Foundations of a Zero Trust Implementation

A framework for application developers, site reliability engineers, and network engineers

Cisco Developer

Table of Contents

Introduction

The Need for Zero Trust Security

Software continues to eat the world—and as more and more aspects of business, government, and everyday life become managed by that software, our systems become more complex, more interconnected, and manage more data than ever. While this has been a boon for technology companies, it also comes at a cost. Malicious actors have ever-increasing incentives to attack systems and steal our data. Ransomware attacks are already **up by 13% this year.** And according to the **2022 Cost of a Data Breach** report from IBM, the global average total cost of a data breach is $4.35 million and growing.

Security, now more than ever, is a top-most concern.

In this guide, we'll look at a major effort to protect these systems—zero trust security. We'll look at exactly what zero trust security is, how it can help your organization, and the best practices for implementation. We'll cover trending topics such as zero trust in the cloud, zero trust with Kubernetes, and more. We'll even cover a few caveats to watch out for.

The SRE's task of maintaining reliable, secure, and resilient operating environments is becoming harder, not easier. But by the end of this guide, you should have the information you need to adopt zero trust within your organization—and be on your way to using it to protect your organization, and users, from increasingly sophisticated attacks.

Let's begin with a definition and core concepts.

Section 01

What is Zero Trust Security?

01

Zero trust security is a paradigm that aims to significantly reduce security risk in large-scale software systems accessed over a network. It can be a nebulous concept, so let's start with a precise definition:

Zero trust is a security approach for networked systems in which every request is validated independently—regardless of attributes such as the source of the request—while giving all actors least-privileged access and working under the assumption that the network might be compromised.

Let's unpack this definition by covering the core concepts.

Never trust, always verify

In zero trust, no request is implicitly trusted. Therefore, every request must be validated independently. When a request arrives, its authenticity is verified based on all available information–such as user identity, location, device health, service or workload, data classification, and anomalies. The caller's identity must be verified based on credentials, and the caller must be authorized to perform that particular request.

Least privilege

The principle of least privilege states that every entity accessing a system should have the minimal set of permissions needed to access the data and actions required to complete their task. In this way, if an entity is compromised, the potential damage is limited.

Assume breach

Finally, in zero trust we assume every access is a breach. While not literally true, in effect we acknowledge that every system, given a determined attacker with appropriate resources, will eventually be breached. Therefore, security teams should invest in measures that contribute to quick detection, minimizing of the blast radius, and clear remediation procedures for every access granted.

In a world where more than 80% of attacks incorporate the use of stolen credentials, and hackers don't even have to break into the systems they exploit, zero trust security is a modern framework that significantly reduces risk.

The implications of a zero trust approach

Practically, the above concepts mean organizations adopting the zero trust model will implement the following high-level steps. (We'll explore in detail later how these steps translate to concrete actions).

01

Establish trust

Verify identity (such as a user accessing an application or a device accessing a network) based on credentials.

02

Enforce trust-based access, using the principle of least privilege

Allow access (with the minimum set of permissions necessary) based on that established trust.

03

Continuously verify trust

Subsequent requests cannot be implicitly trusted. Instead, trust must be reestablished with every new request.

04

Respond to changes in trust

By denying access, asking the user to remediate, or granting additional access.

Now that we've covered the basic definition and principles, let's deepen our understanding by exploring origins.

Section 02

The Origins of Zero Trust Security

To fully understand zero trust security, we should understand its roots. Let's look at the world before zero trust security and how research over the last 25 years has led to the growing adoption of zero trust.

02

The world before zero trust

Before mobile apps and the cloud, the world (from the perspective of networking) was simpler. The network architecture of large-scale enterprise systems could be roughly described with an analogy: castles and moats. The castles were corporate networks, and the moats were the perimeters—such as firewalls that allowed entry through approved IP addresses and VPN connections—that protected those networks. Public access was achieved through a demilitarized zone that was carefully managed.

However, once a user passed the moat and into the internal network, they had full freedom within the castle and could access any other internal system. This approach was acceptable at the time. Employees, by and large, worked in an office with computers connected to the LAN via cables. Physical security was a major component of corporate security. Remotely accessing the internal network—if it was even possible—was cumbersome, and often managed as a special case by the IT department.

Then came the rise of mobile phones, laptops, tablets, and the bring-your-own-device (BYOD) revolution. COVID-19 changed how and where employees worked—in their homes, in a café, or on the road—and the hybrid workforce became the norm. The security paradigms no longer worked; the network security world was ready for a disruptive paradigm shift.

That shift was zero trust security.

The emergence of zero trust: from thesis to standards

“Zero trust” began as a phrase coined in a **doctoral thesis** by **Stephen Marsh** in 1994. By the early 2000s, security researchers and working groups called for the **“de-perimeterization” of corporate networks**. In 2009, Google came out with the **BeyondCorp initiative**, which was the first massive-scale, zero trust architecture. However, even with the work done by Google, zero trust architectures still weren't very common.

Then, in 2018, security researchers from the United States **National Institute of Standards and Technology (NIST)** published an influential paper: **Zero Trust Architecture (SP 800-207).**

Since then, the adoption of zero trust has grown exponentially. Industry analysts have also contributed to zero trust architectures and frameworks, such as Forrester's **Zero Trust eXtended Ecosystem (ZTX)** or Gartner's **Continuous Adaptive Risk and Trust Assessment (CARTA)**. However, the architecture outlined by NIST (SP 800-207) is the most comprehensive and considered best-in-class. We’ll examine it in more detail in **The Architecture that Facilitates Zero Trust section.**

Now let’s look at the current state of the world today regarding zero trust.

Section 03

The State of Zero Trust Security

03

To look at the current state of zero trust security today, and where and how it is being adopted, let's break down the market into three segments with diverse concerns: cloud providers, industry, and government.

Zero trust and the cloud

Because the principles of zero trust are crucial to the proper functioning of the cloud, modern cloud platforms have become the vanguard of zero trust security.

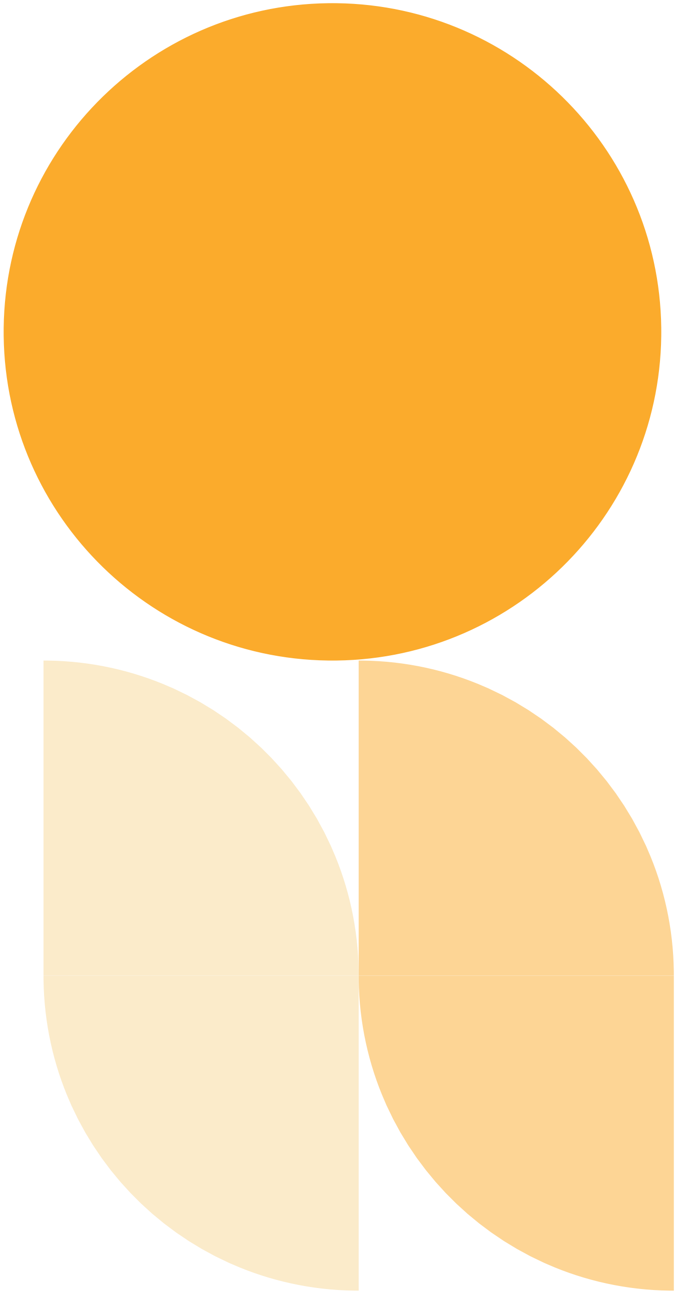
- Cloud providers expose their functionality—and data—through public APIs, which must have robust verification measures in place to prevent abuse.
- Public clouds offer their services, for the most part, via multi-tenancy. When multiple tenants share the same physical machines and hardware, strong isolation is paramount, and cloud providers achieve this isolation through virtual machines (VMs).
- Cloud providers also emphasize the principle of least privilege with a very granular permission model.
- The comprehensive monitoring and audit capabilities of cloud platforms adhere to the “assume breach” tenet, enabling the quick detection and neutralization of attackers and threats.

With cloud providers leading the way in adopting zero trust, it’s no surprise that software and technology industry leaders are following suit.

Zero trust and industry

As organizations shift their systems to the cloud, their approach is naturally aligning with the zero trust security model. Additionally, as organizations typically use Kubernetes to manage their complex systems, a zero trust approach is becoming more of a necessity.

Let’s look at some examples.



The elastic nature of the cloud—in which IP addresses are allocated dynamically—makes the old-world "moats" approach of approving static IPs untenable. As a result, many enterprises have adopted identity and access management (IAM), which has broad availability and support among cloud providers and allows for a straightforward implementation of "never trust, always verify."

In the Kubernetes world, the scale and complexity of systems grow rapidly. Enterprises are building their systems on top of multiple clusters deployed across multiple geographical regions, sometimes across multiple cloud providers or a hybrid of cloud-based and on-prem clusters. The old castles and moats model—with its firewalls and VPNs—doesn't work for this level of complexity and its ever-changing network boundaries. Additionally, the service mesh is a recent phenomenon that supports zero trust security at the micro-service level by enabling mutual TLS and strong policies to curate lateral access within an internal network.

The emerging focus on data protection and data privacy (for example, see GDPR, CCPA, or CTDPA) also makes it clear that zero trust security is much more suitable to address the challenges of protecting organizations from a data breach.

Zero trust and government

The public sector is one of the prime movers and active implementers of the zero trust approach. In early 2022, the White House announced a federal zero trust strategy. The strategy document, **M-22-09**, requires all federal agencies to meet certain zero trust goals by the end of 2024.

This announcement came on the heels of multiple cybersecurity attacks and data breaches—including the **Colonial Pipeline ransomware attack**—that exposed the security flaws of critical infrastructure and highlighted the need for better cybersecurity standards and implementation.

This announcement was significant for several reasons:

01 In the US, there are tens of millions of federal, state, and local government employees, and they use a vast number of information systems. All these systems now must adopt the zero trust security model.

02 All government suppliers will be required to adopt the zero trust security model.

03 Government adoption often leads to and accelerates the adoption of new technologies and standards by enterprise organizations.

Less than a year after NIST published its guidelines on zero trust architecture, the **US Cybersecurity and Infrastructure Security Agency (CISA)** published its **Zero Trust Maturity Model** guidelines. The **Department of Defense (DoD)** also published its **Zero Trust Reference Architecture.**

Other governments across the globe are also very aware of the importance of zero trust. For example, the UK's NCSC published its own set of **zero trust architecture design principles.**

The executive memorandum (M-22-09) laid out goals for achieving security trust which were organized around the zero trust maturity model developed by CISA. CISA's model clearly lays out how an organization can approach and evaluate its gradual implementation of zero trust security. This model is meant to provide a clear path for modernizing systems, towards a goal of zero trust. We can use it to help us understand the key aspects we need to examine in our own systems.

Section 04

The Zero Trust Maturity Model: A Roadmap

04

The Zero Trust Maturity Model from CISA consists of five pillars. These pillars represent five areas in an organization where advancements can be made towards zero trust.

01

Identity: The identity pillar concerns the unique set of attributes associated with an entity (be it a human user or a software component). The set of attributes unambiguously identifies the entity. Authentication establishes the identity of an entity. Authorization verifies that the request sent by the entity is allowed by its set of permissions.

02

Device: The device pillar covers the management of hardware assets on the network. Devices can be servers, personal computers, mobile phones, tablets, sensors, robots, and other IoT appliances. Devices may belong to the enterprise or to human users via BYOD.

03

Network/environment: The network/environment pillar addresses connectivity and the different methods that devices use to communicate with one another. That includes the public internet, internal networks, wireless networks, and more. In this context, the segmentation of networks is very important.

04

Application workload: The application workload pillar addresses the applications running on the devices. Ensuring that applications behave as they are supposed to and are not malicious is a major concern as, by definition, they already run on devices that are part of the zero trust network.

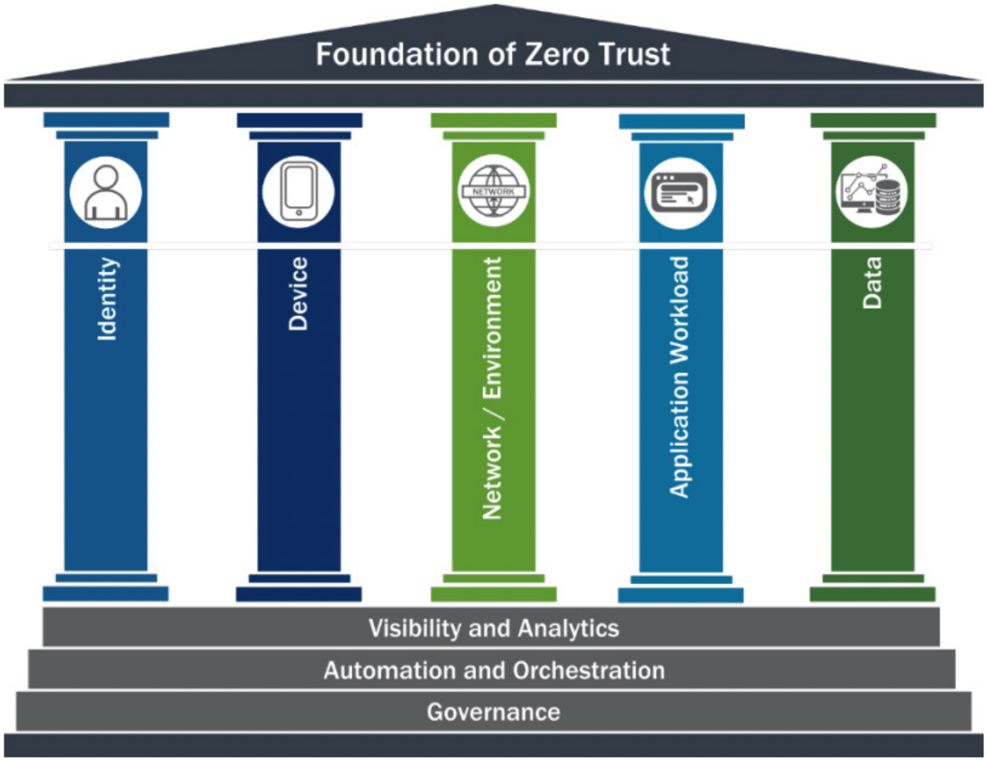
05

Data: The data pillar concerns the management of data that is stored and exchanged by the enterprise and its users. That data is often the organization's most important asset, and protecting it is a high priority for zero trust implementations.

Each pillar requires visibility, analytics, automation, and orchestration in order to be properly implemented into the zero trust model.

The model from CISA acknowledges that implementations will be gradual rather than all or nothing:

As implementers transition towards optimal zero trust implementations, their solutions increase in reliance upon automated processes and systems, more fully integrate across pillars, and become more dynamic in their policy-enforcement decisions.



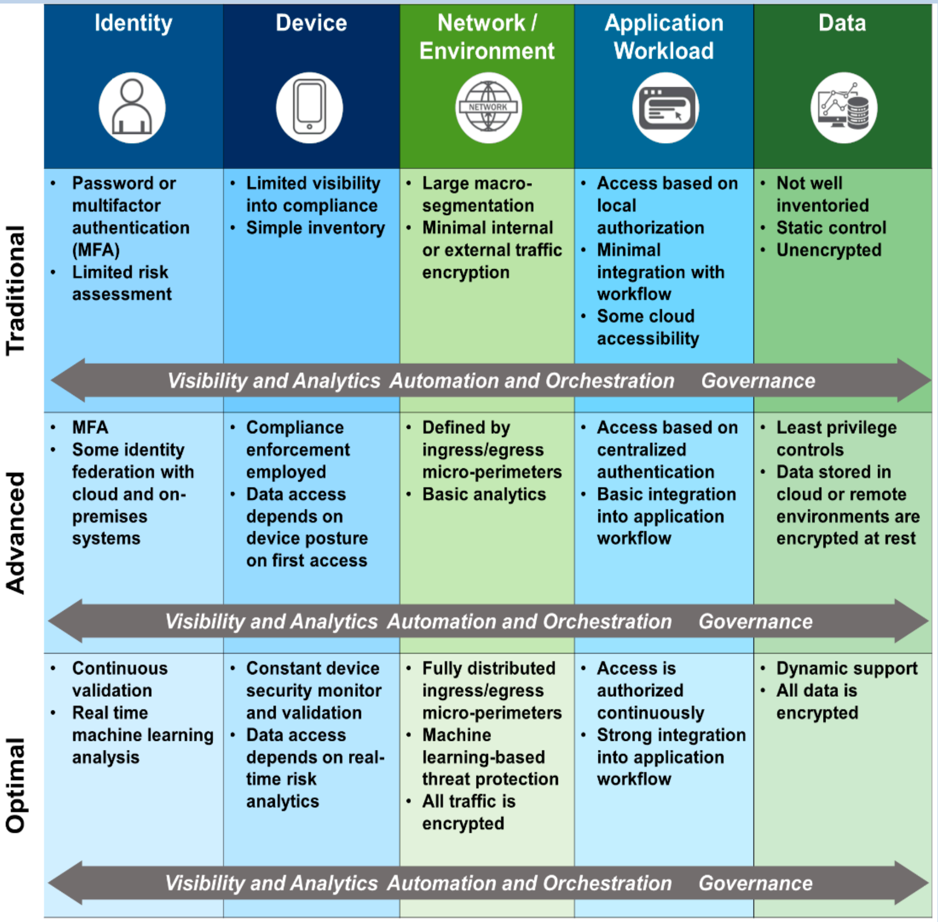
For most organizations, the attainment of full-fledged zero trust will require a significant journey. CISA divides that migration toward optimal zero trust security into three maturity stages:

01 **Traditional**

02 **Advanced**

03 **Optimal**

CISA shows what an enterprise's implementation looks like for each pillar and at each stage. The following rubric helps organizations understand how mature they are in implementing zero trust. For example, a traditional approach to securing the identity pillar is to use passwords or multifactor authentication (MFA). However, to move toward advanced maturity, an organization must enforce MFA for all its users.



	Identity	Device	Network / Environment	Application Workload	Data
Traditional	• Password or multifactor authentication (MFA) • Limited risk assessment	• Limited visibility into compliance • Simple inventory	• Large macro-segmentation • Minimal internal or external traffic encryption	• Access based on local authorization • Minimal integration with workflow • Some cloud accessibility	• Not well inventoried • Static control • Unencrypted
	Visibility and Analytics Automation and Orchestration Governance				
Advanced	• MFA • Some identity federation with cloud and on-premises systems	• Compliance enforcement employed • Data access depends on device posture on first access	• Defined by ingress/egress micro-perimeters • Basic analytics	• Access based on centralized authentication • Basic integration into application workflow	• Least privilege controls • Data stored in cloud or remote environments are encrypted at rest
	Visibility and Analytics Automation and Orchestration Governance				
Optimal	• Continuous validation • Real time machine learning analysis	• Constant device security monitor and validation • Data access depends on real-time risk analytics	• Fully distributed ingress/egress micro-perimeters • Machine learning-based threat protection • All traffic is encrypted	• Access is authorized continuously • Strong integration into application workflow	• Dynamic support • All data is encrypted
	Visibility and Analytics Automation and Orchestration Governance				

Source: CISA

Zero Trust Maturity Model

The goal, of course, is for an organization to move toward “optimal” maturity across all five pillars. In this way, the rubric from CISA serves as an implementation roadmap.

Now that we’ve seen a basic roadmap of how an enterprise can move toward zero trust, let’s walk through the specific architecture that will help an enterprise to get there.

Section 05

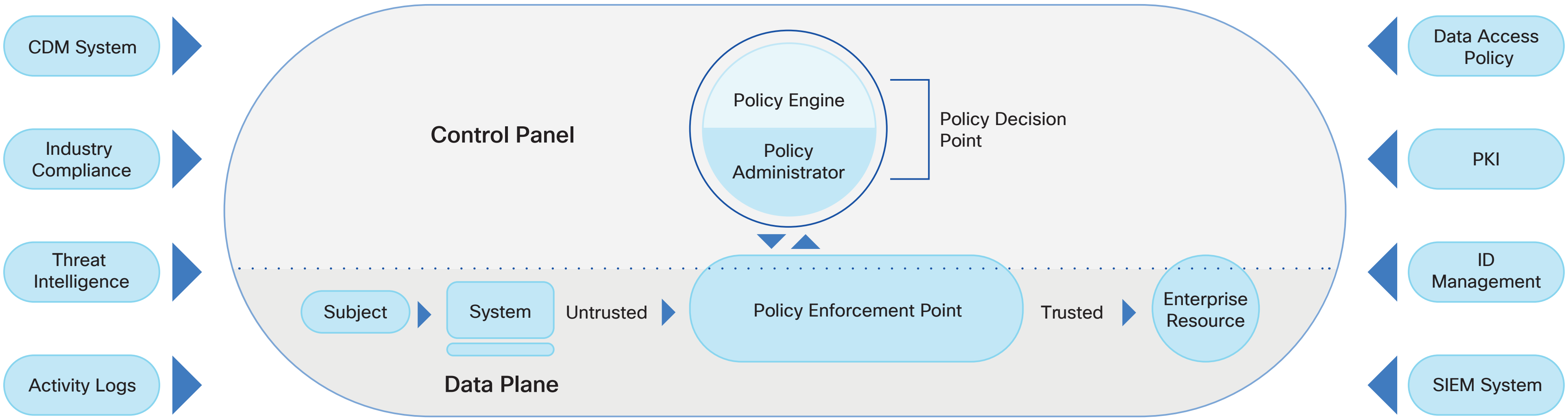
The Architecture that Facilitates Zero Trust

05

Zero Trust Architecture

But what exactly does Zero Trust look like in practice? Let’s look at an example of what the architecture supporting zero trust might look like.

A zero trust architecture is a collection of components and processes that work together to establish the zero trust security model. We’ll examine each of the logical components one at a time.



Policy engine

The policy engine is the component responsible for deciding if a request to access a resource should be granted or denied. The decision will be based on the identity of the actor requesting access and the configured access policies. Information from auxiliary systems can play into the trust algorithm employed by the policy engine.

Policy enforcement point

The policy enforcement point manages the actual connection between an actor and a resource. It enables, monitors, and terminates connections according to the decisions of the policy engine as mediated by the policy administrator.

Industry compliance system

The job of the industry compliance system is to configure compliance policies for the policy engine. Compliance is a key concern of large enterprises in highly regulated industries. The policy engine must be aware of compliance rules and ensure they are followed. A policy engine that grants non-compliant access exposes the enterprise to litigation, reputation damage, and potential fines.

Policy administrator

The policy administrator takes the decision of the policy engine and executes it. Among other actions, that execution may include:

- Establishing or shutting down the communication path between an actor and a resource
- Generating a temporary access token or some other form of credential

If the actor is authenticated and the session is authorized, then the policy administrator will configure the policy enforcement point (see below) accordingly. Note that the policy engine and the policy administrator may be implemented as a single entity, but conceptually these are two separate components.

Continuous diagnostics and mitigation (CDM) system

The CDM system is one of the auxiliary systems that the policy engine may use when rendering policy decisions. The dynamic nature of enterprise assets requires a separate system to keep track of the current state and ensure that the accessed resources are valid. For example, if a critical OS patch has not yet been applied to a resource, then access will be denied even if the actor is authorized to access the resource.

Threat intelligence feeds

New threats are discovered every day, and vulnerable resources should not be accessed. This leads to two implications:

- A malicious actor must not be able to extract data from a compromised resource.
- A legitimate actor must not access a compromised resource because the data might have been corrupted.

Threat intelligence feeds can assist the policy engine in making dynamic decisions regarding vulnerable or at-risk assets.

Network and system activity logs

Logs provide essential input to the trust algorithm. The network and system activity logs of all the components can be aggregated to assess the security state of the overall system, identify abnormal patterns, block suspicious actors, and isolate compromised assets.

Public key infrastructure (PKI)

Public key infrastructure is a key component of zero trust architecture. The PKI is responsible for issuing unforgeable certificates to actors, services, and applications so they can prove their identity to one another, allowing the policy engine to perform its duties confidently in regard to the identity of actors and resources. The PKI is often used by the identity management system.

Identity management system

The identity management system creates, stores, and manages user accounts and service accounts, which may contain additional metadata such as name, email, title, and team.

Security information and event management (SIEM) system

The SIEM system collects security-centric information for later analysis. This data is then used to refine policies, alert administrators to attacks on the system, and inform post-mortems after attacks have been mitigated.

Having covered the technical components that make up a zero trust architecture, let's walk through the steps for implementing zero trust security in your enterprise.

Section 06

Implementing Zero Trust in Your Enterprise

Implementing zero trust in a large enterprise organization can be a complicated process. Let's understand what is involved by following the steps recommended by NIST's **Zero Trust Architecture framework.**

06

Initial survey and inventory

The first step in a zero trust implementation effort is an organization-wide inventory of actors, assets, and processes. An enterprise must understand the current state of its operations before it can craft a security policy.

Identify actors

Identify every actor—human and machine—that interacts with the system. Actors may be employees, partners, contractors, third-party researchers, or collaborators. Actors may also be service accounts that are associated with in-house services and applications as well as third-party services.

Identify and manage assets

To craft a policy that determines how actors can access assets, an organization must first inventory and manage its assets. Assets may include enterprise-owned devices, employee devices as part of BYOD, and virtual resources. Monitoring and managing assets goes far beyond simply curating a list of devices. For each device, the enterprise's assets management system needs to be able to:

- Dynamically view the configuration, OS version, packages, and applications installed
- Monitor changes and, in some cases, allow/disallow certain changes
- View and manage keys and certificates on the device
- Update or install additional tools when onboarding devices

Identify key processes and assess their risk

The activity of an enterprise organization can be seen as a set of business processes that:

- Involves actors accessing data on different assets
- Uses automation to ingest and operate on data
- Generates data artifacts

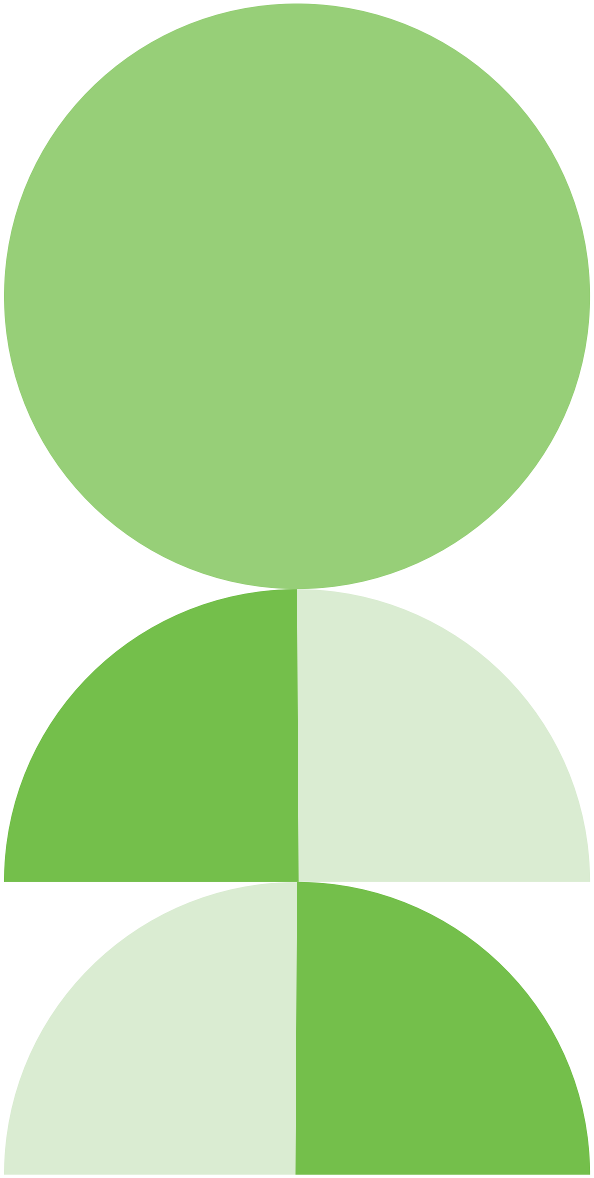
Every business process and data flow needs to be explicitly identified and evaluated for risk. Once all key processes are properly assessed, their security can be improved as part of an integration into zero trust. The goal is to protect the chain of custody of sensitive data to prevent attackers from extracting the data and corrupting or disrupting data processing.

At this point, you have a good view of your enterprise. All actors, assets, and key processes are known and properly cataloged. With this inventory in place, the next step is to consider policy.

Define policies

Consider each key process, the actors that participate in it, and the assets they need to access. With this information in mind, establish policies that allow the actors to perform their duties with the least amount of access. Keep in mind that the "least amount of access" doesn't only imply restricting the set of assets an actor can access. It can also include restricting access to specific times (for example, only during office hours on weekdays) or limits (for example, querying no more than one GB per day).

At this point, a major part of the preparation work for a zero trust architecture is done. It's time to choose a concrete solution.



Choose a candidate solution

Implementing a zero trust architecture is a big deal. As with many complex solutions, you might build it yourself, but it's likely more time and cost effective (and secure) to instead invest in hardened, battle-tested solutions from organizations that build their business around providing zero trust solutions.

As you evaluate solution providers, you should carefully weigh your enterprise needs:

- Be sure they can implement zero trust everywhere your workload runs—regardless of who is running or accessing your apps.
- Be sure they provide visibility and control across all products and all environments—not just cloud-native.
- Be sure the solutions are straightforward to implement. If your SREs don't understand the solution, it probably won't work.
- Be sure the solutions create a better user experience, not a more cumbersome one.

The solution must be mature, and you should be able to trust your zero trust provider. Note that even if the solution you choose addresses your current needs, it must be able to address future needs as well, especially as your enterprise grows and incorporates additional business processes via internal evolution or acquisitions.

Deploy and monitor an initial solution

After selecting a solution, deploying the initial version is a major project. You should select a set of key business processes that is significant, but not too ambitious. You should have an escape plan in place for halting the zero trust architecture and reverting back to the previous system if you encounter major issues.

It is also a good idea to run the new zero trust architecture initially in reporting mode only. In this mode, the policy engine detects policy violations and informs administrators, but does not block access. After you gain confidence that the zero trust architecture operates correctly and makes good decisions, then you can turn on enforcement mode. You always have the option to go back to reporting mode if things go wrong.

Iterate

The initial zero trust architecture typically manages just a small subset of business processes. But once you've gained experience onboarding business processes into the system and observing the system's behavior, you can begin to expand. Carefully introduce more and more processes to zero trust, following the same guard rails as the initial deployment.

Zero Trust Implementation: Caveats

Like any other large scale strategic change, a zero trust implementation is complicated and requires cooperation from a range of stakeholders. For all its benefits, zero trust is not a silver bullet. Organizations often struggle with their zero trust implementations because of the following factors.

Incomplete preparation

Mapping the actors, assets, and key business processes of a large enterprise is a massive effort. And it's not a static, one-and-done procedure; it's continually evolving. Since effective zero trust relies on an accurate view of actors, assets, and key processes, you'll need to build systems that can provide real-time views into this data.

Misconfigured policies

A successful zero trust algorithm operates based on the configured policies, and shortcuts can cause failure. For example, while working through issues in implementation, organizations may end up replicating their traditional perimeter-based security model by giving all applications and services full access to one another. While this makes migration easier, and might help to meet a deadline, circumventing the principle of least privilege will defeat your efforts to move toward zero trust.

Organizational change failures

Current key business processes are likely entrenched, not documented well, and rely on wide access to resources. When transitioning to a model such as zero trust that requires a meticulous understanding of the key business processes and precise definitions of the access needed for each asset and actor, internal resistance can be high.

Missing schedule and budget targets

Like many major projects, the scope and costs of a full-fledged zero trust security implementation in a large enterprise are difficult to estimate. If poor assumptions are made, or information is misunderstood, it's easy to underestimate the resources needed for success.

Section 07

Zero Trust Implemented in Different Environments

07

Now that we’ve covered in detail the steps involved in implementing a zero trust solution, let’s look at a few concerns unique to some of the most common situations.

Multi- and hybrid-cloud

Many enterprise organizations operate on a multi- or hybrid-cloud architecture. In these situations, it is critical for each zero trust implementation to integrate with the others. This includes constantly ingesting cloud asset state, logs, and metrics into a global zero trust implementation. It also involves mapping cloud IAM to the global zero trust identities and policies.

On-Prem

When using your own data center, you have more control, but also more responsibility. The details are too many to list here, but be aware if you want to implement zero trust in your own data center, you’ll need to effectively reproduce all the zero trust capabilities built-in to cloud providers.

Edge

If you run devices on the edge (such as IoT appliances or sensors) that touch vital data, you must include them in your zero trust architecture. Physical security can be a big concern here, and the ability to remotely monitor, upgrade, and shut down these devices in the face of a threat is paramount.

Kubernetes

Although Kubernetes is not a true deployment environment, it's important to understand how Kubernetes deployments impact zero trust architectures. Kubernetes serves as an abstraction layer that often hides the underlying environment and provides its own concepts of users, services accounts, role-based access control, and policies. You’ll need to consider all these aspects for your zero trust architecture.

Conclusion

Enterprises can no longer rely on traditional perimeter-based security. Distributed networks, modern applications, and new ways of working all require adopting a new approach to security—zero trust. While implementation of zero trust is not always simple, taking the time to understand how it works, how it can be implemented at your organization, and your wide range of options can result in a more secure, reliable, and resilient environment.

About Cisco Developer

Cisco Developer helps developers deliver more secure, better-performing software with forward-thinking, actionable insights into applications and environments, wherever developers choose to build.

Duo offers simple, powerful access security, using multi-factor authentication and zero trust principles, to protect your organization at scale.

SecureX is a cloud-native platform connecting your security portfolio to your infrastructure. It amplifies network security with unified visibility and intuitive automation.

Identity Services Engine (ISE) manages endpoint, user, and device access to network resources within a zero trust architecture.