Persistence, Tactic TA0028 - Mobile

Archived: 2026-04-05 15:32:51 UTC

The adversary is trying to maintain their foothold.

Persistence is any access, action, or configuration change to a mobile device that gives an attacker a persistent presence on the device. Attackers often will need to maintain access to mobile devices through interruptions such as device reboots and potentially even factory data resets.

ID: TA0028

Created: 17 October 2018

Last Modified: 25 April 2025

Techniques

Techniques: 8

ID	Name	Description
T1398	Boot or Logon Initialization Scripts	Adversaries may use scripts automatically executed at boot or logon initialization to establish persistence. Initialization scripts are part of the underlying operating system and are not accessible to the user unless the device has been rooted or jailbroken.
T1577	Compromise Application Executable	Adversaries may modify applications installed on a device to establish persistent access to a victim. These malicious modifications can be used to make legitimate applications carry out adversary tasks when these applications are in use.
T1645	Compromise Client Software Binary	Adversaries may modify system software binaries to establish persistent access to devices. System software binaries are used by the underlying operating system and users over adb or terminal emulators.
T1624	Event Triggered Execution	Adversaries may establish persistence using system mechanisms that trigger execution based on specific events. Mobile operating systems have means to subscribe to events such as receiving an SMS message, device boot completion, or other device activities.
.001	Broadcast Receivers	Adversaries may establish persistence using system mechanisms that trigger execution based on specific events. Mobile operating systems have means to

ID	Name	Description
		subscribe to events such as receiving an SMS message, device boot completion, or other device activities.
T1541	Foreground Persistence	Adversaries may abuse Android's `startForeground()` API method to maintain continuous sensor access. Beginning in Android 9, idle applications running in the background no longer have access to device sensors, such as the camera, microphone, and gyroscope. Applications can retain sensor access by running in the foreground, using Android's `startForeground()` API method. This informs the system that the user is actively interacting with the application, and it should not be killed. The only requirement to start a foreground service is showing a persistent notification to the user.
T1625	Hijack Execution Flow	Adversaries may execute their own malicious payloads by hijacking the way operating systems run applications. Hijacking execution flow can be for the purposes of persistence since this hijacked execution may reoccur over time.
.001	System Runtime API Hijacking	Adversaries may execute their own malicious payloads by hijacking the way an operating system runs applications. Hijacking execution flow can be for the purposes of persistence since this hijacked execution may reoccur at later points in time.
T1676	Linked Devices	Adversaries may abuse the "linked devices" feature on messaging applications, such as Signal and WhatsApp, to register the user's account to an adversary-controlled device. By abusing the "linked devices" feature, adversaries may achieve and maintain persistence through the user's account, may collect information, such as the user's messages and contacts list, and may send future messages from the linked device.
T1603	Scheduled Task/Job	Adversaries may abuse task scheduling functionality to facilitate initial or recurring execution of malicious code. On Android and iOS, APIs and libraries exist to facilitate scheduling tasks to execute at a specified date, time, or interval.

Source: https://attack.mitre.org/tactics/TA0028