SOLUTION BRIEF

altOS Threat Model

Version 14.4

March 2026

Contents

Terms and Definitions

Term	Meaning
AAID	**Android Advertising ID** – a 32-character hexadecimal identifier used by advertisers and third parties to link a user's activities across multiple apps ("cross-app tracking"), to build profiles of user interests, preferences, and behaviors. It also facilitates data aggregation by allowing data brokers and ad networks to cross-link information collected from different sources to create detailed user profiles. AAID is also known as the Google Advertising ID (GAID).
ADB	**Android Debug Bridge** – a versatile command-line tool is intended for use by software developers (to install and debug software on Android devices, transfer files, view device logs, access hidden features, and run shell commands). ADB over USB is also a common attack vector for forensic analysis tools.
AOSP	**Android Open-Source Project** – an open-source project led by Google, which develops and licenses the reference implementation of the Android OS. AOSP may also refer to the open-source version of the Android framework code, system components, default apps, and build tools and scripts that the AOSP licenses under the Apache open-source license.
ASM	**Attack Surface Management** – the practice of identifying, monitoring, and reducing vulnerabilities to minimize the potential attack surface of a corporate network, for example. For example, Censys and Shodan provide services and a database that is used for ASM purposes by corporations.
AS	**Autonomous System** – one of the networks (e.g., an ISP network) that makes up the Internet. Each AS is identified via a unique identifier – the AS number or "ASN". Data packets crossing the Internet are routed across multiple ASs until they reach the destination IP address.
DNS	**Domain Name System** – the Internet Domain Name System, which is most commonly queried to obtain public IP addresses corresponding to domains.
Fingerprinting	The creation of an identifier for a device based on a combination of observable attributes without relying on traditional tracking methods like cookies or advertising IDs. A fingerprint can then be used to track users across apps and websites, deliver targeted advertising, or bypass privacy controls (e.g., resetting or zeroing an Ad ID).
AWS-GC	**AWS GovCloud** – the AWS cloud computing service for sensitive US government applications and data. AWS-GC is exclusively for US government agencies, contractors, and other organizations operating under US regulations, including FedRAMP, ITAR, and DoD SRG.
GMS	**Google Mobile Services –** proprietary apps, such as Play, Gmail, Chrome, and Maps, which Google develops.
IMEI	**International Mobile Equipment Identity** – a unique 15-digit number assigned to every mobile phone, as well as some satellite phones and tablets. IMEIs are used to identify devices to cellular networks, but they may also be used for tracking purposes.

IMSI	**International Mobile Subscriber Identity** – a unique 15-digit number used to identify a specific user on a cellular network. The IMSI is inside the Subscriber Identity Module (SIM) card or assigned to an e-SIM (electronic SIM).
MAC Address	**Media Access Control Address** – a unique identifier assigned to a network interface controller (e.g., an Ethernet, Bluetooth, or Wi-Fi NIC) for use as a network address within a network segment.
MTD	**Moving Target Defense** – a proactive defense mechanism that dynamically changes an attack surface, making it more difficult for threat actors to identify and/or target.
OSINT	**Open-Source Intelligence** – the collection and analysis of data from open sources to produce actionable intelligence.
SNI	**Server Name Indication –** an extension to the TLS protocol that enables a client to indicate the host it is attempting to connect to at the start of a TLS handshake. This enables name-based virtual hosting as multiple web servers can be hosted on the same IP address.
TLD	**Top Level Domain** – the highest level of the Domain Name System after the root domain. Common TLDs include ".com", ".org", and ".net", and country-specific TLDs such as ".uk" (United Kingdom) and ".cn" (China). There are 1500+ TLDs, of which 308 are country code top-level domains (ccTLDs)
TLS	**Transport Layer Security** – a cryptographic protocol that provides secure communications over a network. TLS is widely used for email, instant messaging, and web browsing (HTTPS).
WPA	**Wi-Fi Protected Access** – a security protocol for wireless LANs to prevent unauthorized access to WLANs and to protect the confidentiality and integrity of packets transmitted on WLANs. The initial version, WPA, was developed in 2003. Improvements in security were introduced with WPA2 in 2004 and WPA3 in 2018.

1 Introduction

The objective of a threat model is to identify potential threats and vulnerabilities for a system, as well as the countermeasures or defenses in place to address the threats and remediate any vulnerabilities.

The altOS mobile platform is designed to address the operational, security, and privacy requirements of government agencies and mobile device users exposed to high threat levels. The threat model assumes the platform and its users are exposed to the following threats:

- **Normal cyberattacks** – opportunistic attacks on vulnerable devices or servers by criminals, hacktivists, or independent threat actors that have limited resources. The objective of these attacks may be profit, to inflict reputational damage, to claim bragging rights, or ideology,
- **Targeted cyberattacks** – highly directed or focused cyberattacks against specific government agencies, undertaken by nation-state, terrorist, or cartel threat actors with significant resources (funding, technical talent, and access to intelligence) for espionage, sabotage, political gain, or military advantage,
- **Targeted attacks on users** – highly directed attacks on device users (threats, coercion, bribes, detention, kinetic attacks, etc.) that stem from the ability to identify and profile devices via digital data obtained via device tracking vectors and OSINT, and
- **Loss of physical control** – mobile device users may lose or misplace a device, or they may have to relinquish physical control of a device temporarily. In the worst-case scenario, a smartphone is subject to a cursory inspection (e.g., at a border crossing) that may reveal sensitive apps and/or data on the device. This, in turn, may lead to further questions of the user's intentions, role, or activities, and/or forensic extraction of data from a device.

This whitepaper discusses the altOS threat model in detail and is intended for potential customers. It is also intended for internal use - to identify new threats, risks, and defenses.

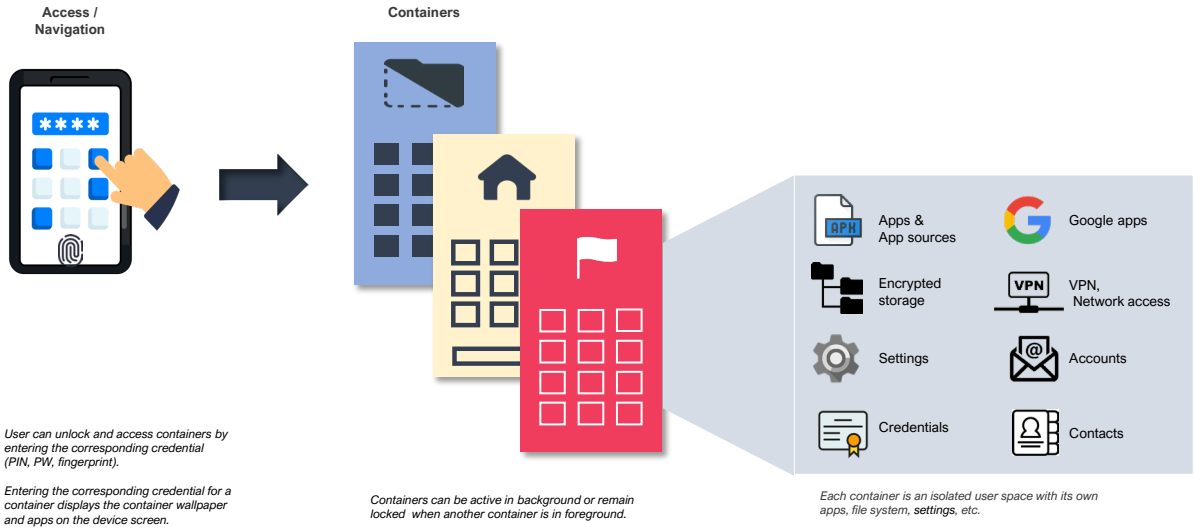
Note that this version of the document applies to version 14.4 of the altOS Platform (comprising both the Backend server web application and mobile device system image).

2 altOS Platform

The altOS Platform provides sophisticated container and management capabilities while maintaining an intuitive user experience. Figure 1 shows a device configured with multiple containers. Each container is a separate Android user space and has different apps, encrypted file systems, app sources, contacts, accounts, phone profiles, security policies, wallpapers, etc.

The user experience in each container is almost identical to the stock Android user experience, save for a limited number of security functions the user may have to trigger or initiate.

Figure 1. Containers and user access



Users unlock and access specific containers by entering the PIN, password, or fingerprint associated with the container. Containers can be in one of three states, depending on the actions the user takes concerning unlocking and accessing different containers, and subject to administrator policy settings. These states are:

- **Locked** – the container file system is encrypted and inaccessible. Apps are inactive (unless they have been written specifically to use Device Encrypted memory) and the user has entered the password or PIN for the primary container.
- **Unlocked in foreground** – the user has entered the container access credential, and the container (wallpaper, apps, notifications) is visible on the device display. Apps can access the container file system, sensors, and RF modems as per the container settings and the administrator policy settings, and
- **Unlocked in background** – the user has entered the container access credential and then switched containers to bring another container to the foreground. In this state, access to system services may be limited by the operating system (e.g., location data) or by some administrator settings (e.g., exclusive network access will prevent containers in the background from accessing the network if applied to the foreground container).

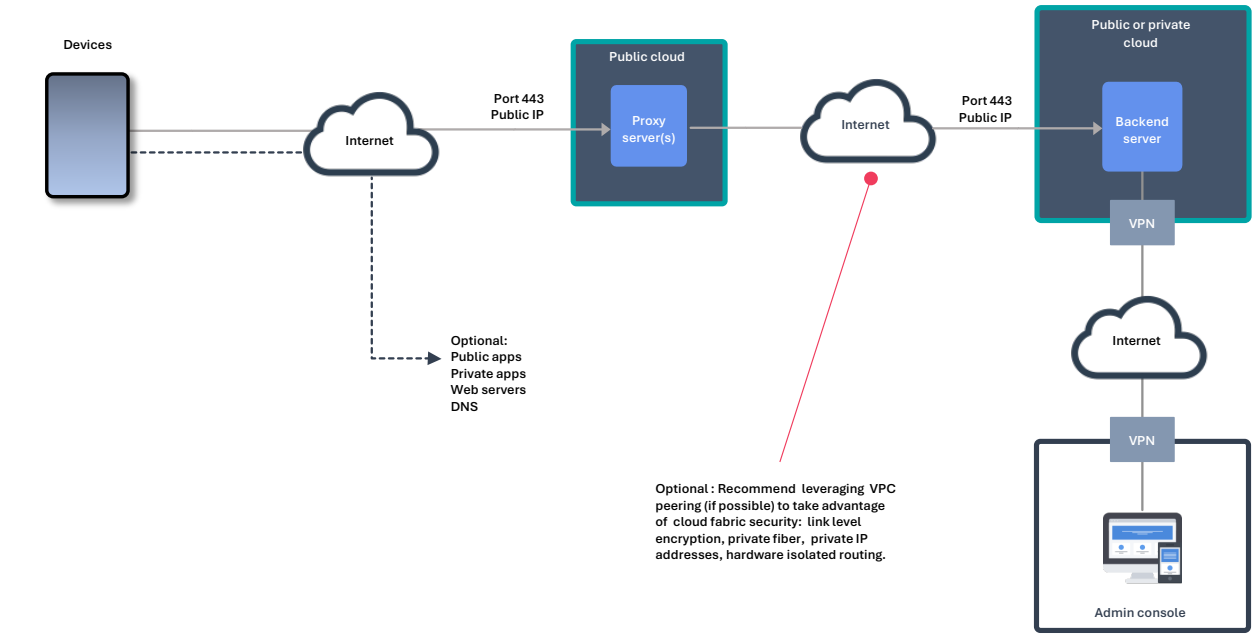
While containers may be provisioned for personal use (usually the primary container), the administrator retains the ability to control, update, and delete all containers on the device and retains control of critical policy settings at the device level. For example, the administrator may grant access to Google Play in a personal container but can define an app blacklist that is applied at the device level. This could be used, for example, to blacklist an app

that is found to be harmful, and if this app is installed in any container, it will be deleted from the container along with the app data.

Figure 1 provides a system diagram that includes:

- **Devices** – these are consumer Android smartphones that have the OEM system image[1] replaced with the altOS system image. The altOS system image is signed by CIS Secure and integrity verified during the device boot process. Devices are enrolled with the Backend server via invitations (QR codes) that can be user-locked and distributed out-of-band. Once a device is enrolled with the Backend server, multiple containers can be installed and managed on the device remotely,
- **Proxy servers** – these are NGINX servers that relay TLS connections to the Backend server. All devices connect to the Backend via a Proxy server to isolate it from direct Internet access. Multiple Proxy servers can be active at one time, with different groups of devices "homed" onto different Proxies. Additionally, Proxy servers can be added at any time and administrators can direct devices to the new Proxy servers by updating device *Proxy Profiles* or on a schedule (date/time) specified in a *Proxy Profile* previously pushed to a device, and
- **Backend server** – this is comprised of several services and micro-services that provide the administrator web console, device and container management, logging, app and file download, device backup and restore, and OTA updates.

Figure 2. System diagram



The platform supports the following deployment models:

- **Cloud** – Proxy and Backend servers deployed in AWS, Azure, or GovCloud. Proxy servers can be deployed in different cloud data centers to obfuscate the location of the Backend server. For example, it may be

[1] The OEM version of the Android operating system and any pre-installed apps that are flashed onto the device in the factory.

desirable to have devices connect to Proxies hosted overseas when these devices are used in a foreign country,

- **Hybrid** – Proxy servers hosted in AWS or Azure, and the Backend hosted in a private cloud or AWS GovCloud, and
- **Private cloud** – Proxy and Backend servers can also be deployed in a private cloud or network. Note that this would make it easier for a potential adversary to associate or link a device with a specific government agency or IP geo-location, as opposed to the scenario where devices connect through Proxy servers deployed in different AWS or Azure data centers.

From a networking and threat perspective, the following should be noted:

- **TLS v1.3** – device to Proxy traffic is TLS v1.3 encrypted, as is all intra-server traffic,
- **Server-to-server traffic** – Proxy servers connect to the Backend server via the Internet. While we recommend leveraging VPC or VNet peering in AWS or Azure, if possible, the threat model does not make this assumption,
- **Device-to-Proxy traffic** – packet metadata is assumed to be visible to an adversary on device-to-proxy traffic. Although this traffic is encrypted, the adversary can capture the source and destination IP address and the Proxy SNI,
- **Backend server connectivity** – access to the Backend server is limited to the Proxy servers and the admin via VPN; Note that this is addressed during the installation process by limiting IP address access to the Backend server or by deploying the Backend server in a private subnet,
- **Local DNS** – devices and servers do not query public or private DNS resolvers to resolve IP addresses. The Proxy server IP addresses are determined locally on the device, and server IP addresses are resolved by domain name queries to the Linux OS /etc/hosts file[2], and
- **Internet access** – the threat model assumes devices can access the Internet and public and private apps, web servers, and other services (e.g., DNS resolvers).

[2] Proxy server IP addresses are resolved locally on devices using the IP address and SNI in the assigned Proxy Profile. The Proxy servers and Backend server resolve IP addresses using the Linux /etc/hosts file on each node. The /etc/hosts file is a system file on Linux that is used to map hostnames to IP addresses, acting as a local DNS resolver. The /etc/hosts file is consulted before any DNS resolver is queried.

3 Threats

3.1 Introduction

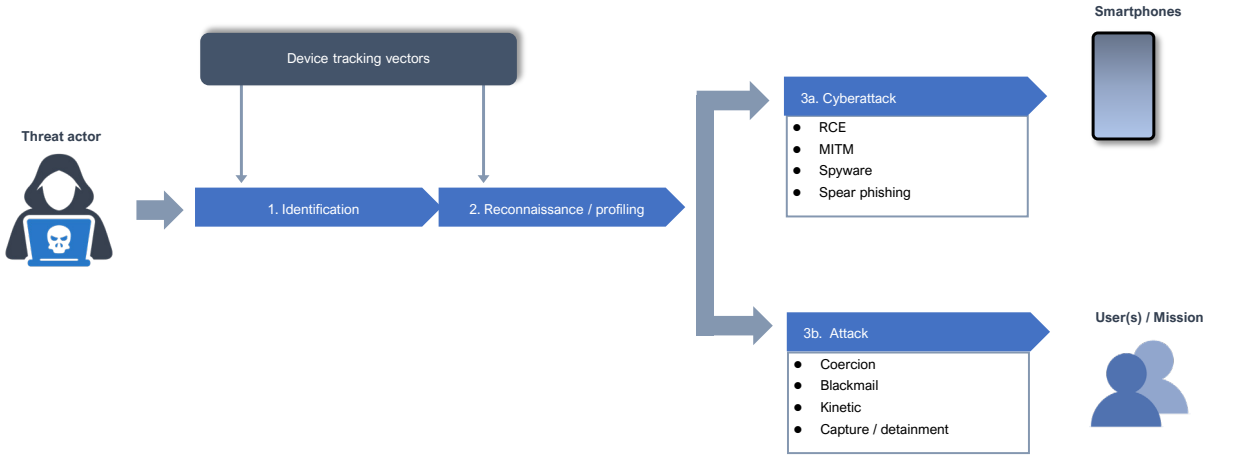
The platform and device users are assumed to be exposed to the following threats:

- **Normal cyberattacks** – opportunistic attacks on vulnerable devices or systems by criminals, hacktivists, or independent threat actors that have limited resources. Regular cyber attacks may be for profit, reputational damage, or ideological reasons,
- **Targeted cyberattacks** – highly directed or focused cyber attacks against specific government agencies, undertaken by nation-state, terrorist, or cartel threat actors with significant resources (funding, technical talent, and access to intelligence) for espionage, sabotage, political gain, or military advantage,
- **Targeted attacks on users** – these are attacks directed at the smartphone user or users (threats, coercion, bribes, capture, detainment, kinetic attacks, etc.) that stem from an adversary's ability to identify and profile devices via digital data exhaust ("device tracking vectors"), and
- **Loss of physical control** – there are scenarios under which a user may lose or misplace a device or may have to relinquish physical control of a device temporarily. In the worst-case scenario, an unlocked smartphone may be subjected to visual inspection (e.g., at a border crossing) that could reveal sensitive apps and/or data on the device. This may lead to further questions regarding the user's intentions, role, or activities and, ultimately, forensic extraction of data from a device.

3.2 Attacks on devices (and users)

Figure 3 shows the stages of a targeted attack[3] on a device or user. The first phase is identifying devices of interest using data from tracking vectors. For example, a device may be flagged as a device of interest because location data indicates an association with a military base. The identification phase is then followed by a reconnaissance phase where the adversary tracks and profiles the device using the same data sources. For example, the device

Figure 3. Targeted attacks on devices (and users)



[3] Note that in a normal cyberattack, the adversary has limited resources (including limited or no access to device tracking vectors) and the attack proceeds directly to Phase 3a without explicitly targeting devices because of who the user is, or where they work, or the function they perform

identified in the previous example is profiled using data from device tracking vectors to determine other locations of interest to the adversary, user patterns of life, and organizational patterns.[4]. The third stage is a targeted attack, which may be directed at the device (e.g., a spyware attack) or the device user (e.g., threats, coercion, or worse), depending on the objectives of the threat actor.

Table 1 lists the device tracking vectors that a sophisticated threat actor is assumed to be able to access to identify and profile devices, as well as the threat level for each vector used independently.

The risk to devices and users stemming from Ad Tech data is assumed to be very high because:

- **Coverage is global**. Devices can be tracked and profiled around the globe (or anywhere there is Internet connectivity),
- **Ad tech data is accessible.** Bid stream data sent to real-time ad auctions is accessible by hundreds or thousands of companies around the globe[5]. It is also for sale by data aggregators.[6], and
- **It contains location data.** Location data may be available in near real-time, assuming the tracking party is an active "bidder" in one or more ad auctions and aggregators provide work and home locations in their data sets.

Bid stream data:

Data is transmitted by ad libraries embedded in apps or JavaScript embedded in websites. The data is sent to online advertising auctions to sell advertising "space" or "inventory" to brands or ad buyers. The data typically contains location and a user profile as well as information about the requesting app or website. There are hundreds of companies worldwide participating in various real-time advertising auctions that have access to this data.

The tracking risk associated with Wi-Fi and Bluetooth tracking vectors is low because:

- **Identifiers are randomized** – MAC addresses used for Wi-Fi probe requests and to connect to an SSID are randomized by default (they change every 24 hours),
- **Coverage is localized** – for Wi-Fi, coverage is limited to the coverage of a single Wi-Fi scanner or access point or multiple access points in a single ESS (Extended Service Set), which may be deployed at an airport, university campus, shopping center, etc. Coverage for Bluetooth is more restricted than Wi-Fi, and hence the tracking risk is lower than Wi-Fi, and
- **Cross-linking tracking data –** Wi-Fi and Bluetooth MAC addresses are inaccessible to apps (and ad libraries embedded in them). This makes it more challenging to cross-link data from other tracking vectors.

[4] *Academic Project Used Marketing Data to Monitor Russian Military Sites*, Byron Tau, Wall Street Journal, July 20, 2020.

[5] Ad auction provider's response to Senators Wyden, Gillibrand, Brown, Cassidy, Warner, and Warren regarding the sharing of bid stream data with foreign companies.

[6] FTC Finalizes Order Banning Mobile Walla from Selling Sensitive Location Data.

Table 1. Device tracking vectors

Tracking Vector	Information available	Coverage / availability	Risk
ADTech Data • Bid stream data • Social media • Location trackers • Data agregators	• Identifiers: AAID, cookies, device fingerprints • Device & carrier information • Apps / app usage / browsing history • Location – work, home, covert, daily routine • Patterns of life • Co-location - colleagues, family members • Organization patterns, supply routes, etc.	Global coverage Data freely available to thousands of companies or for sale by data aggregators.	**VERY HIGH**
System Services • Network Location Provider (NLP) • Enterprise OTA update • Zero Touch Enrollment (ZTE) • App store • Payment	• Identifiers: IMEI, other Device IDs including AAID • Employer • Apps • Location data (via GPS and/or NLP) • Device model, OS version • Accounts / contacts (?) • Installed apps and services	Data is not freely available or sold although it may used to supplement bidstream data. Access might require network intrusion or insider access	**HIGH** Huawei, other OEMs Enterprise resellers **MED.** Google , Apple
Cellular Networks • Domestic • International (roaming)	• Identifiers: IMEI, IMSI • Cellular geo-location • Call logs • DNS queries • IP traffic • Home address / account info (domestic provider) • GPS location (if pre-installed carrier apps)	Regional or national coverage. Access to call logs, cell tower dumps is subject to regulation in many countries.	**MED:** Roaming **VERY HIGH:** Adversary controlled network or conflict zone
IMSI catchers / Rogue cell sites • Active • Passive	• Identifiers: IMEI, IMSI • Location data	Threat increases in proximity to monitored locations. (e.g. near a monitored military base)	**HIGH**: Monitored locations – domestic and int'l
Wi-Fi / Bluetooth • Access point provider • BLE beacons • Scanners	• MAC addresses & proximity (Wi-Fi) • MAC address, service data, and proximity (BLE) • MAC address , device name, and proximity (BT Classic)	Localized coverage	**LOW**

3.3 Attacks on servers

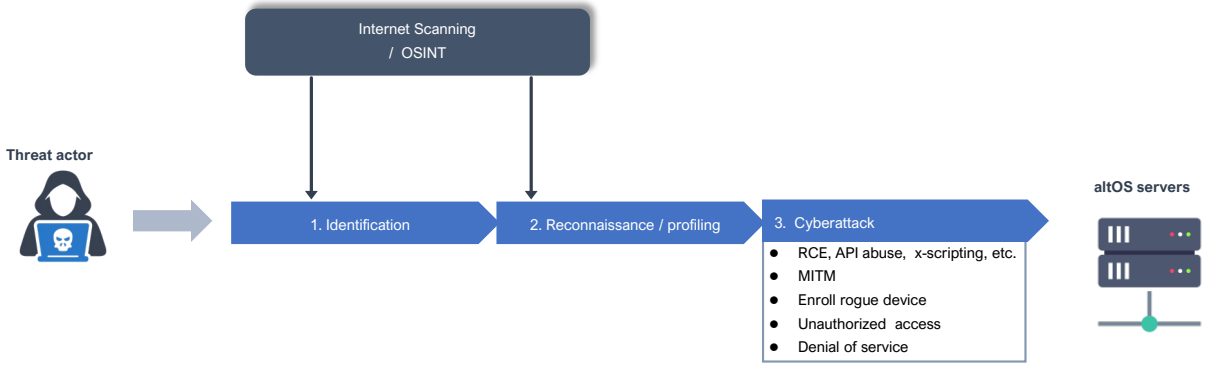
The Backend server houses sensitive data and controls security policy for a fleet of devices. The Proxy servers provide a path from the Internet to the Backend server and are in the Backend server attack chain. Hence, the servers are potential attack targets.

There are two potential paths for targeted attacks on the servers:

- **Scanning** – an adversary (or an AI engine) flags an Internet host using OSINT and/or Internet scanning data. For example, an adversary may monitor CT logs[7] to identify hosts that were recently connected to the Internet in a specific geographic region or to identify hosts belonging to a particular government agency. Once the adversary further profiles the host using active probes and or OSINT, a targeted attack may be directed at the Internet host, and
- **Traffic monitoring** – an adversary monitors traffic from devices under surveillance to determine what the device is connecting to private or public apps and web servers the device is connecting to and communications patterns with those servers. This may, in turn, enable the adversary to identify other devices and users connecting to a server, or it could lead to a targeted attack on the servers.

[7] CT logs are monitored in real time by many Internet scanners. See *Certifiably Vulnerable: Using Certificate Transparency Logs for Target Reconnaissance*, 2023 European Symposium on Security and Privacy.

Figure 4. Scanning initiated attack



In terms of traffic monitoring, there are two categories of monitoring:

- **Local network surveillance** – the adversary can eavesdrop on traffic passing through a network segment. An example would be an adversary that has access to a Wi-Fi access point/router at a coffee shop, and
- **National network surveillance** – the adversary has the capabilities of a national censor with surveillance capabilities over all ISPs and backbone routers in a country. In addition to monitoring and analyzing traffic patterns and metadata, the adversary may also have the ability to block traffic based on IP addresses, SNI, or port and to inject or modify traffic, such as DNS queries.

Figure 5. Traffic monitoring-initiated attack

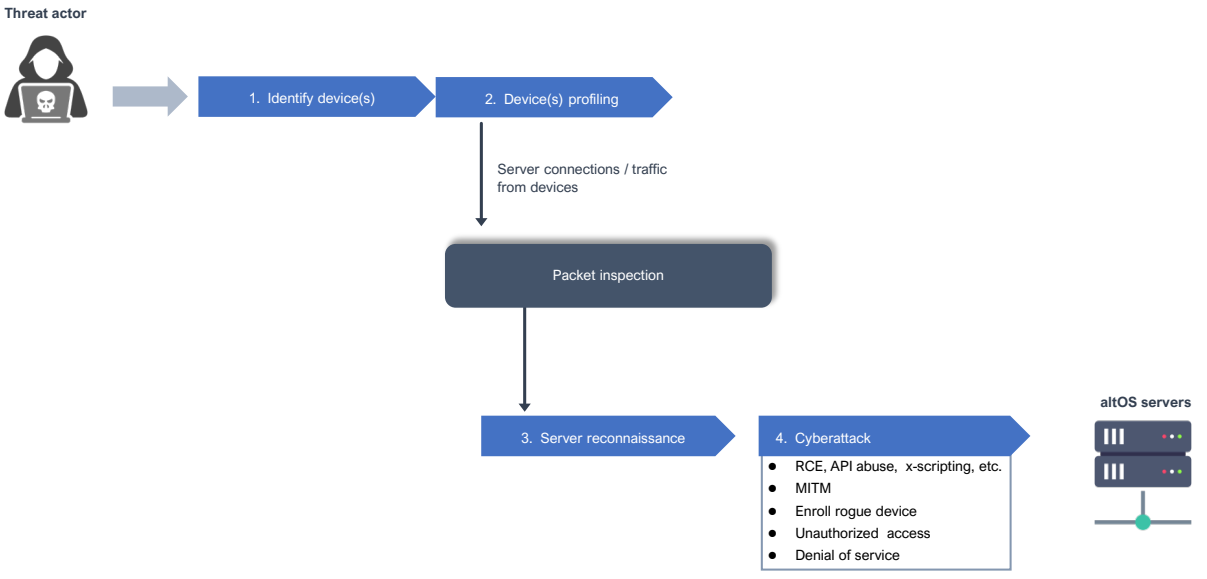


Table 2 identifies the server tracking vectors that may be used by an adversary in the preliminary stages of a targeted attack. Note the following:

- The assigned risk levels assume no countermeasures have been implemented to minimize the threat associated with each tracking vector,
- Commercial ASM databases rely on or use OSINT to supplement their scanning data. For example, Censys pulls certificate data from CT logs and "finds" new IPv6-based hosts by monitoring CT log updates[8],
- Adversaries would most likely use multiple tracking vectors (e.g., packet inspection and Internet scanning), so it is challenging to assign risk levels for each tracking vector independently,
- A low threat level is assigned to localized network scanning (e.g., Wi-Fi scanning or a MITM attack) because the adversary has to be near a device, and the device would have to be actively communicating with a Proxy for an adversary to observe or capture any network traffic. But we would assign a high-risk level to an attack initiated by traffic monitoring if the device is roaming on a foreign network.

Table 2. Server tracking vectors

Tracking Vector	Host meta data	Coverage / availability	Risk
OSINT • Certificate Transparency (CT) Logs • TLD Zone files • WHOIS	• Domains, sub-domains, organization, division, city, state country, email addresses • Service headers • Account info. (TLD registry or registrar)	• Easily accessible as OSINT or CSINT • Access to CCTLD zone files is variable. Zone files for generic TLDs are readily available • TLD registries / registrars may be co-opted	**High**
Internet Scanners • Censys, Shodan, other ASM databases • Proprietary scanners	• IP address & geo-location • Ports • Service banners • OS, software versions • Fingerprint (TCP/IP stack) • Certificates	• IP v4 address space scanned in < 24 hours • ASM services provide API access for fees • Internet hosts scanned in minutes of certificate push to CT logs	**Very High** Attribution risk may also be very high
National network monitoring • DPI • DNS query monitoring / modifying • IP address / port / DNS / SNI filtering	• IP addresses & geo-location • Ports • SNIs • DNS queries • Protocol detection	• Coverage limited to the adversary-controlled portion of the internet – this can	**Very High** when roaming on adversary controlled cellular and ISP networks
Local network monitoring • Control over AP or WI-Fi monitor mode • Wireshark PCAP files	• IP addresses & geo-location • Port • SNI • TLS metadata (version, ALPN, TLS extensions, etc.)	• Localized coverage • Monitor mode is usually limited to 1 802.11 channel at a time	**Low** Unless a device connects to a public Wi-Fi SSID, the adversary is in range, and the device is communicating frequently with a Proxy

[8] Mass scanners scan the entire IPv4 address space in hours to days (depending on the number of ports scanned), but they cannot scan the IPv6 address space in any reasonable time.

4 Cyberattacks on devices

The altOS platform is designed to avoid and prevent normal cyberattacks and targeted cyberattacks on devices and implements a multi-layer defense strategy that includes:

- **Outer layer defenses** – these prevent the identification and digital tracking of devices that may lead to subsequent stages of targeted attacks on devices and/or users., and
- **Inner layer defenses** – these include protections against a typical cyberattack on a device and the later stages of a targeted cyberattack, should the outer layer defenses have been compromised through digital tracking or another vector.

Note that there may be multiple layers of defense within the outer and inner layers. For example, the DNS filter is an outer layer defense against tracking and profiling via Ad Tech data, while administrator control over the Android Advertising ID (AAID) is a second outer layer defense (e.g., in the event there is a third-party tracker domain that is not included in the DNS filter list).

The following labelling conventions apply:

- The platform leverages existing Android and Linux security functions. Security functions or defenses that are intrinsic to Android or Linux are denoted with "(**AOSP**)".
- Defenses that are currently being implemented or planned for a future release of the product are denoted with "(FR, TIME)", where TIME is the expected release date,
- Defenses that require administrator configuration to activate are in blue font (e.g., **Disable Wi-Fi**),
- Defenses that require both administrator configuration and user action are in red font (e.g., **Discreet Mode**), and
- Defenses that do not require administrator or user action are in black font (e.g., **API filtering**).

Table 3. Outer layer defenses

Tracking vector	Details	Countermeasures
AdTech data	Multiple ad libraries are embedded in apps and websites that send bid stream data to electronic ad auctions, where the bid stream data is freely available to hundreds or thousands of companies around the globe. This data may be used to: • Determine where a mobile device user works, • Determine where the user lives, • Identify other personal and family devices used at home, • Identify co-workers, • Identify patterns of life, and • Identify covert locations.	**DNS filter –** The device's operating system features an embedded DNS filter that blocks DNS queries to all known third-party app and website trackers, as well as all major social media domains, using a DNS filter list included with the altOS license. **Zero out AAID** – a policy control to set the AAID to be all zeros ("000..000"). This is the standard value for an AAID when a user opts out of targeted advertising on Android and IOS devices. Setting the AAID to all zeros is a second layer defence against AdTech vector. DNS filtering is the primary defense. **DNS filtering –** the altOS operating system filters data returned from calls to getSystemAvailableFeatures to ensure that apps and third-party trackers cannot identify the underlying OS as altOS. It also filters information returned from getInstalledApps and getInstalledPackages with the same objective. **READ_PRIVILEGED_PHONE_STATE** (AOSP) – Android permission that limits access to persistent identifiers, such as IMEI and serial number, to privileged system apps and apps signed with the vendor key. **Disable Location** (AOSP) – the pulldown message shade on Android devices enables the user to toggle Location ON or OFF. If the administrator allows Location access on the device, via policy, and the user requires GPS location only sporadically, we recommend turning **Location** OFF when the user does not require it. This is another line of defence against tracking.
Cellular networks & rogue cell sites	Device location and activity can be tracked through connections to cell sites and rogue cell sites (e.g., IMSI grabbers or stingrays), and this data can be analyzed. Persistent cellular identifiers (IMEs and IMSIs) can be used to track location and activity over a long period.	**Cellular Identity Management** – these capabilities are implemented on a subset of smartphone models supported for altOS. Please contact CIS Secure for more information. **Discreet Mode –** administrator-defined "dark mode" that may be triggered by the device user to shut down device sensors and modems during periods of elevated threat (e.g., when the user is near a known government facility or a covert facility, meeting other similar users, etc.)**.**
	IMEIs include Type Allocation Codes (TACs) that can be used to identify device OEM and model, and IMSIs contain codes that identify the issuing	**Phone profiles (FR, Q1 2026)** – Phone profiles can be created by the user with unique IMEI and IMSI pairs to enable a user to switch to a phone profile that "blends" in when roaming onto an overseas cellular network.

	cellular network provider. A mobile operator may use these to identify foreign devices roaming on their network.	
	Mobile operators pre-install apps on smartphones they resell to consumers. Like any apps, these may contain third-party advertising or analytics libraries and be used for tracking purposes.	**altOS system image** – CIS Secure does not pre-install any mobile operator apps in the device system partition, and the installation process overwrites any pre-installed apps that may have been present in the stock system image.
System Services	**Enrollment and OTA updates** – using services hosted by the device OEM or a mobile operator for device enrollment and managing OTA updates, shares device identifiers and other sensitive information with the service provider.	**Private OTA** – devices do not receive system image updates from a mobile operator or OEM-hosted OTA update server. The Backend server includes a private OTA update server that allows administrators to schedule and push system image updates to devices in the field. **Private Enrollment** – no device or user identifiers are shared with CIS Secure or any third party during device enrollment with the Backend server.
	Google Mobile Services – apps such as Chrome, Google Play, and YouTube may share sensitive information with Google. Additionally, like most other apps, they contain third-party advertising and analytics libraries.	**Disable Google Apps** – these can be disabled via the container policy controls to prevent any tracking or data leakage that may occur (e.g., work contacts, work accounts) if these apps were present in a container. If disabled, execution of Google apps is prevented by an altOS root daemon. **Disable Location History (FR, 2026)** – Admin policy control to disable Google Location History tracking, eliminating the need to disable it via device or account settings manually.
	Network Location Provider – The Android Fused Location Provider (FLP) utilizes the Google Network Location Provider (NLP) and GPS to estimate the device's Location. Using the NLP requires devices to share nearby cell towers and Wi-Fi access points. These can be used to estimate device location, especially in cases where GPS doesn't function (e.g., inside buildings).	**Disable NLP** – The administrator can disable access to the Google Network Location Provider (NLP) service to prevent location tracking by the NLP service. We recommend disabling the NLP unless there is a requirement for a user to have location data when GPS is not available (e.g., in buildings).
	Cloud backup – when Google Cloud Backup is enabled, the device will back up the following data automatically: • Apps and app data for apps installed from Google Play • Call history – incoming, outgoing, and missed calls. • SMS and MMS messages • Photos and videos	**Disable Cloud Storage (FR, 2026)** – although there is a local device setting to disable this, users may be prompted frequently to enable cloud backup, potentially allowing it by mistake or out of frustration. An administrator policy setting will be provided in a future release to allow remote control of this setting. **Block Google Cloud backup** – The DNS blacklist, in combination with the on-device DNS filter, blocks access to Google Cloud backup domains, preventing any access.

	This will only occur if the container has an active Google account; it may back up sensitive information to the Google cloud.	Note: The DNS filter also blocks access to Dropbox, although this is not a system service. **Disable Google apps** – if Google apps are not enabled in a container, this system service will not be available to the user.
	Web & app activity – this is a Google account setting that is on by default. When enabled or ON, Google collects the following data from the device: • Search queries on Google, click-throughs, and activity on apps like Maps, Play, and News, • Chrome browser history, • App usage, • Voice if you use voice commands with Assistant, and • Location via IP geo-location if GPS location is disabled.	**Disable Web & app activity (FR, 2026)** – while there is a local device setting to disable this, an administrator policy setting will be provided in a future release to enable remote control of this setting. **Disable Google apps** – if Google apps are disabled in a container, web and app activity will not be logged.
Wi-Fi	Mobile devices periodically send probe requests, containing a unique MAC address, to discover nearby Wi-Fi access points, plus they present a unique MAC address when connecting to a Wi-Fi access point. These MAC addresses can be used by nearby scanning devices or access points to track location or identify repeat visits and the time of these visits to places that an adversary may monitor.	**MAC Address Randomization** – the MAC address transmitted in Wi-Fi probe requests is randomized and changes every 24 hours. The Wi-Fi MAC address used to connect to each SSID is randomized and changes every 24 hours. Note: If more frequent rotation of the MAC addresses is required, the operating system provides APIs that can be used to change Wi-Fi MAC addresses more frequently. **Disable Wi-Fi** – the admin can disable Wi-Fi for the entire device. **Disable Wi-Fi Scanning** (FR, Q1 2026) – even if Wi-Fi access is disabled, the Google NLP system service will periodically activate Wi-Fi scanning for crowd-sourcing their NLP database. This should always be disabled. **Discreet Mode –** Discreet Mode can be configured to disable Wi-Fi when triggered by the device user. For example, Discreet mode may be initiated before entering a government facility to prevent an adversary from associating a device with a government facility that an adversary may monitor.

Bluetooth	**Bluetooth Classic – When devices connect to other Bluetooth devices, they transmit a static MAC address. Local scanners can intercept this** MAC address and device name. **Blue Low Energy (BLE) –** device periodically transmit BLE advertisements that include a MAC address and service data. BLE beacons or malicious devices can log these signals to track device movement, estimate proximity (based on signal strength or RSSI), and identify repeat visits to specific locations.	**Disable Bluetooth –** The administrator can enable or disable Bluetooth for each container or the device. Additionally, if Bluetooth is allowed by the administrator, it can be disabled by the user when not required via Discreet mode or the device pulldown menu. **Disable Bluetooth scanning** – even if Bluetooth is disabled, the Google NLP may periodically activate Bluetooth scanning for "crowd-sourcing" NLP data. This should always be disabled. **Discreet Mode** can be configured to disable Bluetooth when triggered by the user. Once triggered, a device remains in Discreet Mode until the user exits Discreet Mode.

Table 4. Inner layer defenses

Threat	Attack	Defenses
Remote attack	**Malware:** **Trojan Horses** – malicious apps that appear legitimate but carry out harmful activities in the background. **Spyware –** software designed to secretly gather user data, including browsing habits, personal info, and communications. **Ransomware –** malicious software that locks or encrypts a mobile device's data and demands payment for its release. **Fast flux attacks –** https://www.cisa.gov/news-events/cybersecurity-advisories/aa25-093a	**App sandbox** (AOSP) – provides process and data isolation between all apps. Apps run in their own VM to provide process isolation and cannot access data or services outside the sandbox without user approval (e.g., via permissions) or unless explicitly allowed by other apps. **HWAC** – access controls in AOSP and OEM versions of Android are enforced at a high level in the operating system. The altOS HWAC (Hardware Access Control) provides a second layer of low-level access controls for select device modems and sensors. **DNS filter** – malware download, C&C servers, and phishing attacks often use free domains and/or domain registration services that offer anonymity (e.g., Freenom). By default, only DNS queries domains with the following TLDs are allowed – ".com", ".org", ".net",".mil", ".org", ".ms". All other queries are blocked unless the domain is specifically whitelisted. **Allow whitelisted domains only –** for malware and spyware prevention and remediation, the DNS filter can be configured to allow access only to whitelisted domains. This is supported operationally by the Management console, which allows the admin to perform app-level and device-level DNS query log analysis. **Discretionary Access Control (DAC)** (AOSP) – the kernel enforces DAC for all apps. Apps are assigned a unique UID, and ownership of files and objects is associated with these UIDs to isolate apps from one another. For example, apps cannot access files and objects associated with another UID without being granted access by the other app. **App Permissions** (AOSP) – these are high-level semantic permissions in the Android OS (such as location or camera access) that an app publisher request via the APK manifest file. Users can selectively grant requested permissions to the APK at runtime or install time. Permissions allow an app to call services outside the application sandbox.

	OS exploits: **Rooting** – attempts to gain privileged control over the Android OS ("root" access), to bypass any built-in security mechanisms. **Privilege Escalation** – attempts to exploit bugs or vulnerabilities in the OS to escalate privilege or gain code execution (e.g., in a system service or the kernel.	**Android Verified Boot** (AOSP) – a secure boot "standard" supported by Android smartphones that cryptographically verifies the bootloader and OS during the boot process. This ensures that only trusted and authentic code executes on the device to protect against persistent malicious attacks and tampering. **ALSR** (AOSP) – Address space layout randomization is a memory protection method for operating systems that provides an MTD against buffer overflow attacks by randomising the locations where system executables are loaded in memory. **Mandatory Access Control (MAC)** (AOSP) – a security model in which the operating system strictly controls how apps, users, and processes can access objects (files, resources, components), based on rules defined by the system admin or OS vendor. Android MAC is implemented in Security Enhanced Android (SEAndroid) and SECCOMP. **SEAndroid** (AOSP) – enhancements made to Android to add SELinux (Security-Enhanced Linux) support for the kernel and user space to: (1) limit what privileged daemons can access or modify, (2) strengthen the application sandbox, (3) prevent privilege escalation, and (4) strengthen data isolation between apps by restricting Inter-Process Communication (IPC) between apps and between apps and system services. **OTA updates** – CIS Secure provides regular updates that include AOSP and CIS Secure security patches. The Backend server consists of a private OTA Update server that provides the administrator with control over device updates.
	Phishing: **SMS Phishing** – Fraudulent text messages designed to trick users into revealing personal information or downloading malware. **Email Phishing** – emails that pretend to be from legitimate sources and contain URLs directing users to fake websites or encouraging them to install malware.	**DNS filter** – phishing attacks often rely on easy-to-register or free domains. By default, only DNS queries domains that belong to the following TLDs are allowed – ".com", ".org", ".net",".mil", ".org", ".ms". DNS queries for domains under any other TLDs are blocked unless specifically allowed via a DNS filter whitelist. **Allow whitelisted domains only** – the DNS filter can be configured to allow access only to whitelisted domains. This is supported operationally by the Management console, which allows the admin to perform app-level and device-level DNS query log analysis. If this is enabled, the TLD-level blocking supported by the DNS filter list is not required.

Untrusted apps	**App spoofing –** Malicious apps that mimic trusted, legitimate applications in order to trick users into downloading them and exposing their data. **Keystroke loggers** – Malware or malicious apps that log keystrokes on a mobile device, stealing sensitive information like passwords or private messages. **Excessive permissions** – Apps requesting excessive permissions (e.g., access to the camera, microphone, or location data) that they don't need, leading to potential privacy violations. **Silent Permissions Changes –** apps may attempt to modify app permissions to gain unauthorized access to sensitive data. **Undocumented APIs –** apps may attempt to access services, modems, or sensors via undocumented APIs that don't have access controls. **Pre-installed apps** – sometimes referred to as "bloatware", pre-installed apps cannot by removed from the device system partition and often request extensive permissions and can operate with a higher level of privilege than other apps. Pre-installed apps may be a vector for malware and spyware.	**App sandbox** (AOSP) – provides process and data isolation between all apps. Apps run in their own VM to ensure code execution is isolated and cannot access data or services outside the sandbox without user approval (e.g. via permissions) or unless explicitly allows by other apps. **App Management** – app sources, install lists - can be configured and modified at the Container level. **Containers** – isolate apps of different trust levels. For example, installing consumer apps in a separate container mitigates excessive permissions or permission changes (e.g., preventing a consumer app from accessing work contacts or messages). **Discreet Mode –** user-triggered, admin-defined "dark mode" to shut down administrator-defined device sensors and modems during periods of elevated risk. **App Blacklist** – apps defined in the App blacklist are removed, if present on the device, and/or are blocked from installation at a later date. For example, if an app is discovered to contain malware or be the target of frequent attacks (e.g., WhatsApp), it can be included in the App Blacklist. **HWAC** – access controls in AOSP and OEM versions of Android are enforced at a high level in the Android framework. The HWAC (Hardware Access Control) provides a second, low-level layer of access control by shutting down the underlying OS services and HALs or disabling the data flow from/to the underlying operating system services and HALs. **System image** – the altOS system image has no pre-installed apps, except a limited number of Google mobile apps (e.g., Chrome, Play, Maps). These may be disabled or enabled in specific containers by the administrator. A root daemon process enforces the administrator's settings.
Untrusted networks	**Passive eavesdropping** – an attacker intercepts data transmitted over an insecure or public Wi-Fi network (e.g., open hotspots) to gain access to sensitive information.	**Always on VPN** – a VPN app can be defined in a container policy, and all traffic to /from a container can be forced to transit the VPN. **Other** – modern browsers, by default, will not connect to websites that don't implement HTTPS without the user explicitly choosing to bypass a browser warning screen.

MITM attack – an attacker secretly intercepts and potentially modifies communications between two parties that believe they are directly communicating with each other.	**Restrict Wi-Fi Access** – Setting this to ON limits the Wi-Fi SSIDs the device can access to a list predefined by the administrator – the *Wi-Fi configuration list*.

5 Physical control of devices

While the definition of continuous physical control of a device depends on the threat environment and mission, for any mobile device we have to assume that a user may lose or misplace a device, the device may be stolen, or the user may have to relinquish physical control of a device temporarily.

In the worst-case scenario, a user may be required to provide their unlocked device to an authority at a border crossing for inspection and/or forensic extraction of data. This may lead to further device analysis or deeper questioning directed at the user in terms of their employer, job function, or a specific mission.

The following table lists the defenses against loss of device physical control and potential forensic analysis or extraction that could occur under this worst-case scenario.

Table 5 Loss of Continuous Physical control

Threat	Details	Defenses
Physical access to locked device	**Brute force attacks – m**obile devices with weak or easily guessed passwords/PINs may be unlocked and access by an adversary that can guess the password or PIN.	Policies controlled by the admin define the user authentication and access controls for each container: **Authentication** – PIN, Password, Strong Password. **Minimum Length** – the minimum acceptable password or PIN length. **Expiration** – the maximum time a PIN or Password may be used before the user must change it. **Enforce Password History** – prevents re-use of previous PINs or Passwords upon expiration and the subsequent creation of a new PIN or Password by a user. **Wipe container after X failed attempts** – wipes the user data in a container after "X" failed authentication attempts. **Time to Lock Screen** – the period of inactivity in the foreground container until the screen is locked. **Keystore (AOSP)** – PINs and passwords are encrypted using a key stored in the Android keystore which is further protected by a secure element or TEE.

	Shut down on Exit – containers may remain active in background mode meaning CE data can still be decrypted if the container can somehow be accessed. This container policy setting will relock the container CE data (see below).
DAR (Data at Rest) – sensitive data stored insecurely in apps (e.g., plain text passwords, unencrypted data) which can be exploited if the device is compromised.	**File Based Encryption (FBE)** (AOSP) – encrypts data in storage. Every file is encrypted with a unique key when written to the file system and decrypted when loaded into memory. There are two types of data: • **Device Encrypted (DE) data** – files can be decrypted immediately after the device boots using a unique DE key. The DE key is stored in hardware and automatically derived on boot. Apps, such as alarms, that require data access upon boot must be designated as "Direct Boot aware" in their manifest file and must explicitly choose to access DE storage. • **Credential Encrypted (CE) data** – the CE Keys used to decrypt CE data are derived from multiple inputs, including the user credentials. CE data encryption protects sensitive user data by offering higher security, as the CE key is bound to the user's credentials for the container. **File Metadata encryption (AOSP)** – file metadata is also encrypted using a metadata encryption key that is stored in hardware and accessible after device boot. **Remote wipe** – a command that deletes a container and its associated user data. This command also erases the credential encryption (CE) and device encryption (DE) and metadata decryption keys for the specified container. If a remote wipe command is sent to the primary container, the device will be factory reset, so the CE, DE, and metadata decryption keys for each container will be wiped. **USB file transfer=OFF** – this container policy setting prevents USB file transfer to prevent data from being transferred from the device to another device or storage medium. **Mounting physical media=OFF** – this policy setting prevents mounting external media, such as an SD card or other USB storage device, to prevent an adversary from sideloading a forensic agent (e.g., Cellebrite or Magnet) or a custom recovery image.

Physical access to an unlocked device	**Device inspection** – An authority (e.g., a border guard or law enforcement officer) may single out a device for inspection and force the user to unlock the device. The presence of specific apps or data on a device or the appearance of the device may lead to deeper questioning, inspection, and potentially forensic analysis. **Forensic analysis** – an authority may use forensic analysis tools in an attempt to access data on a device. We assume the authority can compel the user to unlock the device – a prerequisite for forensic analysis. Data extraction can be: • **Logical** – the forensic analysis tool communicates with the OS to access contacts, call logs, messages, and files stored in user accessible areas. • **File system** – this is a more comprehensive data collection method that accesses the device's file system more directly to retrieve files and data that are not visible to the user, including system logs and cached data. • **Physical** – This is the most invasive method, performing a bit-by-bit copy of the device's flash memory to recover deleted files and data in unallocated space. The utility of this method on its own has diminished significantly with FBE on Android devices, as the data extracted from flash memory remains encrypted if extracted at this level.	**Form factor** – the altOS operating system is supported on stock consumer smartphone hardware. This means that altOS-enabled smartphones "blend in" or look like a standard consumer smartphone (a "black bar"). **Tells** – we continually work to minimize any visible "tells" that may allow a third party with access to an unlocked device to distinguish it from a standard Android consumer smartphone. **Green boot** – the altOS operating system replaces the stock Android operating system on supported device models (a "custom ROM"). The Android Verified Boot (AVB) specification defines the yellow boot" flow for custom ROMs. We work with OEMs to implement the "green boot" flow on select smartphone models. However, this requires technical cooperation that may supported by the OEM. **Hidden containers** – apps and other data that may act as tells during can be installed in hidden containers that remain locked or in background during device Inspection. While the Primary container on the device (e.g., the user space that a User would present to an authority on demand) can be configured to look like a typical consumer smartphone. **Wipe prefix –** an administrator can define a wipe prefix that can be appended to the primary container password or PIN to trigger the deletion of hidden containers. For example, if the Wipe prefix is "70" and the PIN for the primary container is "4987", the user would enter "704987" to wipe the hidden containers and open the Primary container. In this case, the Primary container is assumed to be innocuous in terms of the apps and data that may be present in it. **Container backup & restore** – The user can back up a container to a designated storage location. The user data is encrypted and uploaded to the designated storage, and the KEK and metadata that define the container are wiped. The user may later re-install the container on the device via a settings menu that is hidden in device settings. For example, prior to travelling to a foreign country, the user can trigger the container backup and wipe function and wait to restore the container until they return home. **Disable USB Debug** – ADB is a versatile command-line tool intended for use by software developers (to install and debug software on Android devices, transfer files, view device logs, access hidden features, and run shell commands) but is also a common attack vector for forensic analysis tools. The administrator can disable developer options on devices via a policy setting to prevent ADB from being used. When this setting is enabled, the ADB daemon in the kernel is shut

	down. Bypassing this low-level access control would require an adversary to gain root access or privileges. **Wipe on USB debug** – as an alternative to disabling USB debug (as this may serve as a tell to a knowledgeable adversary), this setting can be enabled by the administrator to "arm the device". Once armed, any attempt to enable USB debugging will wipe the hidden containers on the device.

6 Cyberattacks on the Servers

Similar to the previous section, this focuses on server threats and countermeasures and describes:

- **Outer layer defenses** – countermeasures that prevent the identification and reconnaissance of servers via tracking vectors and other OSINT to prevent targeted attacks on the servers, and
- **Inner layer defenses** – defenses against a normal or targeted cyberattack on a server.

Defenses that are currently being implemented or planned for a future release of the product are denoted with "(FR)".

Table 6. Outer layer defenses

Threats	Details	Countermeasures
OSINT	**CT Logs** – publicly accessible, append-only ledgers that record information about all publicly issued TLS certificates. CT logs were created to enhance the security and trustworthiness of the TLS ecosystem by providing transparency and the ability to monitor the issuance of certificates by public CAs[9]. Web browsers (e.g., Chrome, Apple Safari) now require CAs to submit all publicly issued SSL/TLS certificates to CT logs for the certificates to be considered valid and trusted by the browser. CT logs are publicly accessible, monitored in real time by Internet scanners[10]They are a potential reconnaissance vector for cyberattacks on Internet-connected hosts. Internet scanners, such as Censys, use CT logs to augment their ASM data and to direct their scanners (e.g., for IPv6 addresses).	**Self-signed certificates** – the Proxy and Backend servers can be provisioned with self-signed TLS certs (recommended)[11]. Using self-signed certificates or private CA certificates eliminates OSINT that would otherwise be available via Certificate Transparency (CT) logs.

9 https://certificate.transparency.dev/howctworks/

10 Certifiably Vulnerable: Using Certificate Transparency Logs for Target Reconnaissance, 2023 European Symposium on Security and Privacy.

11 The use of self-signed certificates is common for Internet-attached hosts - a recent Censys search (27-May-2025) identified 26.9% (1.266 million) self-signed certificates in a total sample of 4.555 million unexpired TLS certificates of the Censys certificate database.

	Domain registration requires sharing account and payment information, as well as the authoritative DNS name servers for the domain. Domain information may be accessible via WHOIS or via TLD zone files. **Public DNS** – relying on public DNS to resolve IP addresses requires domains and sub-domains to be registered with one or more name servers. Domains may be leaked via TLD zone files and sub-domains may be leaked via mass DNS scanning or CT logs[12]. For example, SecurityTrails, a threat intelligence company, has a database containing trillions of DNS records dating back to 2008.	**Not recommended or required** – altOS do not use public DNS to resolve server addresses, so Proxy and Back server domain names and the authoritative name server(s) do not need to be registered. Eliminating domain registration eliminates OSINT and other private information, such as account information, which might be obtained from a co-opted or corrupt TLD registry or registrar. **Local DNS** –devices and servers resolve IP addresses locally versus having to access a public DNS resolver using information contained in a device Proxy profile or information contained in a server /etc/hosts file. The only time an adversary may observe a server domain name is if they are able to monitor a TLS handshake between a device and a Proxy server – under this scenario the adversary would capture the Proxy SNI. Note. The SNIs for the Backend servers are encrypted in the HTTPS packets sent from devices to Proxy servers and are assumed to be inaccessible to a threat actor. Please contact CIS Secure for recommendations on Proxy server SNIs.
Scanners	Mass scanners, such as Censys, and other unknown scanners can identify internet-connected hosts within 1 hour of connecting to the Internet. Scanners may start with an ICMP (Internet Control Message Protocol) scan to identify live hosts before performing more intrusive scans, such as port scanning. Once a port is found to be open, scanners will attempt to determine what service is running and will try to collect metadata including: HTTP banners and headers, TLS certificate chains and parameters, SSH key fingerprints and banners, etc. The resultant scan data, along with CT Logs and other OSINT, can be used to identify and profile Internet-connected servers.	**ICMP scanning** – both AWS and Azure block inbound ICMP scanning by default. Blocking ICMP is a common cloud security defense to reduce the attack surface and avoid reconnaissance that is often a precursor to a cyberattack. **Backend server configuration** – access to the Backend server should be limited to Proxy server IP addresses and a VPN IP address used for admin access to prevent scanning. If possible, the Backend server should reside in a private subnet. **TLS handshake (FR, Q2 2026)** – the Proxy server will be configured to provide a default "loopback" certificate if any third-party device (or scanner) initiates a TLS handshake without the Proxy SNI. The loopback certificate is consistent with 125 M other similar certificates in the Censys database.

[12] For example, SecurityTrails, a threat intelligence company, has a database containing trillions of DNS records dating back to 2008.

		Proxy headers – the Proxy service headers are the default NGINX HTTPS response headers and provide no information that would enable an adversary to de-anonymize a Proxy.
		DNS mass scanning – the altOS platform does not use public DNS resolvers or nameservers to resolve server IP addresses. Proxy server IP addresses are defined in an administrator-defined "Proxy Profile," and the Backend server IP addresses are determined using information in the /etc/hosts[13] file.
Network surveillance	There are two levels of surveillance: • **Local traffic monitoring** – adversary is able to capture traffic crossing a network LAN segment. For example, an adversary may have access to a Wi-Fi access point (AP) at a coffee shop and have the ability to observe metadata in the traffic sent by a device through the AP. • **National networks** – the adversary has the capabilities of a national censor with surveillance capabilities over all ISPs and backbone routers in a country. In addition to monitoring and analyzing traffic patterns and metadata, the adversary may also be able to block traffic based on IP addresses, SNI, or port and to inject or modify traffic, such as DNS queries.	**Reverse Proxy servers** – obfuscate device connections to the Backend server. Only the Proxy IP address and SNI can be observed by an adversary that can monitor network traffic. The Backend server SNI is encrypted in the HTTPS payload between a device and a Proxy server. **Connectivity controls** – device connectivity to the Backend (via a Proxy) can be triggered by the user periodically, connect intermittently (every x hours), or maintain a persistent (e.g., a web socket) connection with a Proxy server. For connections made on domestic networks, it may be desirable for the device to maintain a persistent connection with its assigned Proxy (to enable real-time logging or nearly instantaneous policy changes), while for connections made across adversary-controlled networks, it may be desirable to have infrequent, aperiodic connections to the assigned Proxy server. **Burner Proxies** – a Proxy profile may define multiple Proxies and the start date/time at which the device will "home" onto different Proxies listed in the Profile. Switching is automatic based on the predefined schedule. This can be used to implement a moving target defense (MTD) against traffic monitoring.

[13] The /etc/hosts file is a plain text file used by operating systems (including Linux and Windows to map IP addresses to domains and acts as a local DNS service that takes precedence over mappings provided by network DNS resolvers.

		Location-based Proxies – A Proxy profile may point to a Proxy in an offshore AWS or Azure data center for specific intervals (e.g., while a user is traveling) to avoid associating a device with domestic locations and IP addresses.

Table 7. Inner layer defenses

Threat Vector	Attack	Defenses
Unauthorized access	An insider could misuse access to the Management console to perform tasks they are not authorized to perform.	**RBAC** – different administrator roles are defined in the console. The administrator role is assigned by an administrator with the highest level of privilege (tenant administrator) when an account is created or added to the console. **Two-factor authentication** – administrator login to the console requires a valid password and a second factor of authentication, such as a hardware token. The second factor can be any factor supported by Keycloak.
Remote attacks	An adversary may launch a remote attack on the Backend server or Proxy servers with the objective of establishing presence on a server. This presence or control could be used to modify device policies, identify devices and users, or to interfere with a mission.	**Nessus scans** – CIS Secure regularly perform Nessus scans on the altOS servers to check for software vulnerabilities and missing software patches. **Software updates** – CIS Secure provides regular updates to the Proxy and Backend server software to address potential vulnerabilities that may be reported. These can be implemented by uploading the software to a control host that is created during server installation process. Note that the control host can be shut down or the cloud VM used by the Control host can be placed in hibernation when it is not in use. **Administrator access** – In the recommended deployment model, administrator access to the Backend server must take place over a VPN, and access to the Backend is limited to the VPN gateway IP address. **Reverse Proxies** – devices communicate with the Backend server via a Proxy to isolate the Backend server from the Internet. We recommend that access to the Backend server be limited to Proxy server IP addresses. Additionally, we recommend using VPC peering (AWS) or VNet peering (Azure) to connect Proxy servers to the Backend server. This eliminates any need for the Backend server to have a public IP address.

		Device authentication (FR, Q1 2026) – the TLS handshake between devices and Proxies will be extended to support device authentication to the server via mutual TLS. **TLS handshake (FR, TBD)** – the Proxy server will be configured to provide a default "loopback" certificate if any third-party device (or scanner) initiates a TLS handshake without the Proxy SNI. The loopback certificate is consistent with 125 M other similar certificates in the Censys database.
Malicious enrollment	An attacker may attempt to register a rogue device that could receive sensitive configurations or apps or that may serve as a launch point for further attacks on the servers or devices.	**User Invitations** – An administrator creates invitations (QR codes) via the management console for enrollment purposes. Invitations can be assigned to specific users and require entry of a password to prevent unauthorized use of an invitation. **Enrollment limitations** – Enrollment is limited to specific device models supported by altOS, which must have the proprietary altOS operating system installed. This is necessary to enroll with the Backend and process policies and configuration data sent to the device. **One-time use Invitations** (FR, 2026) – invitations can be configured to "expire" after the initial use. **Device identifiers (FR, TBD)** – the Backend servers limit device enrollment to devices that provide a "valid IMEI" during the enrollment process. A valid IMEI matches an IMEI on a list of devices imported to the Backend server by an administrator.
MitM attack	A man-in-the-middle attack (e.g., to impersonate a Proxy to a device) would enable an adversary to view unencrypted traffic between a device and a Proxy or to inject traffic into the communications between a device and a Proxy server.	**TLS v1.3 –** The device uses TLS v1.3 encryption for proxy connections. The server is authenticated using a TLS cert that is validated against the device root store. Note: If the server certs are self-signed (recommended) or signed by a private CA, the TLS cert is validated against the private CA cert or self-signed cert provided in an invitation (during enrollment) or in the Proxy profile (post enrollment). If proxy certificates are signed by a public CA (not recommended), the certificate presented by a Proxy will be validated using CA certificates in the main device root store. **Local DNS** – devices will only connect to the Proxy server defined in an administrator-defined Proxy Profile and don't rely on public DNS to resolve the IP address of the Proxy. This effectively prevents MITM attacks that rely on DNS spoofing.

	The certificates used by devices to authenticate Proxy servers are validated using a root CA certificate in the device operating system root store. The default AOSP root store contains 150 to 170 root CA certificates from CAs around the globe. A certificate from a compromised or untrusted CA could be presented to a device in an attempt to implement a MitM attack.	**Root store[14] whitelist (FR, Q4 2025)** – This is an administrator-defined Root store whitelist that enables an administrator to limit the trusted public CAs to decrease the attack surface for a MITM attack.
Denial of Service	An adversary with control of a network (e.g., censor capabilities) may block or modify DNS queries to prevent device connection to the Proxy server or block traffic based on the IP address of the Proxy server.	**Local DNS** – devices use information contained in Proxy Profiles to connect to the assigned Proxy – they do not access external DNS resolvers for this purpose. This prevents DNS blocking, modification, or tampering attacks that could be used to deny access to the Proxy and Backend servers. **Proxy Failover** – when a Proxy server is created, one or more failover Proxy servers can be assigned with different static IP addresses. If a device cannot connect to the Proxy server defined in its Proxy Profile within 90 seconds, the device will connect to the failover Proxy. If a device cannot connect to a failover Proxy, it will connect to the next failover Proxy and so on.

[14] The default AOSP list of trusted CA credentials contains approximately 150 CA certificates, of which five public CAs (Let's Encrypt, GlobalSign, Sectigo, GoDaddy, and DigiCert) account for over 99 percent of SSL certs.