Path Traversal to Arbitrary File Write in Apktool 3.0.1 (Regression in commit e10a045)

High **iBotPeaches** published **GHSA-m8mh-x359-vm8m** 2 days ago

Package

org.apktool:apktool-lib (Maven)

Affected versions	Patched versions
>= 3.0.1	3.0.2

Description

A path traversal vulnerability in `brut/androlib/res/decoder/ResFileDecoder.java` allows a maliciously crafted APK to write arbitrary files to the filesystem during standard decoding (`apktool d`). This is a security regression introduced in commit e10a045 (PR #4041, December 12, 2025), which removed the `BrutIO.sanitizePath()` call that previously prevented path traversal in resource file output paths.

An attacker can embed `../` sequences in the `resources.arsc` Type String Pool to escape the output directory and write files to arbitrary locations, including `~/.ssh/config`, `~/.bashrc`, or Windows Startup folders, escalating to RCE.

Fix: Re-introduce `BrutIO.sanitizePath()` in `ResFileDecoder.java` before file write operations.

Severity

High 7.1 / 10

CVSS v3 base metrics

Attack vector	Local
Attack complexity	Low
Privileges required	None

User interaction	Required
Scope	Unchanged
Confidentiality	High
Integrity	High
Availability	None

Learn more about base metrics

CVSS:3.1/AV:L/AC:L/PR:N/UI:R/S:U/C:H/I:H/A:N

CVE ID

CVE-2026-39973

Weaknesses

▶ CWE-22

Credits

caveeroo — Reporter

IgorEisberg — Remediation developer