Engineer

Zero trust

a model that grants no trust on network location alone

What it is

Zero trust is a security model in which nothing is trusted because of where it sits on the network. NIST SP 800-207, the US federal publication that set out the model in 2020, defines it as a paradigm focused on protecting resources, on the premise that trust is never granted implicitly and must be evaluated continually. Being inside the office network, on the VPN or in the same data centre earns a request nothing on its own. Every request for a resource is decided on its merits: who is asking, from which device, for what, and under what conditions. Written with a hyphen, zero-trust qualifies a noun, as in a zero-trust design, and the plan that puts the model to work is called a zero trust architecture, or ZTA.

NIST names the parties precisely. A subject is whoever or whatever makes a request: a user, a service or a device. A resource is anything worth protecting, and the publication counts printers and compute capacity alongside data. Between the two stand three logical components. The policy engine decides whether a request is granted and logs the decision. The policy administrator carries it out, opening or closing the path between subject and resource and issuing the credentials or tokens for that one session. The policy enforcement point sits in the path itself: it lets the session through, watches it and ends it when told to.

The publication explains the shift with an airport. Passengers pass one screening checkpoint, and beyond it travellers, crew and airport staff mill about the terminal, all treated as trusted. That terminal is what NIST calls an implicit trust zone: an area where everyone is trusted to the level of the last checkpoint they passed. A flat office network works the same way. Zero trust keeps the zone as small as possible by moving the decision and its enforcement closer to the resource.

Use the concept in a decision

Put “What is zero trust?” to work in a real decision. We can help relate the concept to the system you are building or buying. Accounts, passwords and permissions arranged so people hold exactly the access their job needs, and leavers lose it that day.

Explore our services Get in touch →

Test the assumptions behind the term

Use the explanation and linked work to frame questions for your project. We can examine the assumptions with you before a tool is chosen. Wrote 1,000 RBAC rules to secure Linux servers — 1,000 access rules written to close a security gap

Wrote 1,000 RBAC rules to secure Linux servers Get in touch →

Why it matters

The older model draws one line around the organisation and trusts what is inside it. NIST’s objection is plain: once attackers breach the perimeter, lateral movement is unhindered, and unauthorised lateral movement has been one of the biggest challenges for the agencies the publication was written for. The line has also stopped matching the work. Staff sign in from home and hotel networks, personal devices are allowed onto some systems, and much of what they use runs in someone else’s cloud. NIST’s assumptions follow from that: the private network is no implicit trust zone, an attacker may already be on it, and remote staff should treat the local network they connect from as hostile.

What a team gains is containment. Under the third of NIST’s seven tenets, access is granted per session with the least privilege the task needs, and authenticating to one resource grants nothing on another. A stolen password or a compromised laptop then reaches what that account and device are allowed to reach, at that moment and through one enforcement point, rather than the whole internal network. The decisions also leave a trail. The policy engine records every grant and every denial, so the question of who could reach a system, and why, is answered from records rather than from memory.

What it looks like in practice

Take a purchasing system, the kind of application NIST suggests for a first migration because a defined group of staff uses it, unlike email, which the whole organisation depends on. A buyer opens it from a company laptop. The request reaches the enforcement point in front of the application, which asks the policy engine. The engine’s trust algorithm weighs the inputs NIST lists: the request itself, including the device’s operating system version and patch level; the subject database; the asset database; the resource’s own requirements, which may name

an authentication assurance level or refuse addresses abroad; and threat intelligence. An illustrative decision record, with field names of our own rather than any product's, might read:

```
request   buyer-17 on laptop-0417 -> purchasing app, read and write
subject   procurement team; signed in with a phishing-resistant key
asset     enterprise-managed, OS patch level current, disk encrypted
resource  needs AAL2 or higher, a managed device, a domestic source IP
threat    source address absent from the threat feeds consulted
decision  grant read and write for this session; log as approved
```

The grant covers this one session and this one application. When the same buyer, on the same laptop, opens the supplier contracts kept in a document store, the engine makes a new access decision: authentication to the first resource does not automatically grant access to the second. Context counts too. One of NIST's examples is an accountant who normally works office hours and suddenly asks for the financial system in the middle of the night from a location the engine does not recognise: a contextual trust algorithm can raise an alert and require the subject to meet a stricter confidence level first. Another is an HR employee who reads 20 to 30 personnel records on a normal day and then asks for more than 100.

For cloud services, NIST points out that clients can go to the service directly, with the organisation's enforcement points checking policy on the way, rather than being brought into the internal network over a VPN first. Every one of these decisions depends on a clean picture of who should reach what, and NIST asks for a survey of assets, subjects, data flows and workflows before anything is built. That groundwork is the kind of consolidation described in our case on access management across M365 and SSO: identities and access brought together across M365, 1Password, Red Hat SSO and OKTA SSO, least privilege enforced, and onboarding and offboarding standardised so that access is granted and revoked promptly.

Where it goes wrong

Buying it as a product. NIST states that zero trust is a set of guiding principles rather than a single architecture, and that there is no single set of tools or services for a zero trust enterprise. A new remote-access gateway with the label on the box changes little if every subject it admits still reaches the whole network behind it. Fix: treat the move as, in NIST's words, "a journey rather than

a wholesale replacement", one business process at a time. Judge each product against the tenets it is meant to satisfy, including whether its data formats would let you leave it for another provider later.

Writing policy before the inventory. The policy engine can only decide from what it knows about subjects, assets and processes. Where that knowledge is incomplete, NIST warns, the usual result is a business process that stops because the engine denies requests for lack of information, and shadow IT, the systems set up outside the IT department's management, makes it worse. Fix: survey and catalogue the assets, subjects, data flows and workflows first, shadow IT included, and only then write the criteria for a candidate process.

Administrators keep blanket access. In many older designs, developers and system administrators can reach every resource, and NIST singles these accounts out for extra scrutiny. Moving everyone else behind per-session decisions while the administrator accounts keep their standing reach leaves the most valuable credentials as the shortest way in. Fix: send privileged accounts through the same engine, give them the access their current task needs, and use logs and audits to learn their normal patterns so that departures from them stand out.

The policy engine becomes the new perimeter. No resource talks to another unless the engine and the administrator approve it. Anyone with access to the engine's rules can make unapproved changes or plain mistakes, and an attacker who disrupts the enforcement points or the administrator can stall the business. Fix: log and audit every configuration change, run the enforcement in a properly secured cloud environment or replicate it in several locations, and give the stored traffic data and policy the most restrictive access of all, from dedicated administrator accounts only.

Enforcing everything on day one. Few policy sets are right on the first pass, and NIST expects a tuning phase in which requests that should succeed are denied. Switching straight to strict enforcement turns every gap in the policy into staff locked out of work they need. Fix: run the new workflow in reporting-only mode for a while. Grant most requests, compare the logs with the intended policy, and enforce only the basics, such as denying requests that fail MFA or come from addresses known to be hostile. Tighten the policy once the normal pattern is understood.

Treating MFA as the finish line. MFA lowers the risk from a stolen password, but NIST points out that an attacker holding valid credentials, or a malicious insider, can still reach whatever the account is allowed to reach. Fix: keep each account's reach narrow, and use a contextual trust algorithm that weighs time, location and volume, so that valid credentials used in an unusual way still raise an alert.

Further reading

- NIST SP 800-207, Zero Trust Architecture — published in August 2020: the definition, the seven tenets, the logical components, the threats and a road map for migrating.
- NIST SP 1800-35, Implementing a Zero Trust Architecture — the NIST practice guide published in June 2025, describing 19 example implementations built with 24 collaborators.
- NIST SP 800-207A — published in September 2023: access control for cloud-native applications across several clouds, based on API gateways, sidecar proxies and service identities rather than network addresses.
- NIST SP 800-63B-4 — the authentication assurance levels a resource's requirements can name, of which AAL3 requires phishing resistance.

Glossary

Choose for your actual constraints

Bring the deployment path, the operational risks and the people on call. We can scope a change with checks, a recovery route and clear ownership.

See what it costs Get in touch →

Make the terminology practical

Ask how “What is zero trust?” applies to your organisation. A concrete example from your current system is enough to start the discussion.

Explore our services

welcome@engineer.company +45 50 10 61 84 Copenhagen, Denmark

Take this page with you

Contact card

Carefully Crafted With Love by D.

CVR 45969703 · Tagensvej 105C, 3. th, 2200 København N

Copyright © 2025 – present · Engineer ApS · All rights reserved.

Printed from https://engineer.company/glossary/zero-trust/