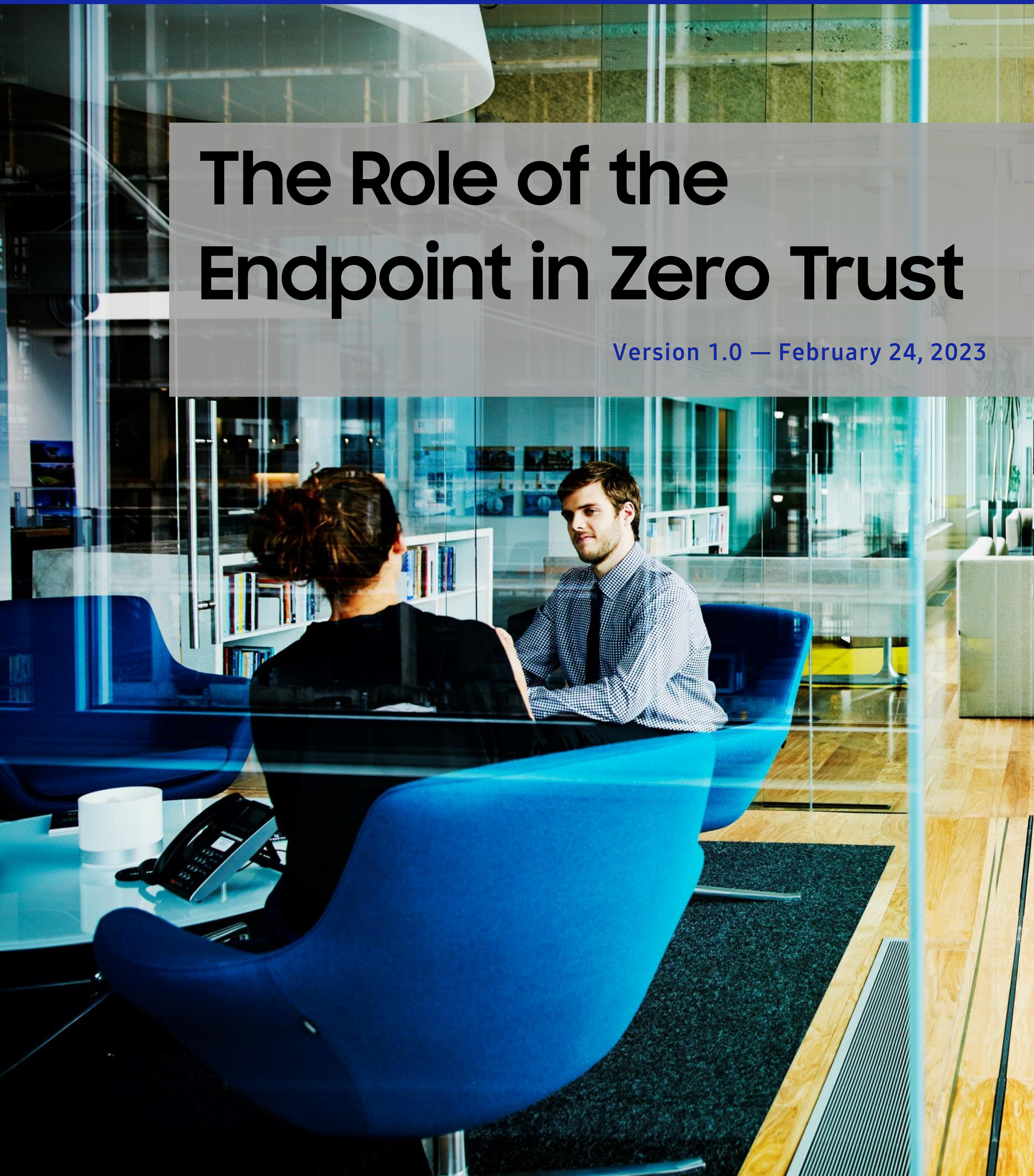
SAMSUNG Knox

White Paper

The Role of the Endpoint in Zero Trust

Version 1.0 — February 24, 2023



Copyright

About this White Paper

This White Paper provides an overview of Samsung's position on the role of secure endpoints in the Zero Trust security strategy, with emphasis on the unique advantages of Samsung Knox that differentiates it from other options in the mobile device security market. This document is designed for C-level executives, security professionals, IT managers, IT admins, and others evaluating Samsung Knox as a solution. For additional information, go to samsungknox.com.

Revision History

Version	Knox Version	Date	Revisions
1.0	23.02	February 24, 2023	First release.

The Role of the Endpoint in Zero Trust

Introduction

Most professionals would like to think that their relationships with employees and partners are based around mutual trust, and studies support this—according to the Harvard Business Review, employees at high-trust companies are demonstrably more satisfied with their work and environment. In the context of security, however, trust can be fatal: the largest identity theft in US history, for example, was conducted by a trusted insider and resulted in a direct financial loss of over $2.7 million. With cybercrime and hacking strategies becoming more and more complex, Zero Trust has become an increasingly popular way of protecting enterprise data. In this article, we will discuss the roles and importance of secure endpoints in enabling a Zero Trust strategy. We will also describe how the Samsung Knox platform provides a foundation on which to build an endpoint that can achieve the Zero Trust vision.

Zero Trust is a security strategy that aims to minimize *implicit* trust in entities that handle enterprise data. Entities, such as users and endpoints[1], need to continuously prove their trustworthiness to the enterprise to be allowed access to resources[2]. For example, just because a device has been authenticated with the enterprise VPN doesn't mean it should be trusted automatically. This approach allows dynamic access control in contrast to traditional perimeter-based approaches such as VPNs, where any entity that is within the enterprise's VPN perimeter is fully implicitly trusted to handle enterprise data. With perimeter-based approaches, if an attacker compromises a single device or user credential, they are usually easily able to breach the entire enterprise network. However, with Zero Trust, access is granted dynamically, which dramatically reduces the impact of a compromised endpoint or credential.

In NIST's Special Publication 800-207 on Zero Trust Architecture, they define a core Zero Trust tenet as "*Access to resources is determined by dynamic policy—including the observable state of client identity, application/service, and the requesting asset—and may include other behavioral and environmental attributes*."

In other words, a key principle of Zero Trust architecture is that the enterprise regulates access to its **resources** based on a continuous evaluation of the **user and device identity**, **device health**, and other contextual information such as location and frequency of access (Figure 1). For each resource request, the device evaluates its health and user identity and gathers other context. The device then sends this information to a Policy Decision and Enforcement Point (PDP, PEP), which decides whether to allow the

[1] Endpoints are user-facing devices that request access to enterprise resources. Endpoints include laptops, desktops, mobile phones, and tablets. In this article, we use the terms "device" and "endpoint" interchangeably.
[2] Enterprise resources include enterprise data, apps, and services such as printers.

endpoint access to the requested enterprise resource. If access is allowed, the requested resource is sent back to the device.

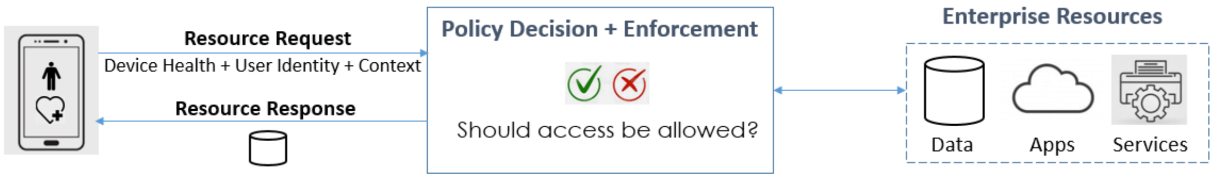


Figure 1. The core operation of Zero Trust.

The endpoint plays a central role in enabling this core operation of Zero Trust of contextualizing user access requests for enterprise resources. First, the endpoint has to continuously collect and evaluate context, such as user identity and device health, with each user request. Second, the endpoint has to protect such context from being forged. If a device is compromised (for example, through malware infecting the OS kernel), then the aforementioned context can be tainted or even forged. Therefore, endpoint security is a critical and necessary component of an overall Zero Trust strategy.

To summarize, a Zero Trust endpoint has to:

- Evaluate and protect user context (such as user identity and behavior)
- Evaluate and protect device context (such as device identity and health)
- Regulate access to local and remote resources

We will now look at each of these responsibilities in more detail.

Figure 2 provides a summary of endpoint responsibilities, associated ZT principles, and enabling endpoint technologies.

Evaluate and Protect User Context

- **Zero Trust principles**:
  - Continuous authentication
  - User context-based access
- **Enabling endpoint technologies:**
  - Anti-phishing
  - Password-less authentication
  - Token binding
  - Secure key storage
  - Biometrics
  - Continuous facial authentication
  - History of duration and time of authentication attempts
  - User behavior monitoring using touchscreen
  - Typing patterns
  - Gait detection

Evaluate and Protect Device Context

- **Zero Trust principles**:
  - Continuous monitoring
  - Device context-based access
- **Enabling endpoint technologies**:
  - Anti-malware
  - Endpoint Detection and Response/Mobile Threat Defense (EDR/MTD)
  - Device analytics
  - Event streams for file/network/process
  - Device health and context attestation
  - Trusted platform

Regulate Access to Local/Remote Resources

- **Zero Trust principles**:
  - Context-based access
  - Fine-grained access control
  - Least privilege
- **Enabling endpoint technologies**:
  - Fine-grained host and network access control
  - Per-app network tunnels
  - Isolation and control of enterprise apps and data

Figure 2. Endpoint responsibilities in Zero Trust, associated principles, and enabling technologies

Evaluate and Protect User Context

Related zero Trust Principles: Continuous authentication, User context-based access

User authentication is central to all enterprise networks. Authentication is the process of evaluating a user's identity to ensure that the user is really who they claim to be. The Zero Trust principles of **continuous authentication** and **access based on user context**, and techniques such as password-less and multi-factor biometric authentication address several security challenges in protecting and evaluating user identity. Realizing these techniques requires device support. Below, we discuss these challenges and solutions further.

Protect User Identity

Attackers use a number of techniques to steal user identity information such as passwords and session tokens. Social engineering attacks like phishing trick the user into entering their credentials, such as passwords or SMS authentication codes, into an attacker-controlled website. The attacker is then able to use these credentials to impersonate the user and get access to enterprise resources. Malware is another commonly used technique to steal session tokens or cookies. Session tokens or cookies are sent to the device after a user authenticates successfully. Malware on the device reads and exfiltrates these cookies, typically stored in files, to an attacker. The attacker then replays this cookie on another device to get access to enterprise resources. This attack is known as **session hijacking.** A study showed that phishing and credential stealing were the top two causes for real-world data breaches in enterprises in 2022.

Endpoints need to enable a range of techniques to protect user identities from such attacks. First, devices need to continuously monitor, detect, and block malware and phishing attacks that steal user credentials. Endpoint Detection and Response (EDR) and Mobile Threat Defense (MTD) systems are typically responsible to protect against these attacks, and are discussed in the next section. Endpoint technologies such as cloud browsers also help isolate users from these attacks by controlling access to websites and disallowing downloads of files to device.

Another approach to secure user identity is for the device to support password-less authentication and token binding. In password-less authentication standards, such as FIDO2, passkeys, WebAuthN, and TLS client authentication, a server uses an on-device private key instead of a password to authenticate the user. Token binding standards tie tokens, such as cookies, to the device, by binding them to an on-device private key stored in secure storage. Possession of both the cookie file and the private key is required to use the cookie. Therefore, even if an attacker steals the cookie file, the stolen cookie is unusable on any other device in the absence of this private key. Finally, to bind the cookie additionally to user identity, the device allows use of these private keys only after the user authenticates through biometrics. Thus, to support these standards, the device needs to provide a secure environment for storing private keys and to enable secure biometric authentication.

Evaluate User Identity and User Context

Zero Trust addresses several security challenges in properly evaluating user identity. First, user identity may change after initial authentication. For example, an authenticated user could walk away from their device and allow another user to start operating the device. Another user could “shoulder-surf” and look at sensitive information. Even password-less authentication isn’t safe—attackers can steal credentials such as a password or a cookie and replay these stolen credentials from another device or from a malware app on the same device.

To address these challenges, Zero Trust requires **continuous authentication** of user identity even after initial login and **access based on user context.** Endpoint technologies that support these principles are multi-factor biometric authentication, continuous facial authentication, monitoring user behavior using analysis of touchscreen and typing patterns, and other context such as considering which app is making data requests.

Samsung Knox and User Context

Samsung Knox Protects User Identity

Samsung Knox's Network Platform Analytics and Domain Filter firewall features help EDR/MTDs detect and block phishing attempts. Samsung Knox supports secure cryptographic operations in the hardware-enforced ARM TrustZone Trusted Execution Environment (TEE), as well as an isolated tamper-proof processor called the Knox Vault. The Knox Vault is a standalone secure processor, separate from the main application processor that runs software such as Android and user applications. These technologies enable password-less authentication and token binding standards. Knox also supports secure biometrics authentication in the ARM TrustZone TEE. Finally, Samsung Knox's Network Platform Analytics identifies the exact app on a device making a network request. This information provides further confidence to servers that session cookies are not stolen and replayed even by malware apps on the same device.

Samsung Knox Enables Continuous Authentication

Samsung Knox's continuous multi-factor authentication framework allows for the continuous collection and interpretation of information to authenticate user identity, such as behavioral touch dynamics to check whether the current user's typing pattern matches that of the device's owner and continuous facial authentication. In future releases, we'll continue to expand the number of data sources that can be used to enable continuous authentication to the Knox platform.

Evaluate and Protect Device Context

RELATED ZERO TRUST PRINCIPLES: CONTINUOUS MONITORING, DEVICE CONTEXT-BASED ACCESS

In Zero Trust, information about device health is critical to the policy decision point to determine if the device is sufficiently trustworthy to be granted access to the requested resource.

The device has an important role in gathering, evaluating, and protecting device health and identity. The Zero Trust principle of **continuous monitoring** requires a device to continuously evaluate and protect its health from being compromised by attacks. In addition to device health, the device has to gather additional context such as location and send it to a remote policy decision point (PDP), which determines if the device is sufficiently trustworthy to be granted access to the requested resource according to the Zero Trust principle of **device context-based access**. Finally, and most importantly, the device needs to provide proof to the PDP that these context values from the device are authentic and not faked by malicious software.

Evaluate Device Health and Identity

The device's first responsibility is to gather rich contextual information such as device health and supply it to the PDP. Endpoint Detection and Response (EDR) and Mobile Threat Defense (MTD) subsystems continuously monitor, evaluate and provide device health information. Device intelligence subsystems provide additional contextual information for analytics such as current and historical location, WiFi access point connection history, and app usage. Device identity can be provisioned during device manufacturing or by an enterprise. The device manufacturer (OEM) provisions device identity in hardware or secure storage in the factory. In addition, the device may support protocols such as ACME device attestation to allow an enterprise to provision an identifying certificate on to the device.

The device's second responsibility is to prove the trustworthiness of this contextual information to the PDP. While context is collected from subsystems such as EDR, MTD, and device intelligence, how does the PDP know that these subsystems are not compromised or faking this information? The solution is to use device health attestation backed by a hardware-rooted trusted platform. First, the trusted platform verifies that every security-critical platform component, starting from the very first hardware ROM process during device boot all the way up to security-critical subsystems such as EDR and MTD, is cryptographically signed by an authorized key before being allowed to run. This way the device can trust data and measurements generated by these verified components, since they are known to run only authorized code. Second, an attestation provides evidence of this same fact - that these measurements were generated by verified platform components and can be trusted - to the remote PDP using cryptographic signatures.

Protect Device Health

Endpoint security protects the device from compromise by using a platform that provides boot-time and run-time security. Once a platform has booted up using only approved platform components as described above, EDR and MTD subsystems continuously monitor device activity to detect suspicious behavior, malware, vulnerable software, and exploit penetration attempts such as rooting or jailbreaks. To help EDR/MTD systems effectively perform these functions, the device needs to provide full system-wide visibility and control over events across apps, network, the OS kernel, file system, and other firmware components. Further, the platform's access control has to protect and isolate security-critical subsystems such as EDR/MTD and producers of contextual data from potentially malicious untrusted software, such as third-party apps. Devices also typically provide host firewalls to block traffic that is not explicitly permitted.

Samsung Knox and Device Context

Samsung Knox Protects Device Health

Samsung Knox's trusted platform provides a basis to trust any Samsung device, whether the device is enterprise-owned, BYOD, or unmanaged. All Samsung Knox devices support trusted boot and Device Health Attestation and provide verifiable guarantees that only Samsung-authorized platform software components are running on the device. Further, Samsung Knox's Real-Time Kernel Protection (RKP) feature defends against rooting attacks at run-time. Significant portions of the Knox platform, including parts of the hypervisor and TrustZone, are written in the memory-safe Rust language, further hardening the platform against run-time attacks.

Samsung Knox Provides Hardware-Attested Device Identity

Samsung provisions a unique ID and signing key per device (called the Samsung Attestation Key (SAK)) together with a signed certificate chain in secure hardware in the factory. Apps can use Samsung's Knox SDK to generate key pairs with a certificate chain containing the device's identity signed by the device's SAK. Remote servers can use this certificate to verify the identity of the device they are communicating with. Further, the generated keypair can be used for token binding, specifically for signing claims including the access token in communicating with remote servers.

Samsung Knox Enables Continuous Monitoring of Endpoint Security

Samsung Knox provides a rich variety of contextual information typically used in Zero Trust. To help MTDs function effectively, the Samsung Knox SDK provides access to hundreds of data points spread across the device's network, kernel, filesystem, and apps and processes for extensive system-wide visibility. Knox's Zero Trust Framework allows on-device agents such as Mobile Device Management (MDM) to be notified on changes in device or user context. Agents can act on such notifications by stopping or blocking certain apps on-device, and can also pass on this information to remote PDP servers to cut off access to enterprise resources. Knox Asset Intelligence provides data such as location tracking, app usage statistics, and WiFi events. Knox's Network Platform Analytics provides network analytics such as the name of the app initiating the network connection. We continue to expand support for continuous monitoring by adding more data sources and Knox platform support for MTDs, device management agents, and remote PDPs.

Regulate Access to Local and Remote Resources

RELATED ZERO TRUST PRINCIPLES: CONTEXT-BASED ACCESS, FINE-GRAINED ACCESS CONTROL, LEAST PRIVILEGE

As we have seen, a key principle of Zero Trust architecture is that a PDP regulates access to enterprise resources based on a continuous evaluation of user and device context. It is the endpoint's responsibility to transmit user and device context to the remote PDP along with each request.

Next, after resource access is granted, the endpoint has to regulate access to local on-device and remote resources following the Zero Trust principles of **least privilege** and **fine-grained access control**. The endpoint also needs to take *remedial* measures if it detects suspicious user or device behavior.

Regulate Access to Remote Resources.

When requesting access to resources, the endpoint has to transmit the collected user and device context to the remote PDP. In certain instantiations of the Zero Trust architecture, such as Zero Trust Network Access (ZTNA), the endpoint has to support intercepting each network request to the enterprise using techniques such as proxies, and augmenting the network request with additional context data.

Next, once access is granted, Zero Trust Network PDPs typically grant the *entire* device access to resource servers. However, following least privilege, the device should ensure that only the particular authorized client app within the endpoint talks to resource servers. Endpoint technologies such as per-app network tunnels and firewall rules enable such fine-grained network access control of individual network flows.

Regulate Access to Local Resources.

Once access is granted, enterprise resource data is downloaded and displayed on the device. The device's fine-grained host access control should ensure that only authorized apps and users are allowed access to these local on-device resources in ways authorized by the enterprise's policy. Technologies such as enterprise workspaces isolate and protect enterprise apps and data and further allow policy controls such as preventing screenshots and disabling the copying and pasting of enterprise data.

Perform Remediation

When a session ends, or when continuous monitoring or authentication indicates suspicious user behavior or an endpoint compromise, the device needs to revoke access to on-device resource based on enterprise policy. This includes actions such as wiping enterprise data from the endpoint after the session ends, locking or blanking the screen, or triggering re-authentication.

Samsung Knox Supports Fine-Grained Access Control

Samsung Knox's Global proxy feature provides mechanisms to intercept network flows. Samsung Knox Platform for Enterprise (KPE) provides strong local isolation and control over enterprise apps and data. This includes features such as sensitive data protection to provide fine-grained access control to user data. Knox supports advanced network tunneling and the ability to control the domains an app can talk to. Knox's Network Platform Analytics feature provides information about which app makes a connection request. These technologies together provide a basis for fine-grained network access control. Knox Guard and KPE also provide several remediation controls to instantly cut off resource access.

Conclusion

As cyberattacks become increasing commonplace and complex, the industry continues to move towards Zero Trust as a way to protect enterprise data from malicious actors. Endpoint security is a key part of Zero Trust, and as explained above, there are a number of important functionalities that must be built in to endpoints in order to enable the Zero Trust vision. Samsung Knox's trusted security platform, combined with its advanced device monitoring capabilities, are a foundation on which to build an endpoint that follows Zero Trust principles. However, perfect cybersecurity is never a static objective, and Samsung continues to strive to improve the Knox platform to better enable secure Zero Trust deployments to protect your enterprise's valuable data.