Alexei Balaganski, Nov 2026

Buyer's Compass

Zero Trust Platforms

© 2026 KuppingerCole Analysts AG

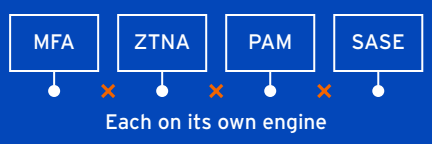
Commissioned by NetFoundry

Zero Trust is now the default architecture for enterprises whose resources are no longer protected by a single perimeter.

Users, workloads, and data are spread across data centers, public clouds, SaaS, Kubernetes clusters, operational sites, partner networks, and the edge, so location no longer indicates whether a connection can be trusted. A growing share of that connectivity is non-human, including workloads, APIs, AI agents, industrial and connected devices, and third-party systems, and much of it crosses network and organizational boundaries.

© 2026 KuppingerCole Analysts AG

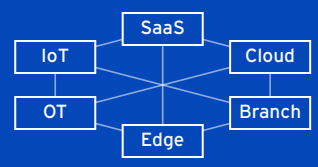
The Challenge



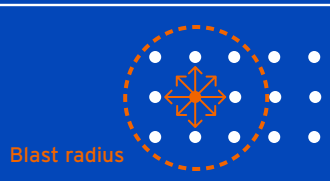
Fragmented Enforcement



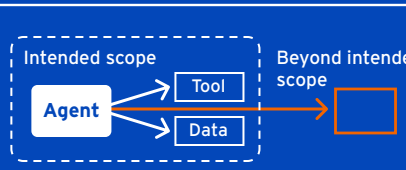
Abuse of Valid Identities



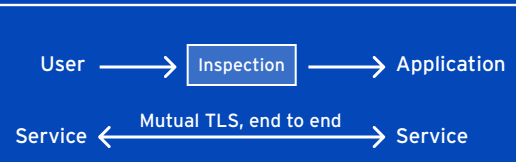
Distributed Infrastructure



Lateral Movement



NHIs and AI Agents



Person or Machine

Most enterprises already own the parts: Multifactor Authentication (MFA), Zero Trust Network Access (ZTNA), endpoint posture, microsegmentation, and cloud workload controls are widely deployed. These controls lack a shared policy model. Each runs on its own engine, with its own owner and view of risk, so they rarely make decisions together. Vendors enter the category from different origins, including ZTNA, Secure Access Service Edge (SASE), Privileged Access Management (PAM), microsegmentation, endpoint and cloud security, API security, and data protection, and each origin shapes what the resulting platform does well.

This report distinguishes throughout between a person and a machine trying to connect. Human access and workload connectivity rest on the same principles, but the policy subject, the credential lifecycle, the enforcement model, the telemetry, and the failure modes differ. A human session can usually tolerate an intermediary that terminates and inspects traffic, while workload connections are symmetric, frequent, and mutually authenticated, so the same intermediary can break end-to-end authentication, expose data to a third party, and add latency. The market broadly agrees that Zero Trust must extend to machines as well as people, even if practice still lags behind. How differently the two must be handled in policy, enforcement, and daily operations remains open.

Human access	Workload connectivity
Connection Human session	**Connection** Symmetric, frequent, mutually authenticated
Intermediary Can usually tolerate one that terminates and inspects traffic	**Intermediary** Can break end-to-end authentication, expose data to a third party, and add latency

Same principles, different policy subject, credential lifecycle, enforcement model, telemetry, and failure modes.

Fragmented Enforcement

Various identity, security, and compliance controls each cover a slice of Zero Trust, rarely under one authoritative policy. Rules diverge across SaaS, private applications, cloud workloads, privileged systems, and third-party access. Enforcement and governance break down most visibly where network, cloud, technology, and organizational boundaries meet, which is where non-human and multi-domain flows are governed least consistently. A signal raised by one tool rarely changes a decision made by another. Policies drift, duplicate, and expire, and no one can name the authoritative source for a given decision. Closing this gap is the first job of a Zero Trust platform, and the first thing a buyer should test.

Abuse of Valid Identities and Credentials

Identity is now the primary attack surface. Stolen credentials, session cookies, OAuth tokens, API keys, and service-account secrets let attackers operate as legitimate users and services, where perimeter defenses and one-time checks see nothing wrong. Containing this requires continuous evaluation of trust across behavior, posture, identity risk, resource sensitivity, and session context, and the ability to act mid-session through token revocation, step-up authentication, privilege reduction, or termination. Machine credentials add risk of their own, since leaked service-account secrets and long-lived keys grant standing access that is rarely reviewed. A shared secret or bearer token proves only possession: the check validates the credential presented but not that the sender is the workload it was issued to, which amounts to checking an ID card without looking at the face. A workload therefore needs its own cryptographic identity, verified on every connection.

© 2026 KuppingerCole Analysts AG

Heterogeneous and Distributed Infrastructure

Enterprise infrastructure is distributed by design, and much of its traffic never touches a user. Workloads, devices, and services run across private data centers, multiple clouds, Kubernetes clusters, branch and factory sites, edge locations, third-party networks, and Internet of Things (IoT) fleets, and they connect across all of them, often within or between customer-controlled environments. Each environment brings its own addressing, logging, identity integration, and constraints, including overlapping and private IP ranges that defeat topology-based controls. A platform must enforce one policy across all of it, without depending on network layout or re-addressing traffic to fit a shared network.

Lateral Movement and Breach Containment

Prevention does not stop an attacker who is already inside, and the damage depends on what that attacker can reach next. A foothold on one host, account, or token becomes a path across a flat network, permissive service-to-service communication, or forgotten administrative routes. Limiting the blast radius means constraining which identities, workloads, and services can communicate at all, and staging, testing, and rolling back those constraints without breaking production. For workload connectivity, containing east-west movement between services is often the entire project from the outset.

Non-Human Identities, Automation, and AI Agents

Workloads are created, scaled, moved, and retired continuously, and policy and identity have to travel with them automatically, without manual re-registration at each event. A platform must govern this machine-to-machine traffic with the rigor applied to people and must do so for identities that cannot answer an MFA prompt or wait for an approval. When a copilot, agent, Large Language Model (LLM) gateway, or Model Context Protocol (MCP) server accesses tools and data on behalf of a user or process, the main risk is an authorized identity acting beyond its intended scope. Agent identity, scoped reachability, and fast revocation address this.

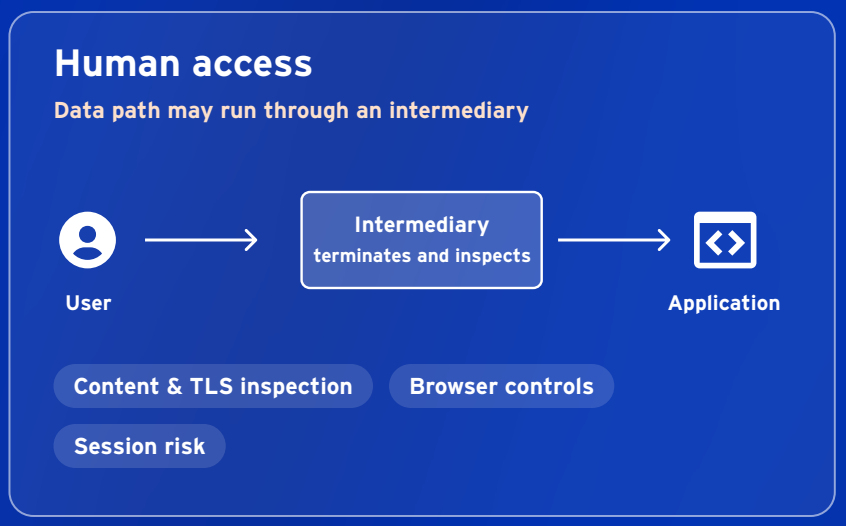
© 2026 KuppingerCole Analysts AG

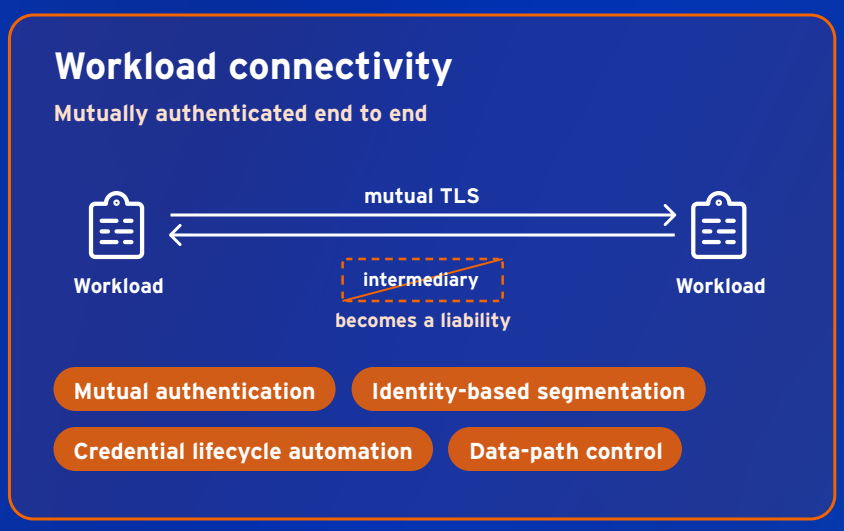
The Solution

A Zero Trust platform governs access by identity, context, and policy rather than by network location. Zero Trust reached human access first, as ZTNA gave people identity-based access to individual applications and allowed the workforce to move away from the VPN. Non-human connectivity has largely not made that shift, and workloads, services, third parties, and devices still reach each other through site-to-site VPNs and firewall rules that are not identity-aware. These flows need the same move to identity-based Zero Trust, delivered by an architecture designed for machines.

The market splits along two design centers. Human access relies on content and Transport Layer Security (TLS) inspection, browser controls, and session risk, and its data path can run through an intermediary that terminates and inspects the session. Workload connectivity depends on mutual authentication, identity-based segmentation, credential lifecycle automation, and control of the data path, where that same intermediary becomes a liability. A platform built for one center rarely covers the other by adding features, because the data path itself differs.

In principle, a Zero Trust platform brings policy, context, and enforcement together across users, workloads, services, and machine identities. No solution on the market covers every use case, and many vendors have focused on securing human access while machine-to-machine connectivity remains far less well served. Buyers should therefore establish where a platform concentrates before assuming it serves every scenario equally.





How the Solution Works

Authenticate Before Connect

Inspection is a powerful control for human sessions, but it is often impractical for non-human and third-party flows, which need stronger validation before a connection opens. A platform should require users, devices, workloads, services, APIs, and agents to prove identity before they can reach anything. Resources remain invisible to any user, device, or workload that is not entitled to reach them, which defeats network scanning, limits what an attacker holding stolen credentials can discover, and removes targets for opportunistic exploitation. Access should be granted to a specific resource under specific conditions. For machine connectivity, neither endpoint should expose a listening port at all.

Machine and AI Identity Protection

Non-human identities (NHIs) account for a large and growing share of access, and shared secrets and standing permissions are their weak points. A platform should extend Zero Trust to them through strong workload identity, short-lived and certificate-based credentials, workload identity federation, secrets reduction, and least-privilege service-to-service authorization. As agents invoke tools and APIs on behalf of users and processes, a platform should give

How the solution works

1. Authenticate before connect
2. Machine and AI identity protection
3. Central policy and contextual authorization
4. Continuous trust evaluation
5. Enforcement across heterogeneous infrastructure
6. Segmentation and breach containment
7. Monitoring, analytics, and response

© 2026 KuppingerCole Analysts AG

each agent its own identity, preserve the originating user context, scope which tools and data it can reach, audit its actions, and require human approval where the action warrants it. For a workload-centric buyer, identity and attestation come first: they establish what is communicating before any decision on whether it may communicate.

Central Policy and Contextual Authorization

A platform's policy engine should weigh identity, role, attributes, device and workload posture, location, network context, behavioral risk, threat intelligence, and resource sensitivity, and a mature engine moves beyond static roles into attribute-based and risk-adaptive decisions, just-in-time and time-bound access, and continuous re-evaluation. For machines, short-lived credential issuance, certificate rotation, and automated re-attestation should replace prompts and approvals. Policy should be governed centrally and enforced close to the resource.

Continuous Trust Evaluation

Authentication at the start of a session says little about the rest of it. When device compliance drifts, a malware alert fires, identity risk shifts, or a workload's posture changes, a platform should respond by revoking tokens, forcing step-up, cutting privilege, or terminating the connection. For workload traffic, it should check posture, certificate validity, and policy throughout the life of the connection, and reduce or withdraw reachability without disturbing the authorized flows around it.

Enforcement Across Heterogeneous Infrastructure

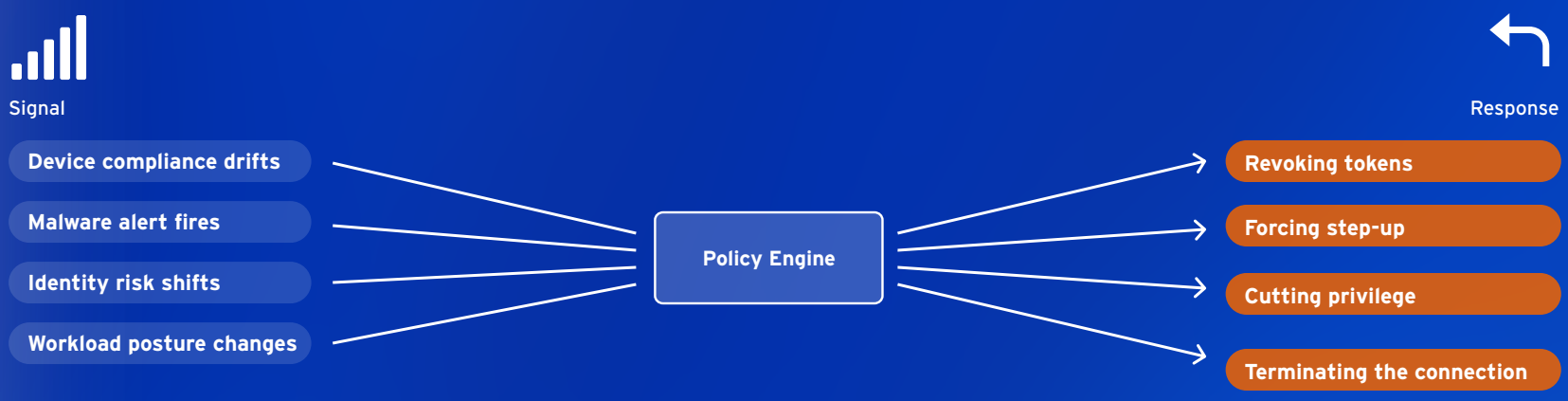
No single enforcement point fits data centers, clouds, clusters, branch sites, and edge at once, so a platform should combine the enforcement models each environment calls for: cloud brokers, private gateways, host agents, workload certificates, service-mesh integration, browser isolation, and cloud-native controls. A buyer should establish whether a cloud-hosted control plane forces a cloud-hosted data path. Policy should stay consistent across the estate, with traffic taking an appropriate route (customer-controlled where required) without backhauling to a central chokepoint.

Segmentation and Breach Containment

Access control cannot prevent every compromise, so a platform's second job is to limit the damage a compromise can do. Restricting which identities, workloads, services, ports, and processes can communicate turns a flat interior into a set of narrow, authorized paths. Platforms deliver this through application-level ZTNA, workload microsegmentation, service-to-service authorization, Kubernetes policy, or rapid quarantine. Segmentation tied to identity and service holds across data centers, clouds, edge, and Operational Technology (OT), with enforcement per connection.

Monitoring, Analytics, and Response

Enforcement must be observable to be trusted and improved. A platform should collect telemetry on access attempts, policy decisions, posture changes, workload flows, session activity, and anomalies, and feed it to Security Information and Event Management (SIEM), Security Orchestration, Automation, and Response (SOAR), Extended Detection and Response (XDR), and IT Service Management (ITSM) for investigation, compliance, tuning, and response. Strength in session and user-behavior telemetry does not imply strength in service-to-service flow telemetry. For machine traffic, telemetry should tie every event to a workload identity, service, and policy, and show both the connections allowed and the connections denied, on both sides of the link.



© 2026 KuppingerCole Analysts AG

Top Use Cases

These use cases are ordered by their weight for a workload-focused buyer, from the segmentation and machine identity projects that usually drive a workload program to the human access scenarios that workload-focused platforms typically treat as secondary.



1 Microsegmentation and Breach Containment

Segmentation limits what a compromise can touch by controlling which workloads, services, and flows may communicate. Zero Trust platforms map dependencies, propose least-privilege policies, and support staged rollout, so that tightening the interior does not break it. Default-deny, workload isolation, and rapid quarantine contain ransomware and unauthorized east-west movement. For workload-centric buyers this is usually the primary objective, and segmentation that follows identity rather than topology keeps it durable as the environment changes.

2 Workload-to-Workload and Site-to-Site Connectivity

Your workloads, APIs, containers, clusters, pipelines, and service accounts connect across clouds, data centers, branch and factory sites, and partner organizations, and each connection needs governing as strictly as any human access. Policy must follow workloads through create, scale, move, and retire events, which calls for close integration with Kubernetes, service meshes, API gateways, and policy-as-code. Mutual authentication between both ends of a connection, short-lived workload credentials, and least-privilege service-to-service authorization are the defining requirements. They apply equally to server-to-server, site-to-site, and third-party business-to-business connectivity, which also needs identity-based connections that cross network and administrative boundaries without a shared network or re-addressing. Workload-focused platforms are designed for this use case.

3 AI Agent and Non-Human Access Governance

Autonomous agents and automated workflows now act on behalf of users and processes, accessing models, tools, APIs, and data through delegated authority. Buyers need to know which agents hold identities of their own, which tools and endpoints they may invoke, when a human must approve, and how access is withdrawn when behavior or context shifts. Extending identity, scoped reachability, and revocation to agents, MCP servers, and LLM gateways keeps a manipulated or over-eager agent inside the boundary its identity was granted. Data-sensitive actions still warrant classification and approval where a human is in the loop, while purely machine flows are governed through the scope of the identity itself.

What buyers need to know

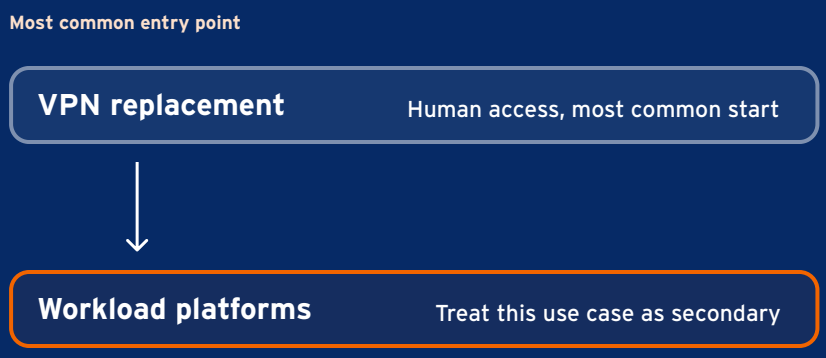
- ? Which agents hold identities of their own?
- ? Which tools and endpoints may they invoke?
- ? When must a human approve?
- ? How is access withdrawn when behavior or context shifts?

4 Secure Hybrid Workforce and Third-Party Access

VPN replacement is still the most common entry point into Zero Trust. Employees, contractors, partners, auditors, and service providers need access without exposing a network segment or trusting a location. A platform delivers this by authenticating the user first, hiding applications from anyone not entitled to them, and granting access to specific resources by identity, posture, context, and risk. Coverage for managed and

© 2026 KuppingerCole Analysts AG

unmanaged devices, clientless access, and legacy protocols such as RDP and SSH is critical in the mixed modern and legacy estates most enterprises run. This use case is human-centric, and platforms built for workload connectivity treat it as a secondary capability.



5 Continuous Risk-Adaptive Access and Session Control

Risk continues for the life of a session, and attackers work inside valid sessions using stolen tokens and compromised endpoints. The platform therefore rechecks identity, posture, behavior, location, and threat intelligence throughout a session and acts the moment the picture changes, through step-up, restriction, revocation, termination, or approval. Just-in-time and time-bound access remove standing privilege. The signals that drive this use case come largely from the surrounding security ecosystem, so integration is essential. The use case is primarily human-centric, since its workload counterpart, the continuous checking of certificate validity and posture, runs without prompts or approvals and is covered by continuous trust evaluation.

© 2026 KuppingerCole Analysts AG

Top Functional Selection Criteria

These capabilities are the yardstick for a Zero Trust platform. Their weights are not fixed: they change with the primary requirements and shift the most between human access and workload connectivity.

Secure Connectivity Enforcement

A platform must deliver identity-aware connectivity across users, devices, workloads, applications, APIs, and services, with authenticate-before-connect access, encryption, application cloaking, and policy-based reach into private resources across hybrid and multi-cloud estates. Look for multiple enforcement models: agents for managed devices, agentless or browser access for unmanaged users, connectors for private applications, and options for cloud, on-premises, and edge. For workloads the bar is higher: neither endpoint should expose a listening port, and reachability should be denied by default outside the authorized pair, so the connection is refused at the network level until identity and policy permit it. Whether mutual authentication is the default rather than an add-on is part of this test.

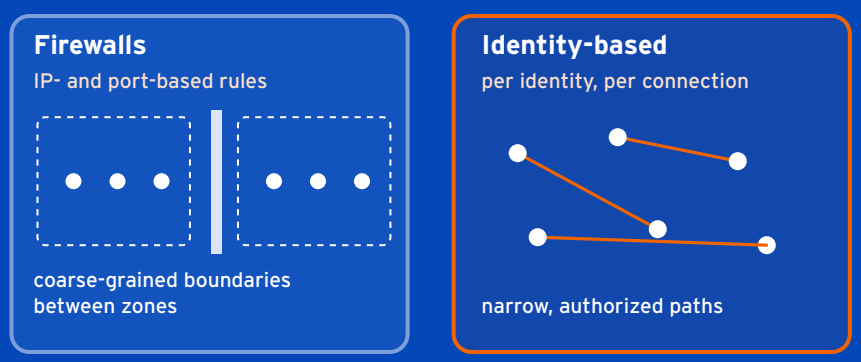
Machine and AI Identity Protection

NHIs are moving to the center of Zero Trust architecture. A platform should protect workloads, APIs, service accounts, containers, devices, and agents with strong workload identity, short-lived and certificate-based credentials, secrets reduction, workload identity federation, service-to-service authorization, and auditability of machine actions. AI agent governance is emerging quickly, so buyers should check whether the platform assigns agents their own identities, preserves user context, enforces least privilege, and requires human approval for high-risk actions.

Segmentation and Lateral Movement

A platform should make lateral movement hard, which takes more than remote access control. Look for identity-aware segmentation, workload isolation, dependency mapping, default-deny, and enforcement that follows identity rather than network topology or IP addresses, applied across data centers, clouds, Kubernetes, endpoints, OT, and service-to-service traffic. Firewalls still have a role as coarse-grained boundaries between zones, but IP- and port-based rules cannot express per-identity, per-connection policy, so segmentation should not depend on them alone. A bad segmentation policy can halt production, so test how safely policies can be modeled, staged, and rolled back. For workload-centric buyers this is often the most relevant capability and the primary project driver, ahead of VPN replacement.

Firewalls and identity-based segmentation



Access Management Policy

Policy is the core of the platform. It should authorize on identity, role, attributes, risk, resource sensitivity, posture, and context, and mature engines add attribute- and policy-based access, just-in-time and time-bound grants, step-up, approval workflows, and continuous re-evaluation. Central management with distributed enforcement is expected. Policy simulation, policy-as-code, APIs, and automation decide whether the model holds up in a real environment.

Context and Posture

Context evaluation decides whether to allow, restrict, challenge, or revoke access. A platform should draw signals from users, devices, workloads, applications, networks, and threat intelligence, with strong posture integration into endpoint management, Endpoint Detection and Response (EDR), and XDR, and identity risk from Identity and Access Management (IAM) and Identity Threat Detection and Response (ITDR). For workloads, location and topology are weak signals and should not be treated as grounds for trust. Provenance, vulnerability state, runtime attestation, and execution context are stronger indicators. Posture requirements

© 2026 KuppingerCole Analysts AG

should differ by resource rather than apply one rule to everything.

Monitoring and Analytics

Enforcement must be observable, with visibility into sessions, policy decisions, access attempts, workload flows, posture, and anomalies, and dashboards that answer who reached what, from where, and with what result. Advanced analytics add behavioral baselining, anomaly detection, risk scoring, and investigation workflows, feeding SIEM, SOAR, XDR, and ITSM. Some platforms provide built-in analytics and risk scoring. Others source these from external tooling by design and keep their own telemetry on identity-correlated flow and policy evidence.

Dashboards should answer

- **Who reached what?**
- **From where?**
- **With what result?**

Threat Prevention

A platform should help prevent and contain abuse of identities, endpoints, workloads, and services, including credential theft, session hijacking, malware propagation, and anomalous service-to-service activity. Depending on architecture this may involve inline inspection, malware scanning, browser isolation, protocol validation, or external detection. These content-inspection mechanisms are built for human-facing risk such as phishing and malicious downloads. For authorized, mutually authenticated service traffic they map poorly, and TLS inspection can be counterproductive, because it requires a third party to hold decryption keys for traffic already protected by mutual TLS. The workload equivalent is authenticated reachability, deny-by-default connectivity, and call-pattern detection. Buyers should judge prevention against the traffic the platform will actually carry.

Human-facing traffic
phishing, malicious downloads

- Inline inspection
- Malware scanning
- Browser isolation
- Protocol validation

Workload traffic
authorized, mutually authenticated

- Authenticated reachability
- Deny-by-default connectivity
- Call-pattern detection

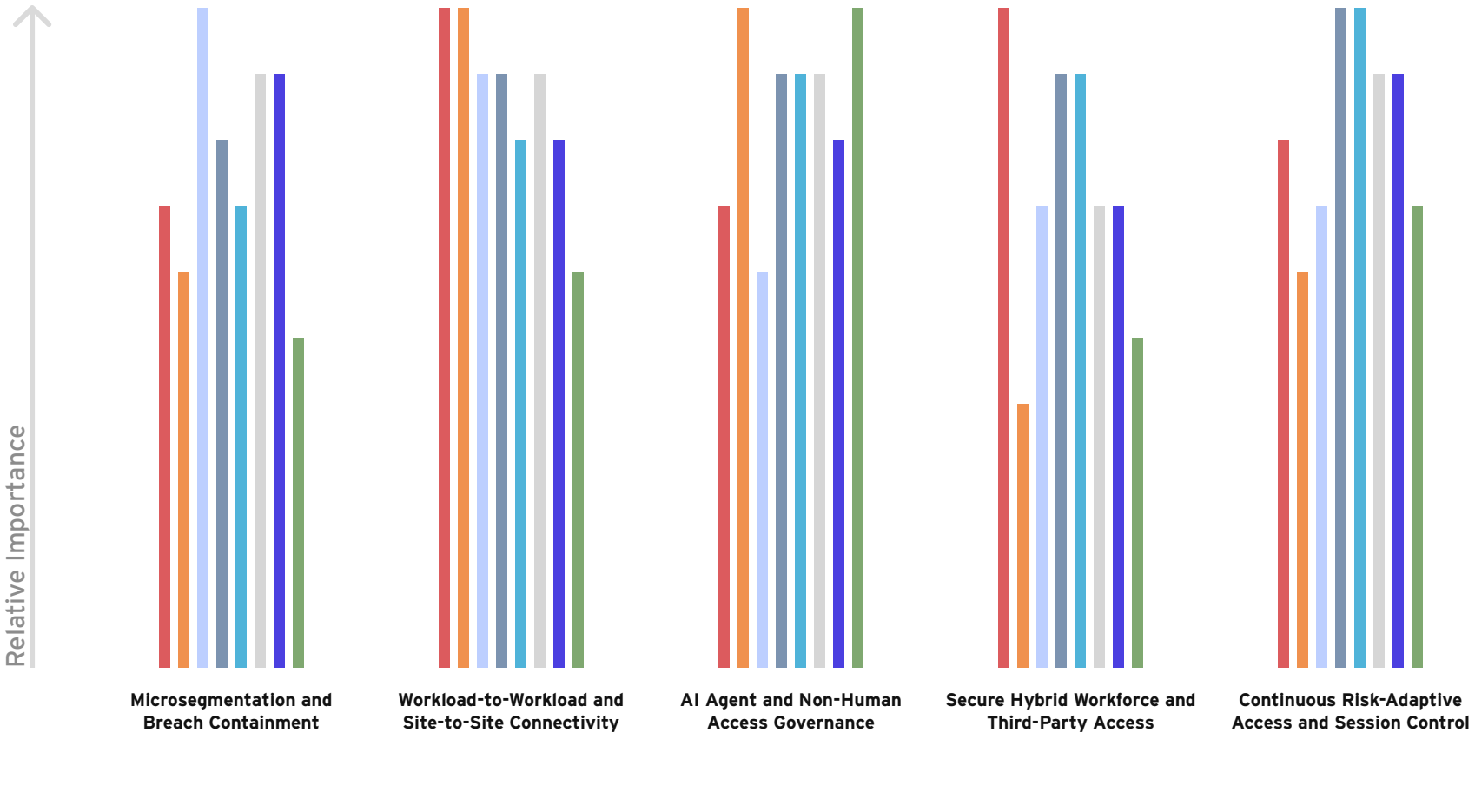
Data-Centric Security

Access to an application does not confer the right to every action inside it. Platforms address this through integration with classification, Data Loss Prevention (DLP), Data Security Posture Management (DSPM), Digital Rights Management (DRM), encryption, browser controls, and audit, applying different controls by data sensitivity, user context, and device. Most of these controls govern human actions inside a session, and download blocking, watermarking, and copy-paste restriction have little bearing on automated machine flows. Buyers whose priority is workload connectivity should weight this criterion accordingly, and where data protection does matter for machines, assess it as a function of workload identity scope rather than content handling.

© 2026 KuppingerCole Analysts AG

Use Case to Capability Mapping

The diagram below plots the weight of each capability across the five use cases. Each use case relies on a different subset of the capabilities, so buyers should map and prioritize them against their own requirements.



The weight of each capability across the five use cases

1 Microsegmentation and Breach Containment	2 Workload-to-Workload and Site-to-Site Connectivity	3 AI Agent and Non-Human Access Governance	4 Secure Hybrid Workforce and Third-Party Access	5 Continuous Risk-Adaptive Access and Session Control
Most weight				
Segmentation and Lateral Movement	Secure Connectivity Enforcement Machine and AI Identity Protection	Machine and AI Identity Protection Data-Centric Security	Secure Connectivity Enforcement	Access Management Policy Context and Posture
Least weight				
Data-Centric Security	Data-Centric Security	Segmentation and Lateral Movement	Machine and AI Identity Protection	Machine and AI Identity Protection

© 2026 KuppingerCole Analysts AG

Vendor Spotlight

Founded in 2019 and headquartered in Charlotte, North Carolina, NetFoundry is the creator and maintainer of OpenZiti, the open-source Zero Trust networking software underlying its platform. It is sold by subscription as vendor-hosted SaaS, self-hosted on-premises or private cloud, or hybrid across the three planes. NetFoundry focuses on non-human and third-party connectivity, machine identity, and microsegmentation, across use cases such as AI, APIs, server-to-server, site-to-site, IoT, and OT, with human remote access a secondary capability through NetFoundry Frontdoor. This focus is deliberate: the market for securing human access is crowded with established solutions, while machine-to-machine connectivity is far less well served, and NetFoundry has built its platform around the latter.

Founded

2019

Headquarters

Charlotte, North Carolina

Compliance

SOC 2 Type II

Platform

Zero Trust networking built on OpenZiti

Deployment

SaaS, self-hosted, or hybrid

NetFoundry

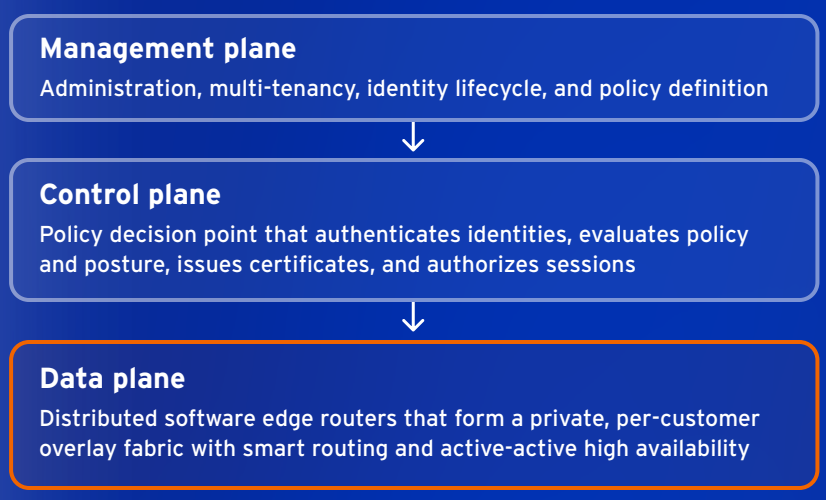
KuppingerCole Leadership Compass

Zero Trust Platforms 2026

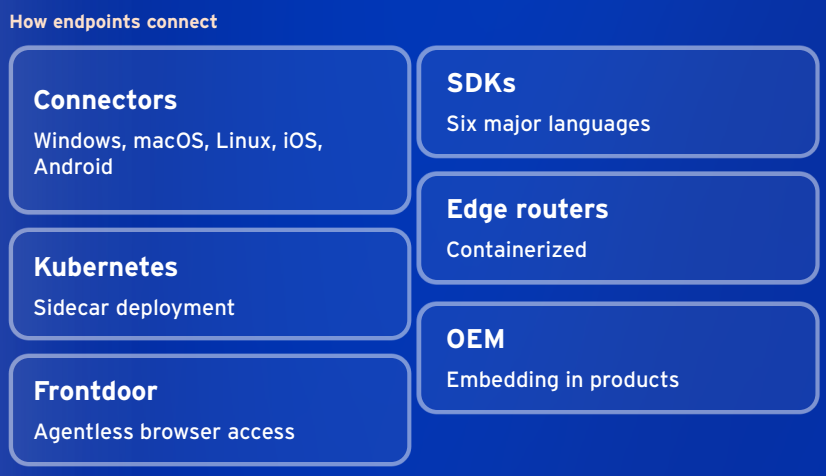


© 2026 KuppingerCole Analysts AG

NetFoundry uses a strict three-plane architecture. The management plane handles administration, multi-tenancy, identity lifecycle, and policy definition. The control plane acts as the policy decision point that authenticates identities, evaluates policy and posture, issues certificates, and authorizes sessions. The data plane consists of distributed software edge routers that form a private, per-customer overlay fabric with smart routing and active-active high availability.



Private keys are generated on the originating endpoint and never leave it, so the control plane issues certificates without access to key material and cannot decrypt customer traffic. Endpoints connect through lightweight connectors for Windows, macOS, Linux, iOS, and Android, embedded SDKs for six major languages, Kubernetes sidecar deployment, containerized edge routers, agentless browser access via Frontdoor, and OEM embedding. Air-gapped and edge deployments are also supported.



For authorization, NetFoundry follows the Attribute-Based Access Control (ABAC) model, with granularity from server and application down to API route, and policies managed through the console, REST APIs, or Terraform. Evaluation is continuous, so policy changes, certificate revocation, or posture failures terminate active sessions and make services invisible to that identity. The connector and SDK assess device posture using signals such as operating system version, domain membership, running processes, and MFA state. Risk scoring, behavioral baselining, and anomaly detection come from third-party EDR, XDR, SIEM, and SOAR tooling, whose signals can trigger policy changes, session termination, or re-authentication.

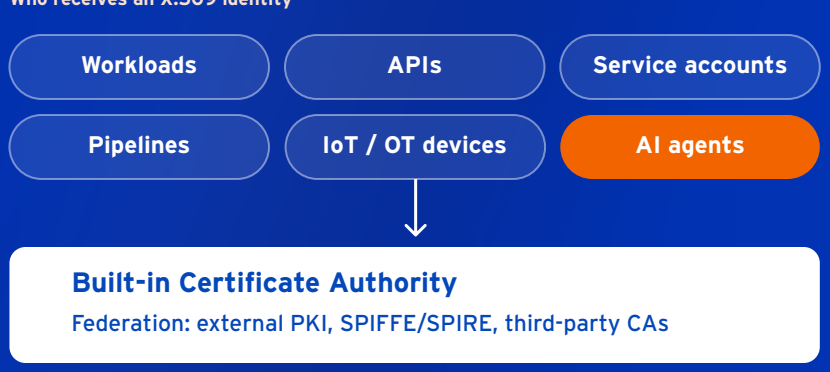
Evaluation is continuous, so policy changes, certificate revocation, or posture failures terminate active sessions and make services invisible to that identity.

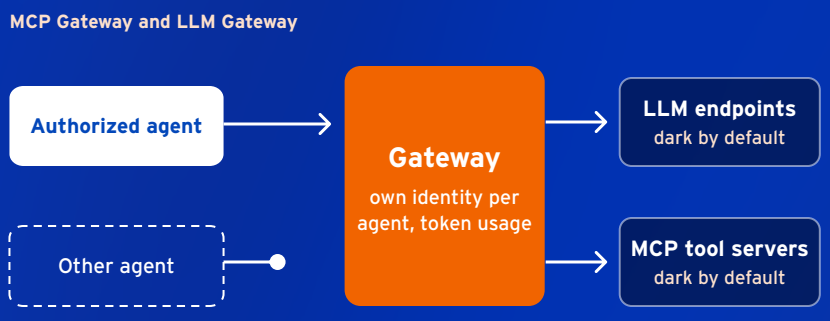
NetFoundry differs from conventional ZTNA by covering non-human and third-party use cases, including flows usually left to site-to-site VPNs and firewall rules, with the same identity model. Its overlay model enforces service-level, identity-based microsegmentation independent of network topology, cloud, or ownership. One policy model governs both human access and machine-to-machine traffic across cloud, on-premises, edge, OT, IoT, and partner domains. Connections form only between authenticated identities and named services, which contains lateral movement. Kubernetes is covered by pod sidecars, container SDKs, and cluster-level edge routers. NetFoundry zLAN uses eBPF in the edge router to extend identity-based segmentation at the IP layer, without deep packet inspection, to IP applications that cannot be modified and to OT and IoT devices that cannot run an endpoint agent. NetFoundry also operates the data path itself, as a private overlay dedicated to each customer, rather than acting only as a policy layer over an existing network.

NetFoundry records per-identity session logs, flow metadata, policy decisions, certificate events, posture results, and plane health, shows them in real time in the console, and streams them by syslog or REST API. Automated responses include session termination, identity revocation, step-up authentication, and device isolation. The platform performs no TLS termination, malware scanning, or protocol inspection, although traffic that needs them can be routed to external tools. Token theft and hijacking are mitigated by endpoint-held private keys, ephemeral session keys bound to mutual TLS, and continuous re-authorization. A connection is established only when the endpoint proves possession of a private key that never leaves it, its posture checks pass, and policy authorizes that specific identity to reach the service.

© 2026 KuppingerCole Analysts AG







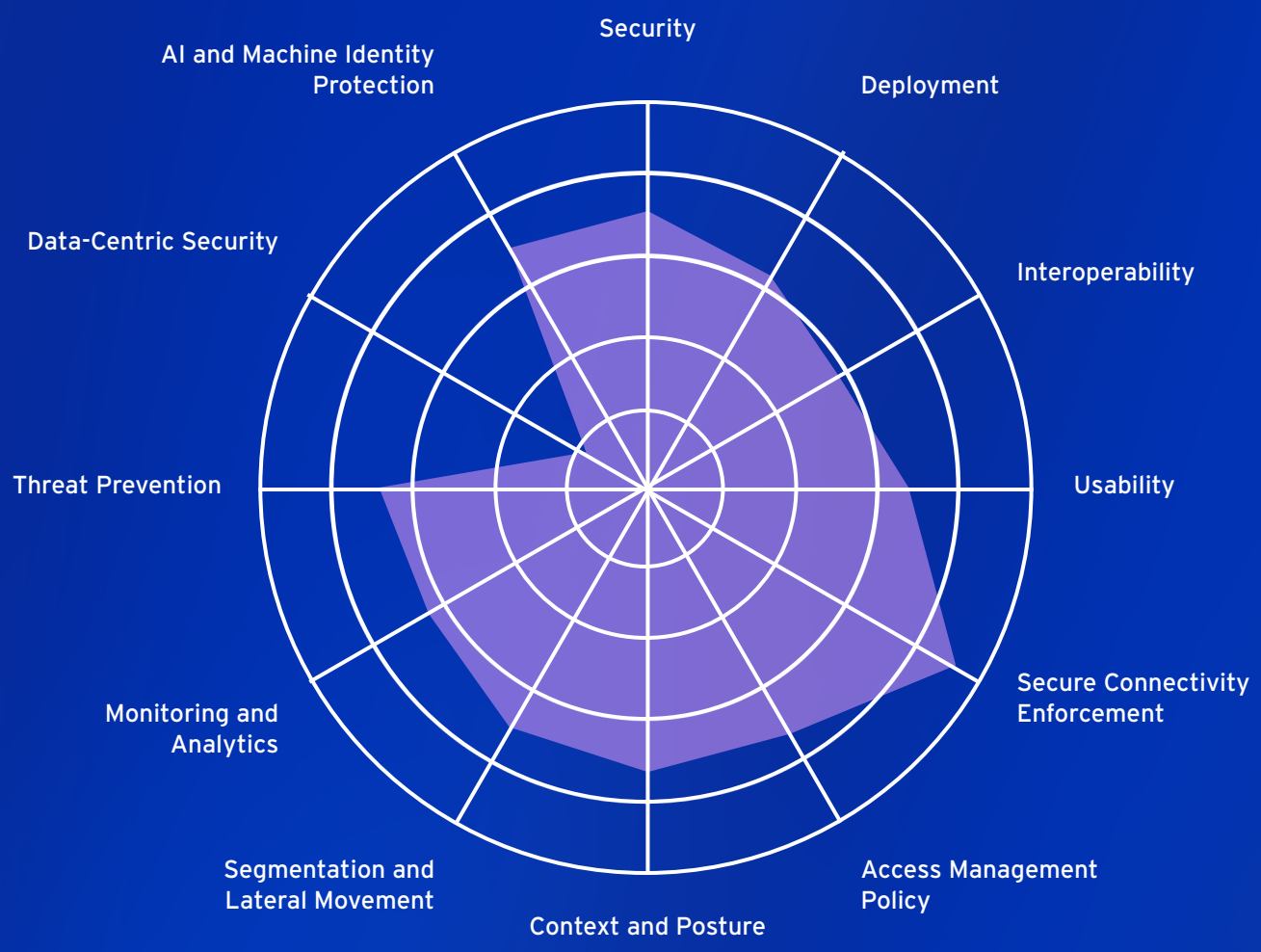
Workloads, APIs, service accounts, pipelines, IoT and OT devices, and AI agents receive X.509 certificates managed automatically by the built-in Certificate Authority (CA), with federation through external Public Key Infrastructure (PKI), SPIFFE/SPIRE, and third-party CAs. NetFoundry's MCP Gateway and LLM Gateway extend the model to agentic AI. They make LLM endpoints and MCP tool servers dark by default unless an agent is authorized, issue agents their own identities instead of inherited credentials, and track AI token usage. NetFoundry holds SOC 2 Type II attestation. Policy-as-code is mature through SDKs and Terraform.

NetFoundry suits organizations whose priorities are Zero Trust connectivity and microsegmentation for non-human workloads, including AI, server-to-server and site-to-site connectivity, OT, IoT, and third-party systems. It also lets developers, Independent Software Vendors (ISVs), and SaaS providers embed Zero Trust connectivity inside their own products, including under original equipment manufacturer (OEM) and white-label models. These networks run NetFoundry-hosted or self-hosted, with self-hosting covering air-gapped and data-sovereignty-sensitive environments, and in either model private keys stay on the endpoints. In the KuppingerCole Leadership Compass on Zero Trust Platforms 2026, NetFoundry is rated as Product, Innovation, and Overall Leader.

© 2026 KuppingerCole Analysts AG

NetFoundry

Capabilities



+ Strengths

+ A single identity model covers human, non-human, and third-party connectivity, including flows otherwise handled by site-to-site VPNs and firewall rules.
+ Private keys are generated on the originating endpoint and never leave it, so NetFoundry cannot decrypt customer traffic.
+ Service-level microsegmentation for workload-to-workload, service-to-service, API-to-API, and machine-to-machine flows from a single overlay fabric.
+ MCP Gateway and LLM Gateway make LLM endpoints and MCP servers invisible by default and issue cryptographic identities to AI agents.
+ OT/IoT and unmodified IP application support through NetFoundry zLAN, including devices that cannot run an agent.
+ SDKs embed Zero Trust enforcement directly in applications.
+ ISV customer-connect support and OEM and white-label models allow NetFoundry to be embedded in third-party products.

- Challenges

- No built-in behavioral analytics or risk scoring; relies on external SIEM, SOAR, and identity tooling.
- Workload, API, OT/IoT, and developer-embedded use cases are the primary focus; human remote access is a secondary capability.
- No built-in DLP, malware scanning, or content inspection; inline inspection requires external tools.
- Limited market presence and brand recognition, primarily known in OT/IoT and developer communities, although its enterprise workload deployments are growing.

© 2026 KuppingerCole Analysts AG

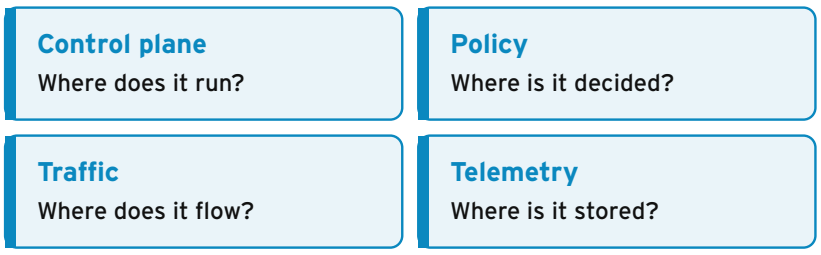
Considerations

Before initiating a Request for Information (RFI), Request for Proposal (RFP), or proof of concept, an organization should be clear about its architectural, operational, and governance requirements. Select for the closest fit to the organization's Zero Trust maturity, infrastructure, risk profile, and intended architecture.

Architectural Considerations

Deployment Model

Platforms are delivered as SaaS control planes, cloud-delivered fabrics, customer-deployed gateways, host agents, browser access, workload-side enforcement, service-mesh integrations, and self-managed or hybrid builds. SaaS simplifies onboarding and distribution. Regulated, government, industrial, and sovereignty-sensitive environments often require private cloud, on-premises, or air-gapped options. Establish where the control plane runs, where policy is decided, where traffic flows, and where telemetry is stored. Some platforms route traffic through vendor-operated clouds, while others keep the data plane inside the customer environment. The right choice turns on latency, resilience, residency, compliance, and the sensitivity of what is protected.



Data-plane Ownership and Path Locality

For workload connectivity this deserves its own line of inquiry. High-frequency service-to-service traffic tolerates far less added latency than a human session, and much of it is east-west inside customer-controlled environments. Ask whether a cloud-hosted control plane forces a cloud-hosted data path, whether the provider or any third party can decrypt traffic already protected by mutual TLS, and whether a provider outage would take down internal service-to-service communication rather than only internet-bound traffic. Key custody, path locality, throughput under bursty machine traffic, and failure behavior should be measured against actual workload patterns.

Cryptographic Agility and Longevity

The certificates, keys, and ciphers behind every connection will need replacing during the life of the deployment, whether because an algorithm is deprecated, regulation demands FIPS 140-3 validated modules, or post-quantum algorithms become mandated. Crypto-agility is the ability to make that change through software and policy, without new hardware or a redesign of the environment. Ask how an algorithm or module is replaced and its predecessor retired across the estate, whether credentials are short-lived and rotate automatically at scale, and what the vendor's roadmap is for hybrid and post-quantum key exchange. Test how larger keys and certificates affect handshake size and latency, since high-frequency machine-to-machine traffic will feel that overhead first.

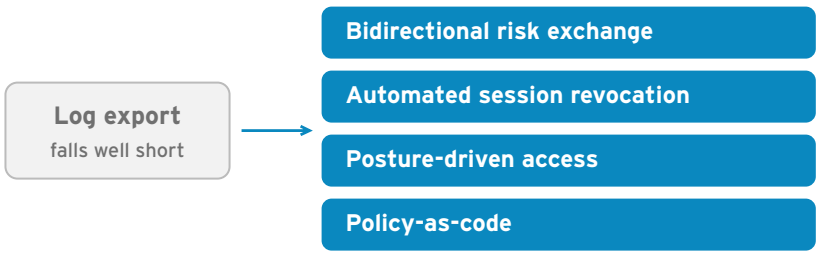
Crypto-Agility checks

- ☑ Replace an algorithm or module through software and policy
- ☑ Retire its predecessor across the estate
- ☑ Short-lived credentials rotate automatically at scale
- ☑ Roadmap for hybrid and post-quantum key exchange
- ☑ Test handshake size and latency with larger keys

Integration and Interoperability

A platform has to work with identity providers (IdPs), MFA, PAM, Identity Governance and Administration (IGA), endpoint management, EDR, XDR, SIEM, SOAR, ITSM, cloud security, DevOps pipelines, Kubernetes, data protection, and sometimes legacy network gear. Integration depth counts for more than the number of integrations, and a log export falls well short of bidirectional risk exchange, automated session revocation, posture-driven access, or policy-as-code. Depth is most critical for platforms that source risk scoring and behavioral analytics externally, where the value depends on how well those integrations close the loop.

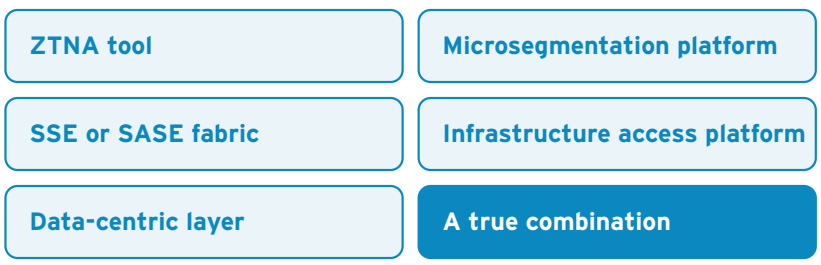
Integration depth over number of integrations



Policy Architecture and Enforcement Scope

Platforms differ widely in what they can enforce. Some products are strong for remote user access and weak on east-west workload traffic. Others segment well but do not touch SaaS, browser, or unmanaged-device scenarios. Establishing whether the product is at heart a ZTNA tool, a microsegmentation platform, a Security Service Edge (SSE) or SASE fabric, an infrastructure access platform, a data-centric layer, or a true combination of several of these prevents unrealistic expectations and badly scoped deployments.

What is the product at heart?



© 2026 KuppingerCole Analysts AG

Pre-deployment Considerations

What are your primary Zero Trust objectives?

Decide whether the immediate goal is microsegmentation, workload connectivity, workload identity, AI agent governance, continuous risk-adaptive authorization, VPN replacement, third-party access, or some combination. A platform ideal for east-west workload segmentation may be a poor fit for browser-based contractor access, and a strong SSE platform may still need a separate segmentation or workload identity solution.

Immediate goal: one or a combination

- Microsegmentation
- Workload connectivity
- Workload identity
- AI agent governance
- Continuous risk-adaptive authorization
- VPN replacement
- Third-party access

Which environments and identities are in scope?

Map the users, devices, applications, workloads, services, APIs, and data that need protection, including managed and Bring Your Own Device (BYOD) endpoints, contractors, administrators, service accounts, cloud workloads, Kubernetes clusters, APIs, OT systems, and AI agents. Identify legacy protocols such as RDP, SSH, SMB, and database access, which drive practical constraints. Treat human access and machine-to-machine connectivity as separate sets of requirements.

What existing tools and controls must be integrated?

Identify which identity, security, and management systems must feed context, receive telemetry, or execute response, and plan for both technical compatibility and operational ownership across IAM, networking, security operations, infrastructure, cloud, and compliance teams.

How mature is your policy and access governance model?

Unclear application ownership, stale identity groups, incomplete device inventories, and unmanaged service accounts surface fast under a Zero Trust platform. Assess readiness for least-privilege design, exception handling, approval workflows, policy lifecycle management, and access reviews, and treat policy simulation, staged enforcement, and rollback as essential for segmentation and workload projects.

How will the platform affect users and operations?

For human access, weigh the experience for employees, contractors, administrators, and developers, and the load on help desk, application owners, and SOC teams. For workload connectivity, weigh how much recurring firewall, Network Address Translation (NAT), routing, and VPN change management the platform removes, since that burden stalls many Zero Trust programs.

What are your requirements for AI and NHIs?

Decide whether the platform must govern AI agents, secure MCP servers, manage workload identities, or authorize service-to-service traffic, and separate marketing and AI-assisted administration from actual enforcement over agents, tools, APIs, and data-sensitive actions.

Questions to Ask Vendors during RFPs

The following questions combine KuppingerCole's standard Zero Trust platform evaluation questions with NetFoundry's responses. The responses are edited for length and neutrality and reflect the company's own account of its platform.

100+
points of presence in the NetFoundry-hosted service

99.95%
service level agreements of up to 99.95%

3
deployment models: hosted, hybrid, self-hosted

0
customer keys held by NetFoundry

What Zero Trust use cases does your platform support out of the box?

NetFoundry focuses on what it calls *Zero Trust Workload Connectivity*: securely connecting machine workloads within and across heterogeneous network boundaries, in contrast to solutions designed primarily to connect human users to applications. The platform supports workload-related use cases including governance for AI agents, LLMs, and MCP servers; microsegmentation down to an individual application or API; a compensating control for unpatchable vulnerable assets; workload-to-workload, machine-to-machine, and service-to-service connectivity; site-to-site connectivity; OT and IoT connectivity; and API access and security. Each workload receives a cryptographic identity and an authorized path to a specific service, with everything else left unreachable and unaddressable on the network. Identity issuance, certificate enrollment, built-in PKI and certificate authority, posture evaluation, microsegmentation, and end-to-end encryption ship as core functions. Once an identity is authenticated and authorized, traffic moves across a private fabric dedicated to the customer, independent of network type, location, overlapping IP ranges, or NAT.

How does your architecture separate control plane, data plane, and enforcement points?

NetFoundry separates the control plane from the data plane. Controllers make and distribute policy decisions and never carry application traffic. The data plane is a distributed mesh of routers that moves traffic between endpoints. Every node, including controllers, routers, and endpoints, authenticates with an X.509 certificate and communicates over mutually authenticated TLS. Encryption runs end to end between the two endpoints, and relaying routers hold no endpoint encryption keys. Endpoints attach through connectors deployed as a host agent, container, virtual machine, or gateway, with no application change required, and SDKs allow the connector to be built into the application itself so that the endpoint exposes no listening port. When policy authorizes two endpoints to communicate, the controller establishes a highly available channel across the fabric, which reroutes around path failures while the session stays up. Enforcement takes place at the edge closest to the workload, and no shared inspection point in the path decrypts customer payloads. The control plane runs as a high-availability cluster, and revoking an identity tears down its connections immediately.

© 2026 KuppingerCole Analysts AG

How do you evaluate context and risk during active sessions?

NetFoundry evaluates context continuously for the life of each connection. On every dial and bind request, and at intervals during the session, the enforcement point evaluates posture signals and revokes access when they stop matching policy. Posture signals for workloads and their hosts include operating system and version, network interface identifiers, domain membership, presence and integrity of specific processes, certificate validity, and time-based multifactor for privileged operations. When a signal falls out of policy, the enforcement point denies the next request or tears down the active connection and returns the specific check that failed. Short-lived, session-bound credentials mean an identity that stops meeting policy loses reachability quickly. NetFoundry consumes risk scoring and behavioral analytics signals from external EDR, XDR, identity, and SIEM/SOAR tooling, which, once integrated, can trigger policy changes, session termination, or re-authentication in real time.

How do you support policy lifecycle management?

NetFoundry is API-first. Every policy object is created, changed, and audited through the management API, which the command-line interface and console also call. Policy can be treated as code, kept in version control, and driven through Terraform and GitOps pipelines. Access is deny-by-default, so a service stays unreachable until a policy grants it, and policy changes require no maintenance window. Administrators can stage a policy against a subset of identities before wider rollout, and every change is recorded for audit. Policies separate the right to use a service from the right to host it, role attributes group identities and services, and delegated administration lets a platform team own the fabric while application teams manage their own services within guardrails.

How do you prevent lateral movement after compromise?

A compromised host can see and reach only what its own identity was already permitted to use. Endpoints and routers run with no open inbound listening ports and reach the fabric through outbound connections, so a compromised host has nothing to scan or discover. Segmentation follows identity and service rather than network topology, subnets, or IP addresses, delivering microsegmentation enforced per connection. The blast radius of a compromised workload is limited to the services that one identity was explicitly granted, and revoking the identity severs those connections at once. Enforcement can be introduced service by service in parallel with existing connectivity, without a disruptive cutover.

How do you protect NHIs and AI agents?

NetFoundry treats every NHI as a workload. A service account, container, CI/CD job, API, MCP server, or AI agent each receives its own cryptographically verifiable identity in place of a shared API key or network-level trust relationship. Identities are issued through built-in PKI, scoped to least privilege, and re-verified on every request. Credentials are certificate-based and short-lived, which limits the window a leaked secret remains usable and reduces long-lived static secrets. Private keys are stored in a secure enclave and do not leave the endpoint. For AI, agents and MCP servers connect outward to authorized resources and expose no inbound port, so an unauthorized agent does not discover or reach a protected resource. Policy governs which MCP servers an agent can reach, and enforcement through connectors and SDKs scopes an agent to the specific APIs and data it needs, isolating agent-to-agent and agent-to-tool traffic.

How do you integrate with our existing identity, endpoint, cloud, security operations, and ITSM tools?

NetFoundry integrates with existing identity, endpoint, cloud, and operations tooling through open standards and an API-driven management plane. It accepts any OAuth- or OIDC-compliant identity provider and uses X.509 certificates from external certificate authorities for mutual TLS once those authorities are registered with the controller. Clients enroll through OIDC or JWT-based enrollment and certificate signing requests. Posture checks and multifactor authentication are built in and layer on top of the existing provider. Every session carries an identity rather than an IP address alone, and identity-correlated telemetry is exported through open APIs to a buyer's SIEM, SOAR, or analytics tooling. Endpoint coverage includes connectors for Windows, macOS, Linux, Android, and iOS deployed as host agent, container, virtual machine, or gateway, with SDKs and Kubernetes-native deployment, and the same components run in AWS, Azure, and Google Cloud. Connectivity and policy actions run through the same API, so they integrate with ServiceNow, Jira, and other automation, ticketing, and provisioning pipelines.

What deployment options do you support, and where is customer data stored?

NetFoundry runs the same software in three models. The NetFoundry-hosted service provides a private, dedicated fabric across more than 100 points of presence that NetFoundry hosts and manages, with service level agreements of up to 99.95%. A hybrid model places controllers and routers in the organization's own sites while NetFoundry manages the software. A fully self-hosted model keeps everything inside the organization's boundary, including air-gapped and sovereign sites. Under the hybrid and self-hosted models, application traffic and control data stay in the sites the organization chooses. Each fabric is private and dedicated to one customer, so tenant isolation is a property of the architecture. Endpoints generate a fresh end-to-end encryption key per session and hold it locally, and compatible devices store it in a hardware security module. NetFoundry holds no customer keys, including in the managed service.

Same software, three deployment models

NetFoundry-hosted
Private, dedicated fabric across more than 100 points of presence, service level agreements of up to 99.95%

Hybrid
Controllers and routers in your own sites, software managed by NetFoundry

Self-hosted
Everything inside your boundary, including air-gapped and sovereign sites

How does your platform support crypto-agility, including the ability to change cryptographic algorithms and rotate credentials at scale, and what is your path to post-quantum readiness?

Changing the cipher across every node is a software and policy change rather than a hardware refresh. Encryption ciphers are pluggable: the platform uses libsodium by default, offers FIPS-compliant ciphers as an option, and can integrate other

© 2026 KuppingerCole Analysts AG

cryptographic engines, so the cipher in use changes without re-architecting the network. The platform is software-based and built on certificate-based PKI with short-lived, automatically rotated credentials, which lets an organization re-key identities at scale without manual certificate handling. NetFoundry ships no proprietary post-quantum algorithm today and tracks the NIST Post-Quantum Cryptography (PQC) standards to adopt hybrid and PQC key exchange as the ecosystem matures.

How does the platform perform and scale for high-volume machine-to-machine traffic?

Adding workloads and identities does not require re-architecting the underlying network. Traffic flows across a distributed mesh fabric rather than a single cloud inspection point, and the hosted fabric adds high availability, load balancing, and auto-scaling across its points of presence. Each connection is scoped to one application, so growth adds connections without adding architectural complexity.

Who can decrypt traffic traversing your platform, and where are encryption keys held?

NetFoundry uses end-to-end encryption and holds none of the customer's keys, so it has no means to decrypt traffic crossing the fabric. Encryption is established and terminated at the authorized endpoints, and relaying routers, including those in the NetFoundry-hosted service, do not possess endpoint session keys. Policy enforcement happens at the edge, so traffic is not diverted to NetFoundry or a third party for inspection. Keys are generated on the endpoint and, on compatible devices, stored in a hardware security module.

Can we speak to reference customers with similar architecture, scale, and regulatory requirements?

Yes. NetFoundry matches each reference to the buyer's architecture, scale, and regulatory profile. Production deployments include regulated and complex environments, including healthcare data interoperability, large-scale physical security and surveillance fleets, connected IoT fleets, insurance and financial services, integration platform and B2B connectivity, and private 5G for defense and government, across architecture patterns such as application-embedded connectivity, gateway-based segmentation, multi-tenant B2B, and OT and edge deployments.

© 2026 KuppingerCole Analysts AG

Recommendations

Most organizations already own the individual controls and lack the architecture that makes them coherent. The following recommendations apply broadly, with particular weight for buyers whose priority is workload and machine-to-machine connectivity, including the AI agents and automated pipelines that generate a growing share of that traffic.

1. **Name the primary outcome before looking at products.**
   Separate human-to-application access from workload-to-workload connectivity; the two weight the eight capabilities differently.

2. **Make segmentation and machine identity the center of a workload project.**
   Favor platforms that enforce identity-based segmentation independent of topology, contain blast radius per connection, and issue short-lived, certificate-based identities to workloads, services, APIs, and agents. Confirm that policy and identity follow workloads through create, scale, move, and retire events without manual re-registration.

3. **Examine the data plane as well as the control plane.**
   Establish where traffic flows, whether a cloud control plane forces a cloud data path, who can decrypt traffic, and what happens to internal service-to-service communication if the provider degrades. For high-frequency machine traffic, weigh key custody, path locality, latency, and failure behavior against real workload patterns. Extend the same scrutiny to cryptographic longevity: whether algorithms can be replaced through a software and policy change, whether credentials rotate automatically at scale, and what the path to post-quantum key exchange is.

4. **Match threat prevention to the traffic type.**
   Content and TLS inspection suit human-facing risk but map poorly to authorized, mutually authenticated service traffic, and they can force a third party to hold decryption keys. For workload flows, value authenticated reachability, deny-by-default connectivity, and call-pattern detection over payload inspection.

5. **Weight data-centric controls to the use case.**
   DLP, DRM, and browser controls govern human actions inside a session and do little for automated flows. Where they are essential, verify that they are built in or maturely integrated.

6. **Confirm where the analytics come from.**
   Some platforms provide behavioral analytics and risk scoring in the product, while others rely on external SIEM, SOAR, and identity tooling by design. Either can work, provided the required signals and integrations exist and run in both directions.

7. **Adopt incrementally.**
   Start with well-scoped use cases, validate integrations, map dependencies, simulate policies, and expand enforcement in stages. Enforcing every policy everywhere on day one turns Zero Trust into an expensive denial-of-service attack against the business.

A well-chosen platform moves an organization from isolated controls to trust decisions that are consistent, contextual, and enforceable across users, devices, workloads, services, APIs, and AI systems. Architectural fit, integration depth, and operational safety are more important than feature count. Zero Trust cannot be bought and switched on. The right platform provides the means to operate it at enterprise scale, but the policy model, its ownership, and the discipline to keep both current remain the organization's responsibility.

© 2026 KuppingerCole Analysts AG

Contact

KuppingerCole
Analysts AG

clients@kuppingercole.com

kuppingercole.com

Related Research

Leadership Compass	Zero Trust Platforms
	Non-Human Identity Management
Advisory Note	From Machine Identity to Agentic AI
	Modernizing Authorization: Policy Based Access Management
	Crypto-Agility as an Operational Capability
Whitepaper	From AI Agents to Trusted Digital Workers
Blog Post	Agentic AI and Data Access Control as the New Security Perimeter
	The Last Algorithm: Crypto-Agility, PKI, and the End of Static Identity
Analyst Chat	Defining Identity: Beyond Secrets, Credentials, and Accounts
	AI Escaped the Sandbox
Rising Star	NetFoundry

© 2026 KuppingerCole Analysts AG

Copyright

©2026 KuppingerCole Analysts AG. All rights reserved. Reproducing or distributing this publication in any form is prohibited without prior written permission. The conclusions, recommendations, and predictions in this document reflect KuppingerCole Analysts' initial views. As we gather more information and conduct more profound analysis, the positions presented here may undergo refinement or significant changes. KuppingerCole Analysts disclaims all warranties regarding the completeness, accuracy, and adequacy of this information. Although KuppingerCole Analysts research documents may discuss legal issues related to information security and technology, we do not provide legal services or advice, and our publications should not be used as such. KuppingerCole Analysts assumes no liability for errors or inadequacies in the information contained in this document. Any expressed opinion may change without notice. All product and company names are trademarks™ or registered® trademarks of their respective holders. Their use does not imply any affiliation with or endorsement by them.

KuppingerCole Analysts supports IT professionals with exceptional expertise to define IT strategies and make relevant decisions. As a leading analyst firm, KuppingerCole Analysts offers firsthand, vendor-neutral information. Our services enable you to make decisions crucial to your business with confidence and security.

Founded in 2004, KuppingerCole Analysts is a global, independent analyst organization headquartered in Europe. We specialize in providing vendor-neutral advice, expertise, thought leadership, and practical relevance in Cybersecurity, Digital Identity & IAM (Identity and Access Management), Cloud Risk and Security, and Artificial Intelligence, as well as technologies enabling Digital Transformation. We assist companies, corporate users, integrators, and software manufacturers to address both tactical and strategic challenges by making better decisions for their business success. Balancing immediate implementation with long-term viability is central to our philosophy.

For further information, please contact clients@kuppingercole.com.