN) and VPNs, to address challenges in the field of OT. However, this approach can be problematic because IT infrastructure is designed to connect people to software applications, not industrial machines. Using IT solutions in the OT world can lead to complex and expensive solutions that don't effectively address security concerns.

Virtual Private Networks (VPN)

VPN tunnels are virtual paths used to send data securely across untrusted networks. There are three main concepts that make VPNs work:

- Tunneling: is the process of sending a packet of one protocol using the infrastructure of another protocol, hiding the delivery information of the original packet.
- Encapsulation: tunnelling is done by encapsulating the packet with an extra header so that it can be correctly sent through the network.
- Encryption: is used to protect the data being sent from being accessed by unauthorized parties.

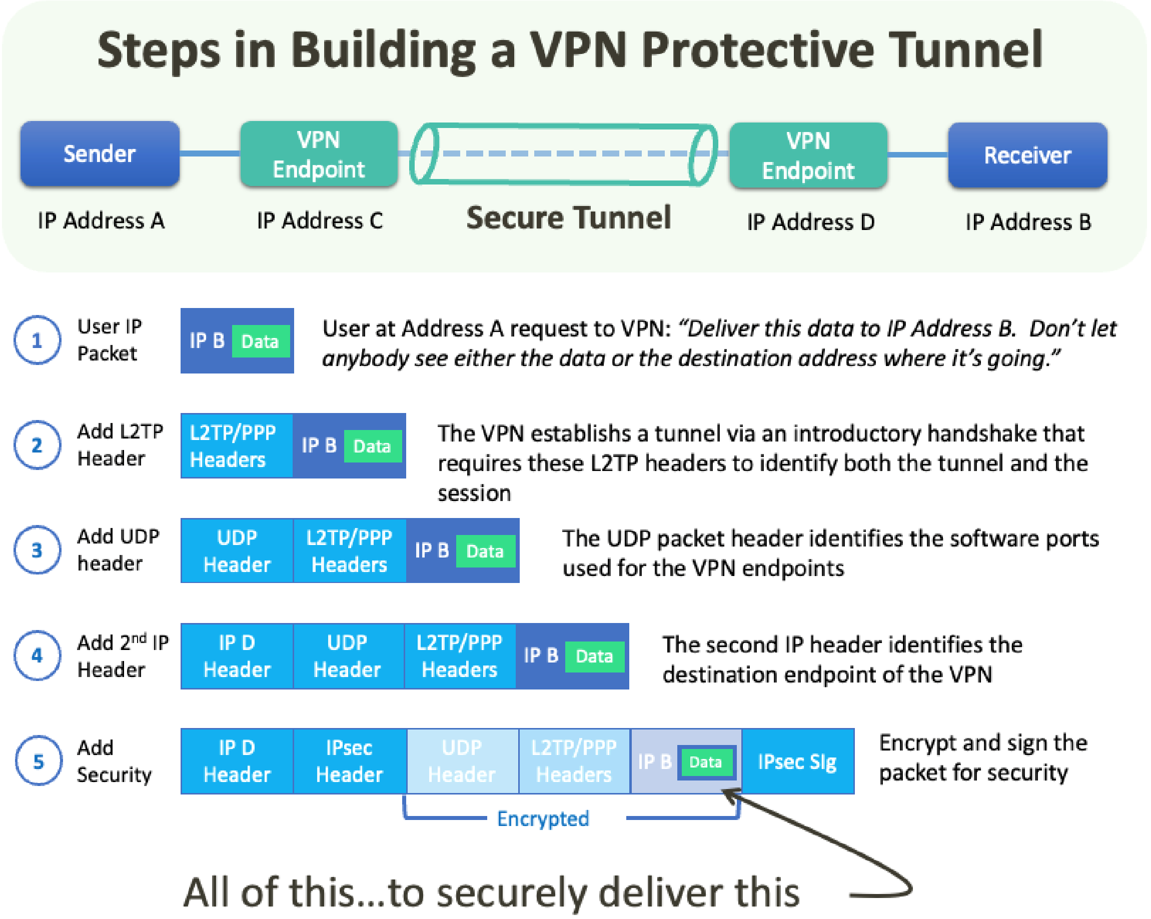
There are different types of tunneling protocols used by VPNs. The three most common ones are:

- Point-to-Point Tunneling Protocol (PPTP)
- Layer-2 Tunneling Protocol (L2TP), which adds IPsec security to PPTP
- OpenVPN, which is based on SSL/TLS security protocols

PPTP has major limitations and works only on IP networks. OpenVPN requires a generic OS server to be exposed to the endpoints and is less often seen in industrial applications. Thus, we will focus on the L2TP/IPsec protocols often used for OT networks.

L2TP is more flexible and secure than many other tunneling protocols because it uses the advanced security and authentication services provided by IPsec (Internet Protocol Security). This gives it robust capabilities for protecting information. However, as shown in Figure 1, L2TP is an IP encapsulation protocol, a complex set of sequential processes which each require user configuration.

Figure 1: Steps in Building a VPN Protective Tunnel



User at Address A request to VPN: *"Deliver this data to IP Address B. Don't let anybody see either the data or the destination address where it's going."*

The VPN establishs a tunnel via an introductory handshake that requires these L2TP headers to identify both the tunnel and the session

The UDP packet header identifies the software ports used for the VPN endpoints

The second IP header identifies the destination endpoint of the VPN

Encrypt and sign the packet for security

All of this…to securely deliver this

IPsec is widely used as a standard for securing communication between networks, as opposed to point-to-point protocols which focus more on securing end-to-end connections. It uses encryption and digital certificates for secure connectivity and can be scaled by adding VPN gateways to new sites. However, when applied to critical OT networks, IPsec VPNs have serious limitations:

- Security risks: VPNs can give third-party vendors full access to a network.
- Revoking access: Once work is completed the revocation of access is often overlooked.
- Ransomware: VPNs can put users' devices on protected OT networks, making them vulnerable to ransomware attacks that can spread throughout the network.
- Supply chain vulnerabilities: VPNs have been known to have severe vulnerabilities, making them a target for hackers.
- Un-patchable systems: Many OT systems use older software that cannot be easily patched, leaving them vulnerable to attacks.
- Lack of accountability and logging: VPNs often provide limited records of third-party vendor activity, making it difficult to track and identify the source of an issue if a breach or mistake occurs.
- Bidirectional access: VPNs create two-way tunnels between networks, but inbound traffic flows can be a source of malicious activity. IP multicast in OT systems can be used to discover all devices and to control them.

- Bridging IT and OT: In some cases, VPNs can connect to a jump box on the IT network, which can bridge the OT network to a third-party, defeating the purpose of separating the two networks.
- Complexity and high support costs: VPNs can be difficult to set up and maintain, increasing the risk of exploitation if not managed properly.

SDN and SD-WAN

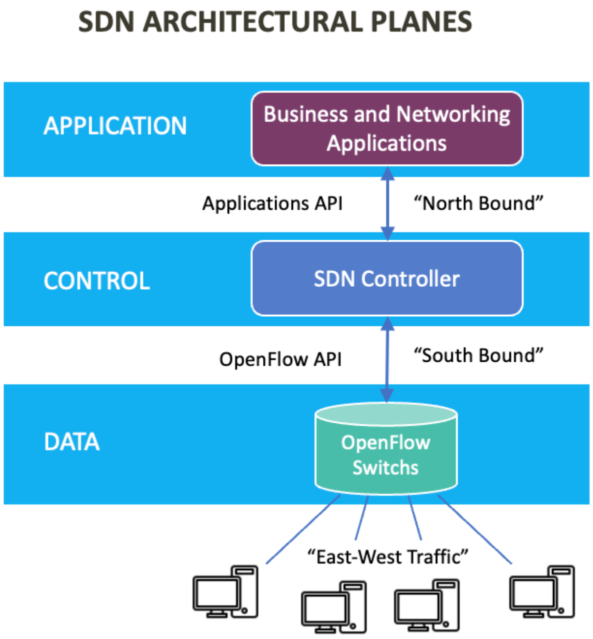
SDN and Software Defined Wide Area Networks (SD-WAN) are ways to manage and control the flow of data through a network using software. Instead of managing each network component separately, operators use APIs (Application Programming Interfaces) to control the network centrally and customize its performance according to the organization's needs.

These virtualization-based solutions are flexible, adaptable, and robust, but can also be complex to set up and maintain, especially in rapidly changing environments like OT networks. There are many similarities between SDN and SD-WAN with the major differences being that SD-WAN focuses on geographically distributed locations and tends to be preprogrammed and less complex. In our security discussion here, we will standardize on the term SDN, while most of the concepts apply also to SD-WAN.

As shown in Figure 2, the principal concept of an SDN architecture is the separation of the control, data, and application planes, resulting in dynamic flexibility and control over how data frames are transmitted through the network.

Each network switch must be adopted by the controller software and flow rules must be applied to each physical port for each logical network flow. This is done in a separate control plane.

Figure 2: SDN Architectural Planes



OpenFlow technology is most commonly used in SDN. It replaces the hardware pipeline in traditional networks with software pipelines based on flow tables. The SDN controller can install rules in the data plane to control how applications and services behave in the SDN fabric:

- The application plane controls the behavior of the network. It is responsible for implementing the network's policies and managing the flow of data through the network. These applications perform a wide range of tasks, including traffic engineering, security, and monitoring.
- The control plane is separate from the underlying physical infrastructure and uses a centralized controller to make decisions and distribute instructions to the actual network devices. The control plane is responsible for managing the flow of data through the network, configuring network devices, and enforcing policies. It also provides an abstraction layer that allows for easy management and automation of the network, enabling network administrators to make changes quickly and easily.
- The data plane is responsible for forwarding and processing packets; making the actual decisions that control the flow of data through the network. The data plane is implemented using

specialized hardware, such as switches and routers, and is controlled by a central controller that communicates with the data plane devices to program their forwarding tables and to configure their behavior.

SDNs offers several benefits, such as deny-by-default protection, easier troubleshooting, network topology visualization, and the ability to change network flow rules in real-time with minimal impact on existing network flows. To achieve these benefits the flow rules must be established securely and be trusted.

However, SDNs have numerous cybersecurity concerns, including:

- Complex configuration: Setting up an SDN can be difficult and requires a detailed analysis of data flow and configuration for each device.
- High initial costs: Setting up the physical network topology, pairing the SDN controller with the switches, and configuring allowed network flows can be expensive and limit the number of devices that can be serviced.
- Security challenges: Ensuring a secure SDN environment is difficult and requires specialized knowledge to properly import and export certificates into switches and controllers.
- Centralized control plane vulnerability: As the control plane of an SDN network is centralized, a compromise of this plane can have a significant impact on the entire network.
- Network segmentation: SDN allows for easy segmentation of the network, but if done improperly, it can lead to security holes. Flows and flow rules can be confusing, making it difficult to understand communication patterns.
- Third-party software vulnerabilities: SDN controllers and applications are often built on third-party software, which can introduce vulnerabilities if not properly secured. OT-focused SDN is a rapidly advancing technology, which may lead to software incompatibilities in delivered equipment. In some cases, SDN software may experience instability and can lead to unexpected behavior.
- Interoperability: SDN controllers and switches from different vendors may not be fully interoperable, which can lead to security issues.
- Insufficient security monitoring: SDN deployments may not have sufficient security monitoring in place, which can make it difficult to detect and respond to security incidents.
- Scaling limitations: As the network size increases, latency may increase, and network situational awareness may be reduced. Additionally, low levels of granularity may make the network less secure, while high levels of granularity can be difficult to maintain.
- Limitations of controllers and switches: The number of switches a single controller can effectively manage is limited, and there is a limit to the number of network flow rules that can be installed into a switch.
- Limited wireless support: SDN does not have the capability to support wireless communication effectively. This is due to the fact that the centralized control and simple router designs of SDN do not align well with the distributed routing algorithms and advanced switch designs used in wireless networks. Additionally, the unique characteristics of wireless channels, such as fading and interference, require specialized modules to be supported by the SDN controller in order to manage interference, track node mobility, and discover the network topology.

Meeting the OT Challenge - It's Time for Zero Trust

VPNs and SDN are technologies that help secure networks, but they also have their limitations and can be vulnerable to attack. Zero trust is considered the next generation of cybersecurity because it addresses the fundamental flaw of perimeter security, which assumes that anything accessing resources from inside the secure perimeter can be trusted. Zero trust, on the other hand, is an identity-based security strategy that assumes that no one should be trusted until they have proven their identity through authentication.

This approach is more effective in addressing the threat of internal and external attacks, as well as the growing number of connected devices and access points. Additionally, Zero Trust Architecture (ZTA) is designed to provide security and access controls at the micro-segment level, rather than relying on a single point of protection. This granular approach allows for a more efficient and effective security strategy, which makes it a key component in the US infrastructure cybersecurity plans, as directed by Executive Order 14028.

Here, we will only touch on the main points.

The primary assumptions of ZTA in cybersecurity are:

- The network is always assumed to be hostile, meaning that all incoming and outgoing network traffic is considered untrusted.
- External and internal threats exist on the network at all times, so all devices, users, and network flows must be authenticated and authorized.
- Network locality is not sufficient for deciding trust in a network, as trust must be based on identity and not location.
- Policies must be dynamic and calculated from as many sources of data as possible.

To implement ZTA, the following five steps must be followed:

1. Every entity (person, computing device, or software application) must have a provable identity.
2. Every communication flow must be irrevocably tied to the entity that generated it.
3. There must be an authority that defines a set of rules for who can communicate with whom.
4. The rules must be securely delivered to all parties on the network.
5. The rules must be checked and enforced at every possible node that handles that flow.

While modern computing and cryptographic algorithms address the first requirement of a provable identity, key generation and storage together with certificate distribution must be handled consistently and securely. In addition, the remaining steps of tying that identity to communications flows and enforcing a meaningful set of rules in a practical technology-based solution for OT networks, remain a challenge. Most IT-developed approaches to ZTA fall short in addressing these challenges.

A ZTA aims to "minimize the blast radius" in the case of a cyber intrusion[1] by implementing strict security measures. Traditional methods, such as network segmentation and multi-factor authentication, can be difficult to implement and may not provide complete protection.

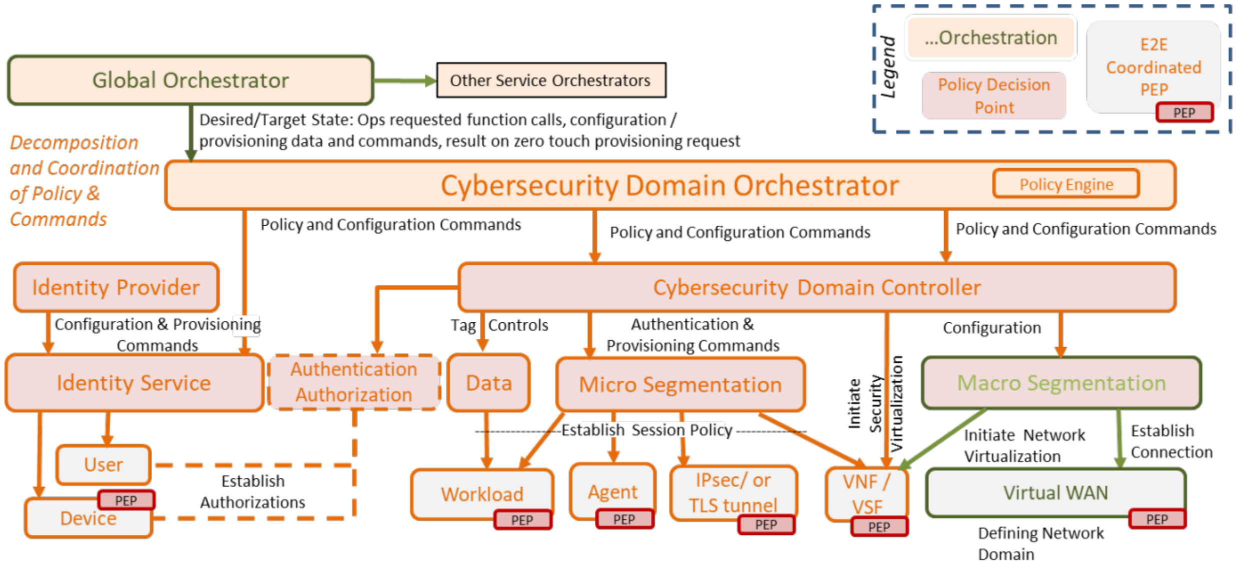
The process of making sure a system is cybersecure involves not only buying a secure product, but also configuring it properly. This can be challenging for industrial customers who may not have expertise in cybersecurity nor the funding to develop it. To help with this, the US Department of Defense created guidelines[2] for configuring a secure system using a combination of SDN and VPN tools, visualized in Figure

[1] https://learn.microsoft.com/en-us/security/zero-trust/zero-trust-overview

[2] https://dodcio.defense.gov/Portals/0/Documents/Library/(U)ZT_RA_v2.0(U)_Sep22.pdf

3. However, the process is still clearly overly complex, inviting errors often leading to system security vulnerabilities.

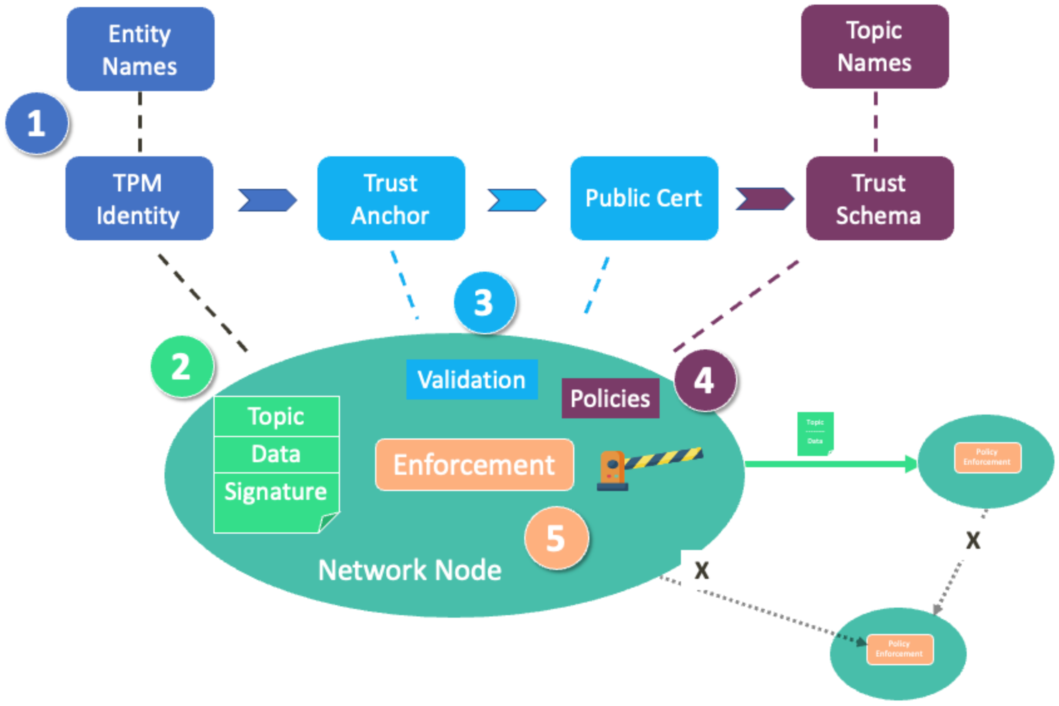
Figure 3: Centralized Orchestration & Policy Management (OV-2)


Legend
...Orchestration
Policy Decision Point
E2E Coordinated PEP
PEP
Global Orchestrator
Other Service Orchestrators
Desired/Target State: Ops requested function calls, configuration / provisioning data and commands, result on zero touch provisioning request
Decomposition and Coordination of Policy & Commands
Cybersecurity Domain Orchestrator
Policy Engine
Policy and Configuration Commands
Policy and Configuration Commands
Policy and Configuration Commands
Identity Provider
Cybersecurity Domain Controller
Configuration & Provisioning Commands
Tag
Controls
Authentication & Provisioning Commands
Configuration
Identity Service
Authentication Authorization
Data
Micro Segmentation
Initiate Security Virtualization
Macro Segmentation
Establish Session Policy
Initiate Network Virtualization
Establish Connection
User
Establish Authorizations
Workload
Agent
IPsec/ or TLS tunnel
VNF / VSF
Virtual WAN
PEP
Device
Defining Network Domain

Zero Trust developed for OT networks

While OT networks are complex, securing them does not have to be. Operant uses a new approach called Named Data Networking (NDN) to encrypt, authenticate, and validate every data packet. This ensures comprehensive security for all data transmissions, regardless of network structure and addresses the five critical steps of Zero-Trust visualized in Figure 4.

Figure 4: Five Critical Steps of Zero-Trust



In the Operant solution, every network entity has a simple lightweight transport agent that uniformly provides a peer-to-peer communications network that enforces zero trust on each and every network packet. Instead of configuring complex and abstract systems, the customer can develop network identities and trust policies using the names and identities they are familiar with from their business context.

Step 1 – Fundamental and Secure Identities

In NDN, trust is not based on the location of network entities (such as machines, software applications, or people). Instead, all NDN packets must be signed by a unique identity, which is verified through an NDN certificate that links a name to a cryptographic key pair. These certified identities are persistent and must be protected by secure storage of the private key. To achieve this, Operant uses three methods: 1) Hardware Trusted Platform Modules (TPMs) for field gateways, 2) Commercial security applications like Azure Key Vault for servers connected to the cloud, and 3) Integration with identity management services such as Active Directory for individual user keys and certificates.

In contrast to NDN, the use of TPMs and key vaults in commercial VPN products is not a standard practice and relies heavily on proper process management, which is vulnerable to human error. As a result, private keys are often stored on local disk and can be accessed by hackers. This has led to several known cases of VPN breaches, such as the NordVPN attack in 2019. While industrial VPNs have often improved their security measures, such as more secure storage of private keys and frequent key changes, NDN has built-in security features based on identity verification, which are embedded in the core of the protocol.

A public key must be authenticated by submitting it as part of a certificate signing request (CSR) to a local certificate authority (CA). This CA uses its own private key to sign the request, which confirms that the entity associated with the key pair is authorized to be a member of the network. These signed public key certificates are then used to validate the authenticity of data packet signatures, as they are authorized by the trusted CA. This local CA is known as the "trust anchor" and its certificate is stored securely in each network entity, such as in the TPM to prevent tampering or substitution. This approach of using a local CA is done to avoid common social engineering compromises, such as altering a DNS CA URL which would lead to a rogue CA authenticating improper access.

Step 2 - Communication Flows Bound to Senders

In the NDN protocol, each data packet carries a signature that is generated using the producer's cryptographic private key at the time of data creation. This binds the packet's name to its content, making the data immutable, meaning that if the producer changes the content of the data packet, it will generate a new packet with a new name to distinguish the different versions of the content. This prevents data from being spoofed or replayed, which can happen with conventional security mechanisms. The signature not only confirms the authenticity of the sender but also ensures the integrity of the data packet, as any changes made to the data during transit will be detected by the receiver.

This feature means that the security of the packet is not affected by the path of delivery. VPNs provide similar security but are limited to providing assurance only between the VPN client and the destination VPN gateway, which may be located on a remote server or with a commercial provider. The client has limited visibility into threats that exist past that gateway portal. NDN solves this problem by effectively seamlessly creating a VPN connection between each and every entity and for every individual data packet. This significantly increases the security of communications, particularly for OT networks with many individual machine entities.

In addition, once a legacy VPN session is started, any number of independent commands can be sent. With NDN, each data packet is treated independently and the access to the data can depend on the *content* of the data packet. For example, one user can be authenticated and allowed to send only Read commands to a controller, while a second entity can be independently authenticated but have access control permissions to both read data and write control commands. All these variations are securely controlled at the NDN network layer.

Step 3 - An Authority That Defines Rules

All cryptographic verifications must terminate at a pre-established trust anchor. In other words, there must eventually be an authority, often termed a trust anchor, that can be trusted as an a priori certainty. In addition to authenticating the identities in Step 1, the CA with the corresponding trust anchor certificate also signs and authenticates the trust rules.

After that trust anchor's public certificate is installed in each entity so that they verify other entity's identities and trust rule signatures by verifying their certificates along the certificate chain to the trust anchor. Trust anchors are usually installed via out-of-band mechanisms, and the development of these supporting mechanisms depends on the trust anchor model in use.

In today's practice, trust anchors are commonly established via the following means:

- Obtaining certificates from commercial certificate authorities (e.g., TLS certificates)
- Installing a single global trust anchor (e.g., DNSSEC)
- Establishing trust in an ad hoc manner (e.g., Trust-On-First-Use, Web-Of-Trust)

There have been recent concerns about commercial CAs as they are an attractive target for advanced cyber-attacks from nation states. One example of this is a recent breach of an Asian CA[3]:

> *"Nation-state hackers based in China recently infected a certificate authority and several government and defense agencies with a potent malware cocktail for burrowing inside a network and stealing sensitive information, researchers said on Tuesday.*
>
> *The successful compromise of the unnamed certificate authority is potentially serious, because these entities are trusted by browsers and operating systems to certify the identities responsible for a particular server or app. In the event the hackers obtained control of the organization's infrastructure, they could use it to digitally sign their malware to make it more easily slip past endpoint protections. They might also be able to cryptographically impersonate trusted websites or intercept encrypted data."*

NDN uses a different approach to trust anchors that is better suited for OT networks. Instead of relying on a single global trust anchor, NDN assumes that there is an authority for each network and that all entities can discover the trust anchor through local system settings. The trust anchor is installed at configuration time and signed for integrity by the entity's private key, which ensures it will not be changed during use. This trust model is similar to the Simple Distributed Security Infrastructure (SDSI/SPKI) in the way trust anchors are established. The trust anchor then validates the identities of each user sending or receiving data, the integrity of the specific data and the detailed trust rules.

Step 4 - Rules Must be Securely Delivered to All Parties:

NDN is a trust management system wherein each sender is responsible for providing all the information needed to prove that a data packet complies with the system's policies. These policies are defined in a construct called a trust schema, which specifies which entities are trusted to perform certain actions and which key should be used for specific data namespaces and purposes. The trust schema certificate is signed by a central authority to prove that these are the defined permissions. The trust schema rules express organizational and operational policies as relationships between the keys of a publication's signing chain, allowing for fine-grained control over what an identity can do, and follow a Zero Trust

[3] https://arstechnica.com/information-technology/2022/11/state-sponsored-hackers-in-china-compromise-certificate-authority/

framework. They enforce standards on message content, adding new capabilities to control the flow of information at the network layer. VPNs and SDNs, in comparison, have less control over the "what and when" of communications. NDN takes these concepts to a higher level of granularity and adds more capabilities to immutably secure the flow of information.

Step 5 - Rules Must be Checked and Enforced at Every Node

The set of trust schema rules is a type of certificate secured via signing with the system trust anchor. Both the trust schema and the trust anchor certificates are stored in each entity for validation. This is done by using trusted enclaves on devices during the enrollment process. This ensures that all entities in the network have a consistent set of trust rules that have been approved by the highest authority, cannot be changed, and are consistently applied at the network layer. This is different from Access Control Lists (ACLs), which are independently configured on each device in the application layer. ACLs are difficult to configure consistently and can be compromised in the application layer. They also typically do not provide enough granularity to authorize specific actions. Additionally, changing system requirements often requires a significant amount of administrative effort to consistently update ACLs throughout the network, and keep track of the changes. While some SDN systems can help manage these configurations, they may not be suitable for critical asset systems that cannot assume cloud connectivity or trust third-party providers.

The trust schema is used to both create and verify messages in a network, ensuring that all parts of the system follow and enforce the same rules, even as those rules change to address new security threats. If an entity attempts to send a message that it is not authorized to perform, the message will be blocked, and the administrator will be alerted. Additionally, even if a rogue actor were to send a false message, the message would still be rejected by the receiving entity, who applies the same trust rules and alerts the authority. This level of security is not provided by other methods of trust control in the application layer. With NDN, this level of security is guaranteed in the network layer with minimal additional complexity for the user. This provides a fully distributed way of enforcing trust policies without relying on a secure physical network or extensive configuration on each device.

Trust rules in a network also help to securely partition data. Each message is constructed such that it must comply with the trust rules in order to be sent. Then, as messages transit to new network segments or sites, they are checked at intermediate points called relays to ensure that they comply with the trust rules for the new area they will enter. Only if the information is validated and approved to progress further on through the new site, will it be passed. Otherwise, it will be blocked and isolated locally where it is needed, per Zero-Trust requirements. Finally, the receiving entity will check the message again and reject any messages that do not comply with the trust schema. All entities and relays in the network are ensured to be following authorized trust rules by the shared trust anchor. NDN greatly extends the concept of existing VPN solutions from only allowing communication between points A and B for any content, to encompassing more meaningful distributed, granular control, and zero trust approaches.

Example - Utility Control of Distributed Energy Resources (DERs)

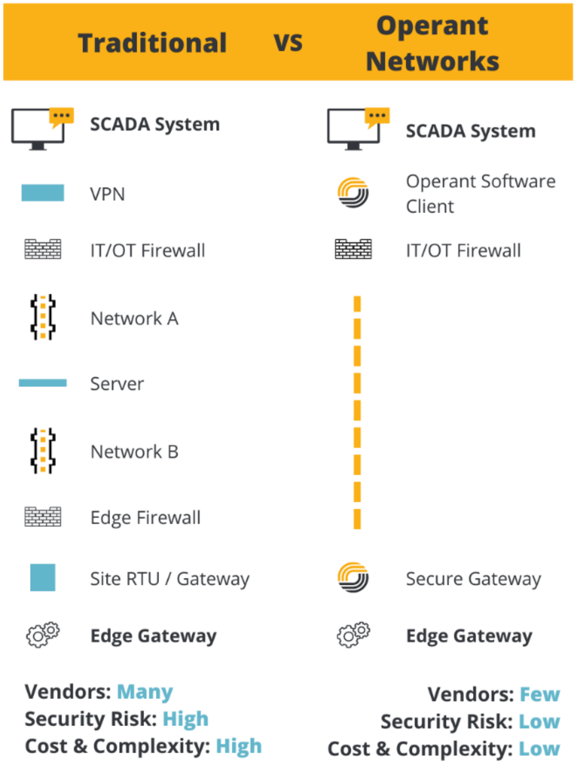
Electrical utility assets have traditionally been secured by proprietary communications links with simple master controller architectures, whose security depends on physical access control and operational obscurity rather than modern cryptography. Utility communications protocols, such as Modbus and DNP3, have limited cybersecurity capabilities and rely on additional standards such as IEC 62351 to add security. This results in patchwork solutions that are difficult to administer and can never be extended to support DERs outside the utilities' direct control.

DER communications, especially those located behind the meter (BTM), are often connected via the public Internet, which provides global connectivity, and is well known to be unsecure. Therefore, industry is moving to address the cybersecurity of DERs via additional internet standards such as IEEE 2030.5. It is widely hoped that integrating these secured internet standards with traditional utility control systems will support future needs. However, these efforts will inevitably be hampered by reliance on the internet's underlying TCP/IP networking protocol, and will quickly reach fundamental limits. VPNs, firewalls, interconnected IT and OT networks all become a burden and drag on the utility's business model.

Operant's NDN solution simplifies the system and provides new capabilities for DER utility integration:

1. Provides end-to-end security over any channel, including the public internet.
2. Includes integrated key distribution, including the ability to issue revocable keys with limited authority or time.
3. Allows multiple trust schema to be defined, each containing numerous trust rules, and securely distribute them to all entities.

Figure 5: Traditional vs Operant Network's NDN Solution



Defining and updating the trust model and its underlying components becomes a manageable task with the NDN solution:

- New entities automatically learn the trust rules as they are commissioned.
- The business needs are met by a system that supports the required relationships.

The implementation method is as follows:

- If computing resources already exist, a simple Operant software client ('OPN Connect') is installed to interface existing applications and assets to the NDN transport.
- At field sites, inexpensive off-the-shelf commodity gateways host the OPN Connect software and provide connections to site equipment.

OPN Connect is a technology that allows for various types of communication between devices. It builds upon standard TCP/IP connections, as well as industry-specific protocols like Modbus and DNP3, and more advanced methods like MQTT publish-subscribe. It can work over a variety of network links including Ethernet LAN, cellular, WiFi, and wireless mesh. It supports multiple connections at once while ensuring secure communication. In simple terms, OPN Connect is a flexible and secure way for devices to talk to each other.

Figure 6: A Commercial Field Gateway

A commercial field gateway is a device that connects distributed sites to control and monitoring services. It allows secure remote access to the distributed sites, which eliminates the risk of using RDP (remote desktop protocol) to troubleshoot problems. Additionally, it provides secure storage for data by encrypting and signing it, so that it is protected while at rest. Overall, the field gateway uses NDN communications, which is a powerful communication protocol, to provide a host of beneficial services for remote distributed sites.

Conclusion

As the number of connected devices deployed into industrial networks grows exponentially, so does the volume and sophistication of targeted cyber-attacks. Hence, the only viable option for companies is to adopt a Zero Trust Architecture. Network administrators should be aware that Zero Trust is not a one-size-fits-all solution, particularly when trying to blanket-deploy technologies developed for human-centric IT networks into machine-centric OT networks. Instead, it is far easier, more secure, and more cost efficient to source solutions, such as Operant's, which have been developed, tested, and recommended specifically for OT use cases.