Special Report

Zero Trust Security Strategy: System

Byung-gwon Hwang, SK Shieldus

■ Overview of the System Pillar

In the context of Zero Trust Architecture, the System Pillar encompasses all servers responsible for operating critical applications or storing and managing sensitive data. This domain includes not only physical and virtual servers, but also virtual machines running on hypervisors, databases, file servers, database servers, container and Kubernetes nodes, as well as public cloud instances—all of which fall within the scope of the System Pillar.

When applying Zero Trust Architecture to the System Pillar, the foremost consideration is the diverse range of system (server) operational environments. Unlike the past, when systems were predominantly on-premises, today's environments have expanded to include public, hybrid, and private clouds. Servers are now rapidly created, modified, and migrated at the level of virtual machines and containers. As the operational landscape grows more heterogeneous, the number of management items in the system domain—such as accounts, access paths, configurations, patches, and backups—increases, and these elements are often managed in a fragmented manner rather than through integrated processes. Therefore, it is imperative to standardize management policies for the System Pillar and to establish a consistent and unified management framework in conjunction with related systems.

Figure 1. Diverse System (Server) Operating Environments

In a Zero Trust environment, the significance of the System Pillar lies not merely in enhancing the security of individual servers, but in managing servers operating across heterogeneous environments according to unified standards. Given the diversity of operating systems and middleware, as well as the coexistence of physical servers, virtual machines, and containers, it is inherently challenging to implement distinct security policies for each server. Thus, the central imperative is to establish an integrated management framework grounded in centrally defined policies.

Unlike traditional approaches that assume "internal servers can be trusted," the System Pillar's methodology does not rely on such a presumption. Even after access is granted, accounts, sessions, commands, queries, and modification activities are continuously logged and monitored; privileges are assigned only for the necessary period and scope, and are automatically revoked upon expiration. The status of each server—such as patch levels, configuration compliance, vulnerability assessments, and backup verification—must also be evaluated in conjunction with account and privilege management, verifying both the user's identity and the system's current security posture.

Within Zero Trust Architecture, the System Pillar is not limited to the role of servers alone, but functions as the foundational space in which an organization's most critical resources reside. Accordingly, it is imperative to implement mechanisms that can identify and collectively manage systems deployed across a wide range of environments. The core elements and systems comprising the System Pillar, as outlined below, serve as essential reference points for establishing a robust Zero Trust environment.

■ Key Elements of the System Pillar

Within Zero Trust Architecture, the System Pillar serves as the central axis for the direct management and protection of all systems comprising an organization's core assets—including servers, critical applications, and data repositories. Systems distributed across diverse environments—on-premises, cloud, and hybrid—constitute the primary aggregation points for information and business operations within today's complex IT infrastructures, while simultaneously representing prime targets for both external and internal threats.

Notably, in a Zero Trust paradigm, trust based on the singular identity or physical location of a system is no longer valid; instead, continuous and granular verification and integrated management must be enforced across all systems, as well as accounts, resources, logs, and processes residing within them. Only through the synergistic integration of various administrative and technical elements—such as system inventory, account management, access control, security policies, patch management, vulnerability management, visibility, system segmentation, and policy administration—can organizations achieve genuine security levels that encompass data protection, operational continuity, and legal compliance across the enterprise.

The following section outlines the principal elements of the System Pillar and details specific management and technical measures required for their implementation, structured according to the Zero Trust maturity model.

1. System Asset Inventory

In a Zero Trust environment, system inventory management constitutes the foundational step in physically and logically identifying all core systems—such as servers, critical applications, and data repositories—operated within an organization, and ensuring that their status is continuously updated. This encompasses a broad spectrum of systems, including not only on-premises but also cloud and hybrid infrastructures: physical and virtual servers, containers, databases, and file servers. All such systems must be centrally registered and catalogued within an integrated asset management framework. Rather than relying on one-time registration at deployment, it is essential that key attributes (such as system owner, IP address, operating system, role, and configuration location) and state information are dynamically updated in real time throughout the entire system lifecycle—covering addition, modification, migration, and decommissioning—which forms the basis for policy automation.

Every asset within the system inventory should be logically grouped by operational environment (on-premises, cloud, hybrid) and functional role (e.g., web server, database server, file server). Grouping information must not be confined to static documentation or ad hoc data entry; instead, integrated asset management systems and monitoring tools must automatically reflect any changes, reclassify assets, and update group policies in real time as system changes occur. This enables unified management of security policies, access permissions, and monitoring frameworks for each group, and allows for the immediate identification and response to policy violations or anomalous activity.

Furthermore, the definition of system zones must transcend simple physical or logical segmentation by enabling multi-layered management according to business purpose, data criticality, network topology, and required security posture. Zone-specific controls should include differentiated access policies, micro-segmentation, session-based multi-factor authentication (MFA), real-time risk assessment, and policy automation. Traffic flows and access rights—both between and within zones—must be continuously and automatically adjusted via integration with asset management systems, ICAM (Identity, Credential, and Access Management), and unified monitoring tools, thereby ensuring real-time visibility.

In an optimized system inventory management framework, all changes in the status of system assets are reflected instantaneously. This real-time information underpins granular privilege control, efficient policy deployment, rapid anomaly detection and incident response, as well as systematic auditing and compliance management, thereby establishing the essential foundation for Zero Trust implementation.

2. System Account Management

In a Zero Trust environment, system account management entails the comprehensive cataloging and unified oversight of all accounts with access to organizational servers (including Unix, Linux, Windows, and others) and critical systems, based on their respective purposes and functions. Administrators must systematically manage not only privileged accounts, but also user-level and service accounts, classifying each by key attributes—such as usage status, privilege level, and group affiliation—and overseeing their entire lifecycle, including modification and decommissioning.

It is insufficient to rely on manual documentation for scattered account information across different systems. Instead, all account data must be centrally managed and updated in real time through an account management system or portal. Essential attributes for each account—such as privileges, affiliation, expiration, and lock status—should be automatically reflected and updated. Any changes in account status (creation, modification, deletion) must be immediately scrutinized for anomalies, with unauthorized or high-risk accounts promptly deactivated or otherwise remediated.

Account management must go beyond mere inventorying, enabling both manual and automated classification and grouping of accounts by criteria such as criticality, privileges, group membership, and usage status. The resulting categorized account information should be integrated with relevant systems to simultaneously enhance availability and security across the infrastructure.

Securing system accounts is paramount. To prevent unauthorized access and account misuse, security settings for each account—including access restrictions, expiration, and least privilege—must be consistently enforced, whether natively or through integration with account management systems or ICAM (Identity, Credential, and Access Management) platforms. When accounts are created or deleted, pre-defined security policies—tailored by operating system (Linux, Windows, macOS, etc.)—should be automatically applied. Security configurations must be linked with unified monitoring and log analysis systems to support real-time monitoring and auditing.

For password management, each account should adhere to stringent password policies (such as minimum length, complexity requirements, and regular rotation) and multi-factor authentication (MFA) should be enforced for high-value accounts. Password status, policy compliance history, and change records must be centrally managed via integration with ICAM, authentication, and monitoring systems, while all related activities are logged. Furthermore, linkage with SIEM/SOAR platforms should enable immediate detection and remediation of at-risk accounts or anomalous password changes.

3. System Access Control

System access control, in accordance with Zero Trust principles, mandates that all permissions be granted based on the principle of least privilege, with granular access rights configured for each system according to its specific function and operational requirements—referencing the system inventory as the authoritative baseline. Access control must encompass a comprehensive range of components, including authentication, authorization, and access management, and should not be limited to controls within individual systems. Instead, enterprise-wide privilege management should be achieved by leveraging integrated management systems such as ICAM, in conjunction with network and application layers.

Each system should enforce access restrictions based on various criteria—such as IP addresses, ports, and accounts—through access rights settings. Where appropriate, integration with SSO (Single Sign-On) and IAM (Identity and Access Management) solutions should facilitate the assignment of granular permissions using RBAC (Role-Based Access Control) or ABAC (Attribute-Based Access Control) models. For real-time monitoring and analysis, access permissions must be managed dynamically via integration with ICAM, SIEM (Security Information and Event Management), SOAR (Security Orchestration, Automation, and Response), and XDR (Extended Detection and Response) systems, ensuring that privileges can be automatically adjusted in response to emerging threats.

Command control within systems involves identifying and managing high-risk or vulnerable commands used in actual operations. Robust policies must be established to specify which commands require control and auditing. Rather than relying solely on per-system settings or shell-based restrictions, command control should be systematically applied through access control platforms, Secure OS, or unified management systems, supporting comprehensive change tracking and real-time monitoring. The usage patterns and anomalies associated with controlled commands should be analyzed and remediated in real time via integration with SIEM, SOAR, and similar platforms, with ongoing verification of the effectiveness and timeliness of these controls.

Real-time session control entails managing the granting and revocation of system access on a per-session basis. Access control systems must monitor and manage the entire lifecycle of each session—including initiation, maintenance, extension, and termination—in real time. Upon detecting anomalous activity, immediate measures such as session termination or additional authentication must be enforced. Rather than relying exclusively on individual system configurations, session policies should be finely tailored per account, group, or business function, through close integration with account and access control systems. Real-time session data should also be linked with unified monitoring and analytics platforms to elevate the overall security posture.

4. System Security

In a Zero Trust environment, system security management entails defining security policies for critical systems such as servers and establishing an effective governance framework to enforce them. Security policies must be explicitly specified for each system group or asset, and enforcement should extend beyond native system security configurations to include integration with asset management platforms, unified monitoring systems, and other centralized controls. Whenever systems are added or modified, security settings should be automatically applied according to predefined policies, and real-time monitoring must enable immediate response to any changes in system status.

For system components and critical data, regular backup and recovery mechanisms must be maintained. To protect against risks such as hardware failures, software errors, or intrusions, backup systems should periodically capture essential configuration files, databases, logs, and other critical information. Recovery plans should be in place to ensure rapid restoration in case of incidents. Redundant configurations and disaster recovery (DR) centers should be employed to facilitate swift recovery during catastrophic events, and backup and recovery policies must be automatically enforced whenever system changes or anomalies occur.

Internal system processes—including creation, execution, monitoring, and termination—must be systematically managed. Critical or high-risk processes should be clearly defined for each system, with execution controlled based on privileges. Monitoring systems should observe the real-time status of designated processes, and any abnormal termination or unexpected behavior must trigger immediate investigation and corrective action. Process anomalies should be managed through automated mechanisms, such as alerts, to ensure rapid response.

Regarding system security functions, regular checks should be conducted on areas including software updates, integrity of critical files, antivirus and malware defenses, and log health. Rather than relying on manual checklists, results should be continuously visualized through integration with antivirus solutions, inspection systems, and vulnerability management platforms. Any detected issues must be addressed immediately. Inspection criteria and items should be periodically updated and applied via automated tools such as monitoring systems and machine learning, and the results should be systematically documented in reports or other formats for ongoing review and compliance.

5. System Segmentation

System segmentation refers to the practice of managing specific servers and critical systems through physical or logical separation based on the system inventory. Segmentation can be implemented via physical infrastructure modifications, network isolation, virtualization (VMs), or access control policies for logical separation, and should be applied according to each system's role, criticality, and service characteristics. When implementing segmentation, designs must maintain compatibility with existing systems while allowing flexible expansion as new systems are introduced. Additionally, dedicated monitoring, surveillance, and security controls must be established for segmented systems. Integration with asset management systems should ensure that when new systems are added to the segmentation scope, labeling policies are automatically inherited and both logical and physical separation are systematically maintained.

For highly critical systems, more granular measures are required to ensure effective management and protection. Various administrative and technical controls—including incident management, change management, patch management, and backup and recovery—should be applied in accordance with the segmentation policy. Services such as web servers and database instances should be physically or logically separated and managed systematically. Changes in the status of critical systems, as well as anomalies, should be continuously monitored in real time, with automated response mechanisms in place where necessary. When new systems are added or the environment changes, detailed management policies and technical measures should be automatically applied to preserve the overall security posture through a dedicated management framework.

6. System Policy Management

System policy management refers to the establishment and consistent enforcement of administrative policies designed to ensure the secure and efficient operation of system environments. Management policies for systems should encompass a wide range of functions, including system operations, access control, security, documentation, reporting, and analysis, while also satisfying non-functional requirements such as accuracy, completeness, consistency, user convenience, scalability, and security. When developing policies, considerations must include compatibility with existing systems, relevant legal and regulatory compliance requirements, and internal standards. Policies should be systematically documented and managed, drawing on corporate guidelines and standard frameworks. Utilizing centralized management systems, policies must be applied consistently across all systems, with integration into monitoring platforms to allow automatic updates based on operational analysis and seamless application to new systems.

Exception management is also essential in system policy administration. Servers requiring exception policies typically include those with unique functions, systems used for testing new technologies or features, and systems handling critical data. A separate exception policy must be established, detailing the list of exempted servers, prioritization, and monitoring and reporting procedures. Systems requiring exceptions should be systematically cataloged and managed through either manual processes or centralized management systems. Integration with monitoring tools must ensure that exception-related items are reflected in real time, allowing immediate response to policy violations or anomalous activity.

In a Zero Trust framework, system policies are not static. As system environments continuously evolve, policies must undergo ongoing evaluation, modification, training, and documentation updates. Analysis systems should be leveraged to assess potential risks associated with policies, while automated policy generation ensures that changes are propagated and enforced across systems without manual intervention. The maturity of a policy management framework is determined by the extent to which these continuous management and automation processes are implemented, thereby enhancing both organizational security posture and operational efficiency.

Similarly, in a Zero Trust environment, network segmentation strategies extend beyond simple physical boundaries. By combining granular access controls tailored to diverse business environments and asset characteristics with automated policy enforcement, organizations can minimize internal risks, prevent lateral movement, and simultaneously achieve a flexible and resilient security environment.

7. System Patch Management

A system patch management policy must explicitly define the standards and procedures for applying security patches to all system components, including operating systems, applications, and firmware. The policy should cover the entire patch lifecycle, including the selection of target systems, patch deployment and installation procedures, backup and recovery measures in case of patch failure, and real-time monitoring of patch compliance. Management should be conducted consistently and systematically through centralized management systems or Patch Management Systems (PMS), ensuring that integration with external patch servers allows immediate reflection of the latest patch policies and continuous maintenance of systems in a secure, up-to-date state.

Patch deployment and execution must be applied accurately and consistently across all servers and systems in accordance with the management policy. All stages of patch deployment—such as patch listing, prioritization, distribution and installation, and pre-deployment functional testing—should be standardized and automated. Approved deployment tools (e.g., PMS) must deliver patch files to target systems, with backup and recovery mechanisms enabling rapid rollback in the event of failure. Newly released patches should first be validated in isolated environments, such as sandboxes, before deployment; any issues detected in the sandbox should trigger automatic exception handling according to PMS policies. These procedures ensure both operational stability and user convenience.

System patch monitoring involves real-time oversight of the entire patch deployment process, enabling immediate detection of installation status, omissions, failures, or delays. Utilizing PMS, integrated monitoring systems, and analytics tools, organizations should visualize patch status in real time, track patch adoption trends, identify causes of failures, and manage follow-up actions such as redeployment and automatic rollback. Monitoring outputs should be automatically generated and distributed in report formats, with instant alerts and analysis enabling automated remediation processes to maintain system security and compliance.

8. System Log Management

System log management requires clearly defining which logs should be collected for each system and establishing a framework for real-time collection and storage according to a formal log collection policy. Collection methods may include built-in system tools, log agents, and custom scripts, ensuring comprehensive capture of all necessary logs, such as custom application logs and audit logs. Real-time and periodic collection targets should be managed separately to optimize efficiency. A centralized environment must be established to enable enterprise-wide real-time collection and integration of system logs, while ensuring data integrity and security through secure storage, transmission, and retention practices.

An effective log management framework must include a robust indexing system. Real-time indexing, search, filtering, pagination, highlighting, and visualization capabilities are essential. Tools such as Splunk, Elasticsearch, or Graylog can be employed to assign and manage index values for critical logs, while continuously improving the overall log management process. Integration with log collection and analysis systems enables real-time monitoring and response capabilities.

System log analysis should provide actionable insights into system activity, performance, and security posture through real-time, correlation, and visual analysis. Using centralized log systems and SIEM platforms, diverse log sources can be correlated to identify patterns, inform system improvements, and anticipate issues such as errors, security threats, or performance degradation using historical data and machine learning. Automated response mechanisms can also be incorporated to mitigate risks proactively.

Within the overall log management framework, periodic automated generation and management of summary, detailed, and comparative reports are necessary to maintain comprehensive visibility into system health. Reports should be deliverable in multiple formats, including HTML, PDF, and CSV, with scheduled distribution. Applying machine learning–based automated analysis allows for real-time detection of critical events and risk factors, enabling immediate remediation. Continuous refinement of the report management process ensures rapid and effective operational response.

9. System Vulnerability Management

A system vulnerability management policy must clearly define the objectives and scope of vulnerability management to maintain system security levels and ensure regulatory compliance. The policy should establish a systematic process covering the entire lifecycle of vulnerability management, including identification, remediation, analysis, and reporting. It should specify the definition of vulnerabilities, severity assessment criteria, diagnostic methods, patch deployment or code remediation procedures, mitigation strategies, and reporting and management protocols. To maintain currency, the policy must be continuously updated and automatically applied across all systems through integration with SOCs, threat intelligence (TI) platforms, and other sources to collect the latest vulnerability information and reflect updates in real time.

Vulnerability detection and remediation should employ both automated and manual scanning, as well as publicly available vulnerability databases such as CVEs, to rapidly and accurately identify weaknesses. Identified vulnerabilities must be prioritized according to policy, and swift remediation—such as patch deployment, code modification, or mitigation measures—must be implemented. Periodic assessments and the integration of the latest vulnerability information into the system ensure consistent coverage, while real-time diagnostics and automated patching should be applied according to risk levels.

Impact assessment of vulnerabilities involves analyzing the root cause and evaluating their effects on the system from multiple perspectives to determine priority. Factors such as exploitability, operational impact, and the necessity for preventive or mitigating measures must be objectively assessed. Based on this evaluation, appropriate responses—including patching, mitigation, or acceptance—should be defined. Integration with ICAM, TI systems, and real-time database feeds facilitates automated analysis and policy enforcement.

Vulnerability management extends beyond mere detection and remediation. Integration with SIEM and other monitoring systems enables real-time tracking and analysis of vulnerability events. All management records, including vulnerability reports, remediation status, and mitigation plans, should be documented through automation tools. Deep analysis using machine learning and big data should be employed to continuously improve both the effectiveness and efficiency of responses. Moreover, integration with antivirus, monitoring, and vulnerability management systems ensures that automated response processes are triggered during risk events, with reports and status updates generated and distributed on a scheduled basis.

10. System Visibility and Analytics

Ensuring system visibility involves establishing a framework for real-time monitoring and analysis of server status, performance, anomalous activities, and security threats, enabling early detection and rapid response to issues. Monitoring policies should be applied across all systems, continuously collecting and analyzing key metrics such as CPU, memory, and disk utilization, critical process states, and other system indicators, while implementing mechanisms for anomaly detection and alerting. Monitoring systems should provide not only system-level visibility but also comprehensive insights across the entire infrastructure. Any changes in system states must be automatically reflected in the monitoring framework, with integration into centralized log and analytics platforms to ensure consistent operations and management.

System analytics capabilities are central to achieving deep understanding and optimization of the system environment through in-depth analysis of data collected from servers. Beyond simple data collection, logs and other system data should be visualized in real time or stored for long-term analysis to identify issues and derive improvement measures. Collected data must undergo cleansing, transformation, and integration, enabling examination through a variety of analytical techniques. Insights from monitoring and analytics processes should feed directly into operational and security policy adjustments, ensuring real-time improvements and enforcement. This framework should be continuously refined through regular reviews and process improvement activities to enhance overall effectiveness and responsiveness.

11. Policies and Processes

System operational procedures constitute a critical component of an organization's IT system management, playing a vital role in achieving both business continuity and system security through efficient and stable operations. Server operational procedures must provide clear and consistent standards across key areas, including system installation, configuration, monitoring, maintenance, and documentation, and should be established and managed from a governance perspective. From a Zero Trust standpoint, operational procedures must embed security-enhancing principles such as least privilege, continuous authentication and authorization, decoupling security from network location, data-centric protection, and rapid incident response. Beyond traditional perimeter-based models, these procedures should incorporate real-time feedback and enforcement mechanisms.

Minimum privilege management is a core principle. Access should be restricted to the smallest necessary group, and all unauthorized access must be proactively blocked. Users should be granted only the minimum permissions required to perform their tasks, implemented through mechanisms such as RBAC (Role-Based Access Control) and ABAC (Attribute-Based Access Control). In accordance with least privilege principles, permission management must be integrated with account and authentication management systems, with procedures for revoking privileges and approval workflows established to prevent misuse across the system. High-value information systems should employ additional layers of control, such as isolated environments or dedicated equipment, to establish multi-tiered defense mechanisms.

Management of personal data systems is equally important. Systems storing personal information must implement both administrative and technical privacy protection policies, aligned with applicable laws and compliance standards. Integration with privacy management systems and portals enables centralized oversight of all systems containing personal data, with automated lifecycle management to monitor and control data flow. Policy development should encompass personal data lifecycle management, flow tracking, and access controls to minimize risks of leakage or misuse.

Building on these elements, the System Pillar functions as the central axis for the direct management and protection of all organizational core assets—including servers, critical applications, and data repositories—within a Zero Trust Architecture. In today's complex and distributed IT infrastructure, systems not only handle the majority of operational data and business processes, but also represent high-impact targets in the event of security incidents, necessitating robust protection against both external and internal threats.

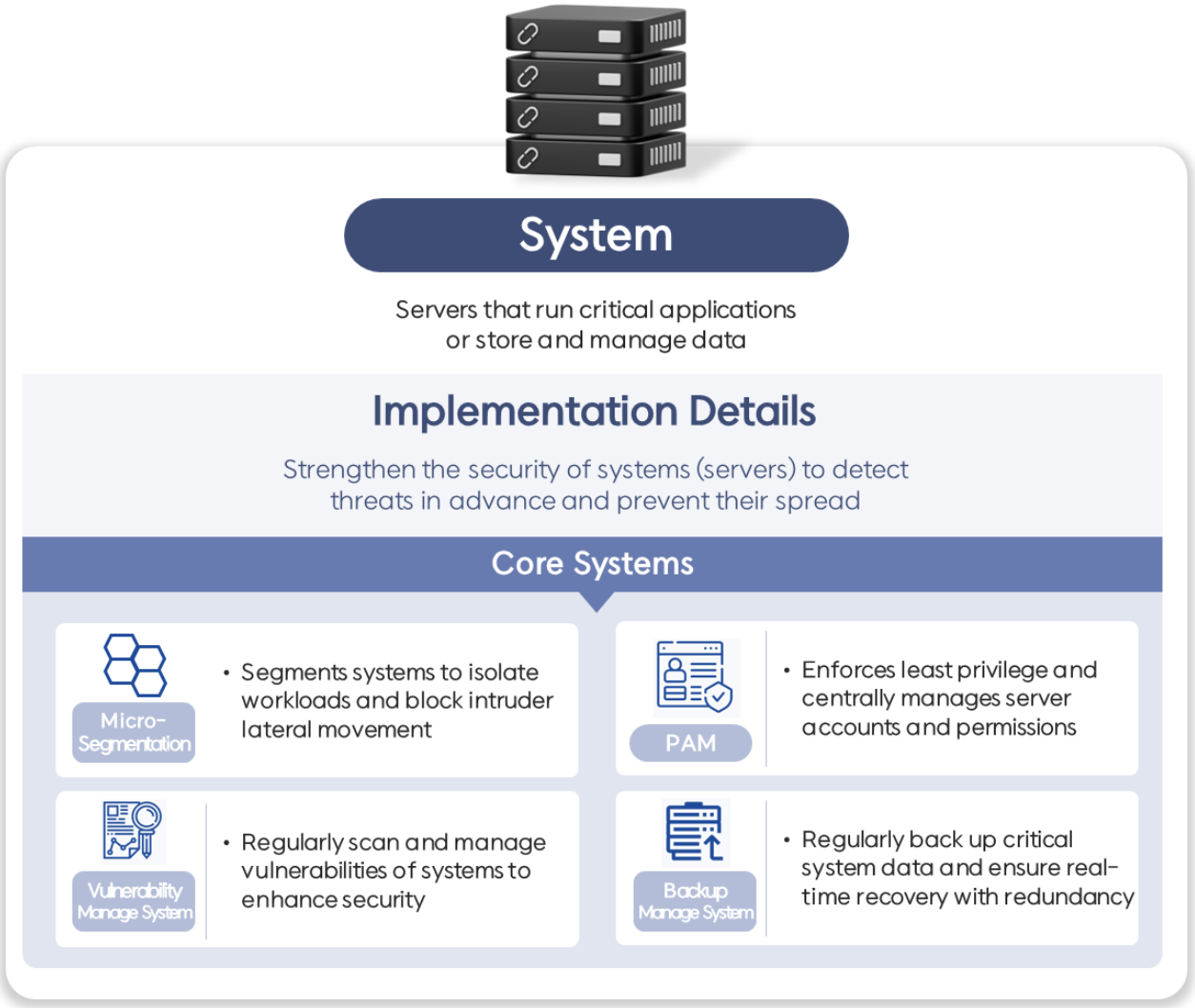
In a Zero Trust environment, access control based on server location or pre-existing trust relationships is no longer sufficient. Each system must implement strong individual authentication, least privilege principles, real-time security monitoring, granular access control, and process management, with these mechanisms interlinked and mutually reinforcing. Additionally, integrated operational and technical controls—including system inventory, account management, security policies, vulnerability and patch management, system segmentation and policy automation, log collection and analysis, and real-time visibility—are required to achieve a substantive Zero Trust security posture across all systems.

The advanced implementation of the System Pillar establishes a management framework and technical foundation that consistently applies Zero Trust principles across the organization's server infrastructure. This enables early detection of anomalies at the system level and rapid response to potential threats. Effective deployment of the System Pillar is essential for securely protecting critical data and core business processes, while safeguarding organizational infrastructure against evolving IT environments and increasingly sophisticated cyber threats.

■ Implementation of Zero Trust Functions for Key Systems

To successfully implement a Zero Trust environment, both technical measures and the systems capable of executing them are essential. Zero Trust Architecture is founded on the principle of "never trust, always verify," and achieving this requires systems that can continuously assess the state of each system, perform ongoing verification, and enforce least-privilege access.

The key systems outlined below play critical roles within a Zero Trust environment and are interconnected to strengthen the organization's overall security posture. For each system, we examine the specific functions necessary to implement Zero Trust principles and the security benefits that organizations can derive from their effective deployment.



* Source: SK Shieldus, "The Beginning of Zero Trust: Realized with SKZT"

Figure 2. Key Systems within the System Pillar

1. PAM (Privileged Access Management)

In the System Pillar, Privileged Access Management (PAM) serves as the central security system. Integrated with the Identity Pillar's IAM (IDP), PAM ultimately enforces and audits "who can access what, from where, when, and to what extent" within a Zero Trust environment. Modern PAM solutions extend beyond traditional server- and database-centric controls to cover enterprise applications, network and security devices, cloud consoles (AWS, Azure, GCP), and various SaaS platforms, allowing comprehensive privileged access management from a single interface. In other words, PAM uniformly governs system access control and database access control while applying consistent principles and procedures across remote access, cloud management consoles, and web- or API-based administration interfaces.

Traditional PAM typically focused on system and database access control, using installed agents to monitor and record sessions. With the expansion of managed environments to applications, SaaS, and cloud infrastructures, web-based architectures have rapidly gained adoption. In this model, a bastion (proxy gateway) intermediates all sessions, allowing users to access required resources via web consoles under least-privilege policies without local keys or accounts. This approach enables consistent control over assets where agent installation is impractical, as well as externally managed services, making it highly favored in operational environments.

Continuous verification—a core principle of Zero Trust—is implemented in PAM through real-time reassessment of privileged sessions. Even after session initiation, signals such as user and device status, access location and time, executed commands, and query patterns are continuously evaluated. Upon detecting risk, PAM can require additional MFA, progressively reduce privileges, or automatically terminate the session. This ongoing verification applies uniformly across servers, databases, and SaaS platforms.

Secure channels and sensitive information management are also fundamental to PAM. SSH keys and privileged passwords are stored and rotated in a secret vault, with access proxied through SSH/SSL/TLS tunnels to avoid key exposure. All activities—including console access, commands, file uploads/downloads, queries, and data extraction—are logged with detailed metadata, supporting incident reconstruction and regulatory audits.

In summary, PAM acts as the single gateway for privileged access within the System Pillar. While IAM (IDP) authenticates “who” the user is, PAM determines and enforces “what, how far, and under what conditions” access is permitted, recording and auditing the results. In a Zero Trust environment, PAM operates as an integrated system across servers and databases via agents/proxies, and across applications and SaaS platforms via web-based proxies, unifying management across both cloud and on-premises infrastructures.

2. Micro-Segmentation

Micro-Segmentation is an advanced security strategy that offers a finer-grained approach compared to traditional macro-segmentation. It separates the network at the OSI Layer 7 (Application layer) level, down to the granularity of business functions, users, and applications, enforcing access controls based on the principle of least privilege.

Where conventional network segmentation primarily relies on physical or logical boundaries such as IP addresses, ports, or VLANs, Micro-Segmentation focuses on the relationships between services and applications, their purposes, and actual traffic flows. This logical division allows organizations to precisely control internal threats and prevent lateral movement by attackers within the network.

Implementation of Micro-Segmentation can be categorized into two approaches: network-based and system (host)-based. Within the System Pillar, Micro-Segmentation is primarily system-based. System-based Micro-Segmentation deploys either agent or agentless solutions on endpoints such as servers or workstations, applying granular security policies and access controls at the individual system level. During this process, the topology between systems and the network is visualized, and actual network traffic flows between applications and services are analyzed to automatically generate and manage segmentation policies. Recent advancements incorporate AI and machine learning to optimize system-to-system paths, detect anomalies, and recommend policy adjustments, enhancing operational efficiency.

The core principle of implementing Micro-Segmentation in the System Pillar is shifting from "zone-level firewalls" to "per-system firewalls." Historically, firewalls were deployed at critical network segments to block major traffic flows. Micro-Segmentation, however, applies policies as if each server has its own firewall, ensuring that lateral movement is blocked even if a breach occurs within the network. While this granular segmentation significantly improves security, it also increases policy complexity. AI and machine learning technologies are leveraged to mitigate this complexity through functions such as learning normal traffic patterns, policy recommendations, consolidation of redundant or unnecessary rules, pre-change simulation, and automated alerts for anomalies. Real-world implementations have demonstrated that AI-enabled Micro-Segmentation solutions effectively enhance policy enforcement and operational management.

3. System (Server) Vulnerability Management System (VMS)

A System (Server) Vulnerability Management System is a critical tool designed to continuously detect, assess, and remediate security weaknesses across an organization's servers, network devices, and cloud instances, thereby reducing risk. It performs periodic or continuous scanning of operating systems, middleware, applications, databases, and web/service processes, organizing the results according to risk levels for effective management. Vulnerability remediation progress, patch deployment status, unresolved issues, and recurrence rates are tracked and visualized through dashboards and reports.

In practice, both agent-based (installed on the server) and agentless (remote authentication scan) methods are employed. Beyond simple version comparisons, authenticated scans assess configuration vulnerabilities such as misconfigurations, unnecessary services, excessive privileges, and weak encryption. In cloud environments, instances that are transient or part of auto-scaling groups are automatically registered and scanned via tags/labels, and golden images (AMIs/templates) are periodically reviewed. Containerized environments are analyzed separately at the host OS and container image levels, with CI/CD pipeline scans performed to identify risks before deployment.

Vulnerability prioritization does not rely solely on CVSS scores. Risk scoring incorporates factors such as known exploited vulnerabilities (KEV), exploit probability (EPSS), internet exposure, business criticality, data sensitivity, and potential for lateral propagation. Based on this prioritized view, patch campaigns are planned and executed according to a standardized remediation playbook, which includes maintenance windows, rollback procedures, and pre/post functional verification. For vulnerabilities that cannot be immediately remediated, exceptions are documented with timeframes and rationale, while compensating controls—such as firewall blocks, WAF virtual patches, privilege reduction, service isolation, and file integrity monitoring—are automatically applied to mitigate residual risk.

From a Zero Trust perspective, a vulnerability management system quantifies the "trust level" of each server and integrates this data with other security tools and policies. For instance, if a server has high-risk unpatched vulnerabilities, access can be restricted through ZTNA or NGFW, PAM can limit privileged access, EDR can isolate the affected server, and IAM/SSO systems can enforce MFA on related administrative sessions. This establishes a dynamic, real-time vulnerability-based framework within the System Pillar, enabling effective implementation of a Zero Trust environment.

4. Backup & Recovery Management System

A Backup Management System is an operational platform designed to create, store, verify, and restore backups to rapidly recover services in the event of system or server failures or security incidents. Rather than merely saving individual files, the system regularly protects complete server images—including operating systems, applications, configurations, and databases—and allows mounting for immediate service restoration or selective recovery of specific files, emails, or database objects. Within a Zero Trust environment, the Backup Management System manages these functions across on-premises, virtualized, cloud, and SaaS environments in a unified manner.

Backup targets are automatically discovered and registered using both agent-based and agentless methods, and snapshots are created to ensure application consistency. Only changed blocks are transmitted to storage, reducing network and storage overhead, while deduplication and compression improve storage efficiency. Backups are distributed across local storage and remote object storage, with critical segments optionally stored in WORM storage to prevent deletion or tampering. Periodic automated verification procedures, including booting and application checks, ensure that backups are recoverable, with results displayed on dashboards and reports.

The same principles apply to cloud and container environments. In the cloud, newly created instances are automatically included in backup policies through tag/label integration, with disk-level backups orchestrated via snapshot APIs. Kubernetes environments preserve etcd, resource manifests, and persistent volumes, enabling namespace-level restoration. CI/CD pipelines are integrated to capture snapshots before and after deployments, allowing rapid rollback. SaaS data—including Microsoft 365, Google Workspace, and Salesforce—is similarly protected and recoverable under the same policy framework.

For advanced backup management, the system must reflect organizational disaster recovery (DR) strategies. Conceptually, cold sites minimize costs but have longer recovery times, relying on backups and configuration (including infrastructure code) to spin up environments as needed. Warm sites use periodic replication and snapshots to pre-stage critical services, achieving intermediate RTO/RPO. Hot sites employ synchronous or low-latency replication and automatic failover to minimize recovery time, albeit at higher cost. The Backup Management System automates these scenarios through runbooks/playbooks—covering sequences, dependencies, and verification—and can conduct uninterrupted DR rehearsals in isolated environments during operational hours, executing failover and failback procedures in actual incidents.

Zero Trust controls are also integrated. High-risk functions such as backup console access and permanent deletion are governed through SSO/IAM and PAM, requiring MFA and approval. Dedicated backup network segments are separated from the operational network using ZTNA, NGFW, or Micro-Segmentation. During backup, suspicious files or anomalous patterns are isolated, and detection results are fed to SIEM/SOAR systems for automated alerts and follow-up actions. Actual recovery is first validated in isolated environments before being applied to production infrastructure.

The Backup Management System ensures business continuity (BCP) and enforces organizational policies, guidelines, and procedures, thereby maintaining the availability and reliability of the System Pillar.

Within the System Pillar of a Zero Trust architecture, key resources stored on servers are centrally controlled using PAM, Micro-Segmentation, Vulnerability Management Systems, and Backup Management Systems. Privileged access is centrally managed, inter-server communications are finely segmented and restricted, vulnerabilities are continuously assessed and remediated, and recovery from failures or incidents is integrated into a single workflow. These core systems interact with other pillars' key systems—including IAM, ZTNA, and SIEM & SOAR—to sustain and strengthen trustworthiness and availability across both on-premises and cloud environments in a comprehensive Zero Trust framework.

■ Conclusion

Within a Zero Trust Architecture, the System Pillar is a concept unique to domestic (Korean) guidelines and does not exist as a separate pillar in most international frameworks. Globally, servers and related resources are typically managed under the Device or Endpoint Pillar. In Korea, however, due to network-segmented environments and a predominance of on-premises operations, the management and protection of systems (servers) are considered critical. Accordingly, KISA included the System Pillar as a distinct category when publishing Zero Trust guidelines tailored to domestic environments.

The primary focus of the System Pillar in a Zero Trust context is the unified management and consistent application of security policies across systems deployed in diverse environments, including on-premises, public cloud, and private cloud. Effective centralized control of the System Pillar requires both appropriate policies and supporting systems. Management standards and policies should be defined based on core elements such as system inventory, account management, access control, policy management, and patch management, and implemented using systems such as PAM (Privileged Access Management), Micro-Segmentation, Vulnerability Management, and Backup Management Systems.

Because the majority of servers are existing operational systems rather than newly deployed, implementing Zero Trust Architecture must account for backward compatibility, which represents one of the greatest challenges. Given the diversity of operating systems and middleware, the System Pillar may employ a mix of agent-based and agentless approaches, while unsupported systems require custom control policies for monitoring and management.

In conclusion, the System Pillar has been classified separately to reflect domestic operational environments, aiming to manage critical system resources under a Zero Trust framework. The System Pillar does not function in isolation; rather, it is designed to integrate organically with other pillars—Identity, Network, and Data—to enable the full implementation of a Zero Trust Architecture tailored to an organization's environment.

■ References

[1] KISA, Zero Trust Guidelines V2.0, December 2024

[2] NIST SP 800-207, "Zero Trust Architecture", 2020.08

[2] NIST SP 1800-35 Final,"Implementing a Zero Trust Architecture:High-Level Document", 2025.06

[3] NIST SP 800-34, "Contingency Planning Guide for Federal Information Systems", 2010.11

[4] DoD, "Zero Trust Overlays", 2024.06

[5] National Cybersecurity Center(South Korea) National Network Security Framework Guidelines (Draft), January 2025

■ Additional Resources

[1] SK Shieldus, "The Beginning of Zero Trust: Realized with SKZT" – Brochure

[2] Gartner, "Best Privileged Access Management Reviews 2025"

[3] Akamai, "What Is Microsegmentation or Micro-Segmentation?"

[4] CyberArk, "What is Privileged Access Management (PAM)? - Definition"

[5] Net & End, "HIWARE Privileged Session Management for System"

[6] Arctera, "Arctera™ System Recovery 24 User's Guide"