29 OCTOBER 2026 / ATTACK ANALYSIS

What happened at Binance?

The path from developer laptops to wallet signing operations.



$6.34B

Approximate value of digital assets impacted, according to Binance's initial report.

How did the chain unfold? Where could an earlier warning have changed the response?

01 / THE TIMELINE

It wasn’t one breach.



JUNE

Tampered tool installation

A’s tampered Homebrew install was detected. Device replaced; credentials rotated.



SEPTEMBER

Fake recruitment interview

A fake interview assignment ran malware during setup on B’s work computer.



20 OCTOBER

Internal AI tool as a springboard

Access expanded from B to A and C. A held a deployment token.



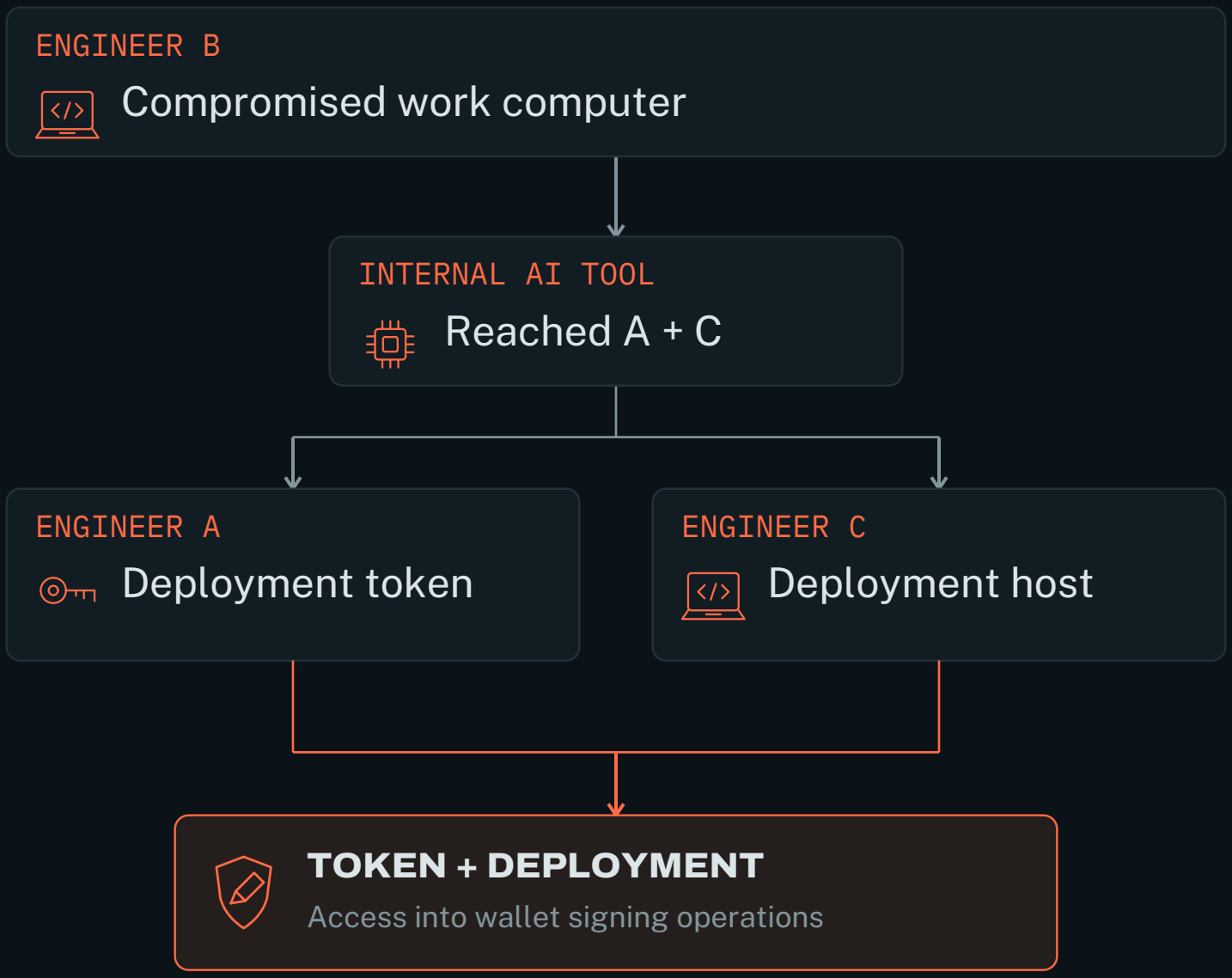
29 OCTOBER

Into wallet signing

From C, the attacker used A’s token to deploy a container and run malicious code.

02 / ACCESS EXPANDS

One foothold. Multiple paths.



03 / DEVELOPER TOOLING

A familiar path. A replaced program.

The malware replaced pyenv inside a familiar developer-tool path.

01 `pyenv_` 3,414 B

Replacement file created

02 `pyenv` 3,344 B

Legitimate file deleted

03 `pyenv_ → pyenv` 3,414 B

Replacement took its place

04 `symlink + zsh`

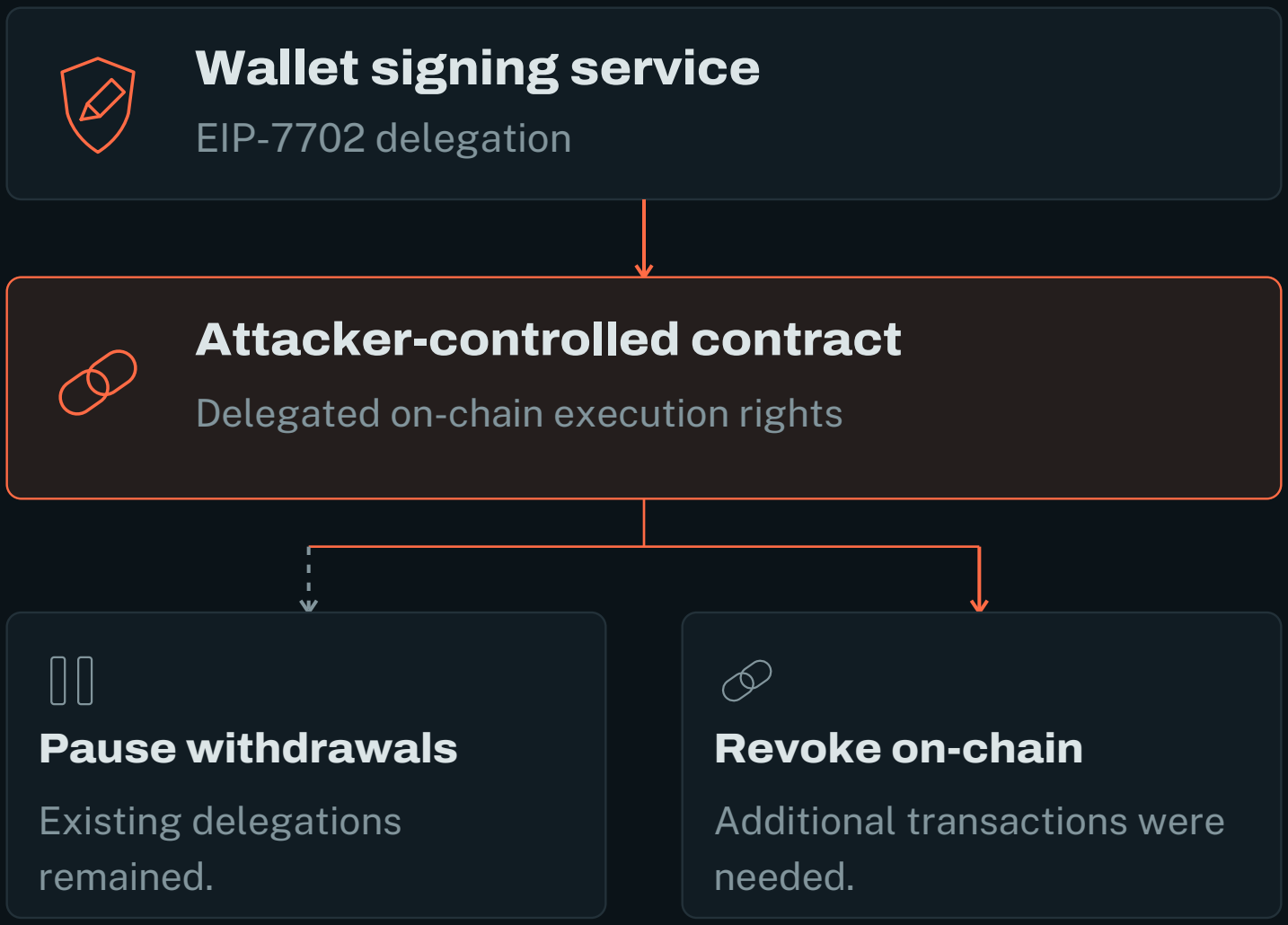
Symlink recreated; shell spawned

Binance says endpoint protection missed the initial malware. The file path still looked familiar.

04 / SIGNING AND ON-CHAIN AUTHORITY

Withdrawals stopped. Permissions remained.

The attacker used the signing service to establish EIP-7702 delegations. Those permissions operated outside the normal withdrawal flow.



05 / AN EARLIER SIGNAL

Turn the search for access into an alert.

FalseTrail could have helped when the attacker searched for and tried credentials.

SEARCH FOR ACCESS

Credentials / configs / service references

FALSETRAIL

Planted decoy

Decoy credential use alerts the SOC and identifies its placement.

THE BINANCE PATH

Production token

Using stolen credentials, the attacker deployed a container on 29 October.

An earlier signal gives the SOC a chance to investigate before the next production step.

06 / THE FALSETRAIL PATH

A decoy. A signal. Evidence of what's next.

01 PLACE

Believable decoy credentials and configs

Identifiable placements with no real production authority.

02 DETECT

Credential use becomes an alert

Send the finding to the security tools the team already uses.

03 OBSERVE

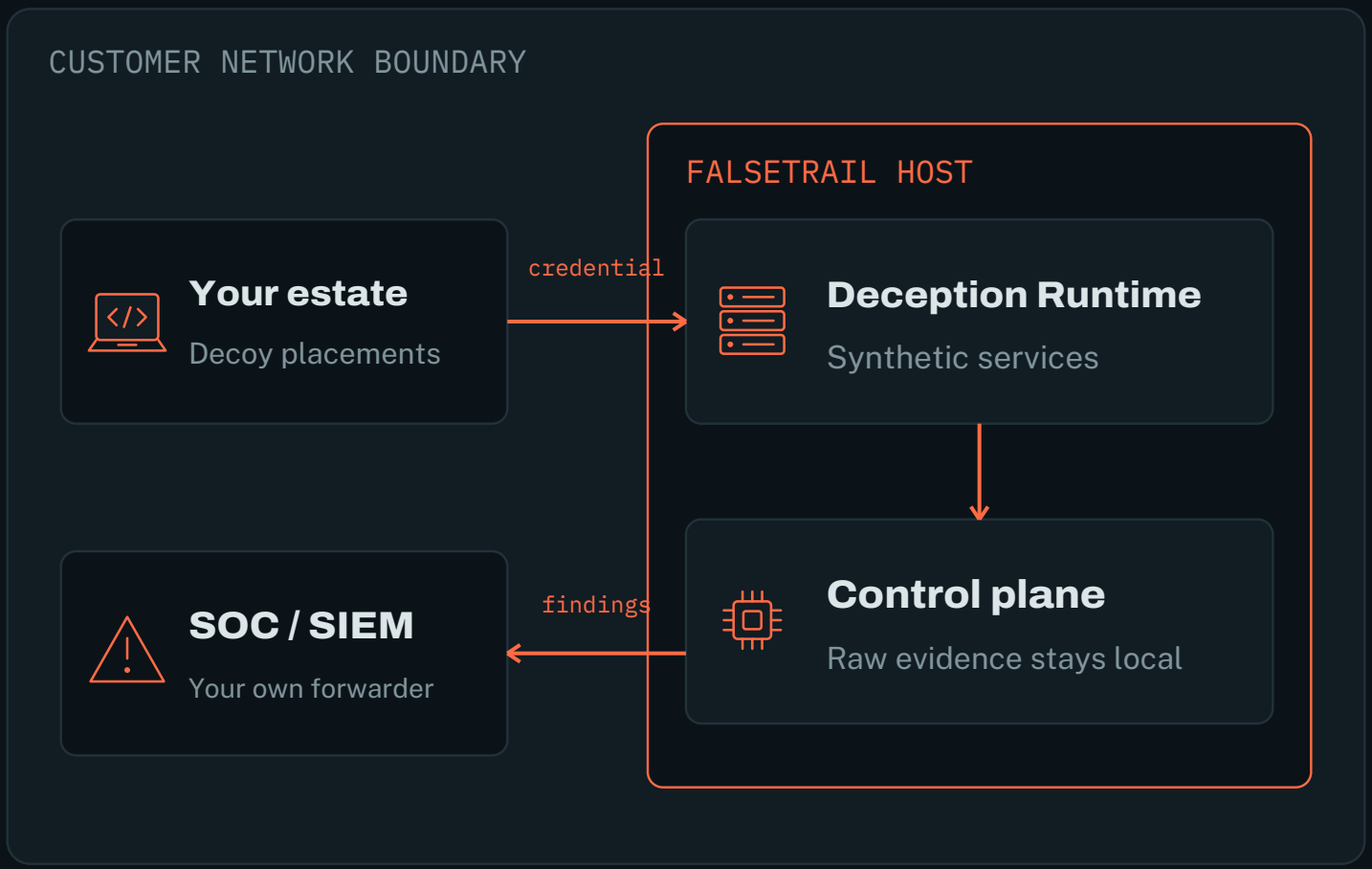
A controlled trail through fake services

If the trail continues, see what they attempt inside those services.

07 / DEPLOYED IN YOUR ENVIRONMENT

Your infrastructure. Your evidence.

On-prem · Your cloud / BYOC · Hybrid



0 **outbound connections**

Raw evidence stays in your environment. Your forwarder decides where findings go.

08 / FOR SECURITY TEAMS

Missing the entry needn’t mean missing the intruder.

An intruder may already be inside. Later use of a planted credential can still expose them.

01

Relevant placements

Place approved decoy configs and service references where an intruder is likely to look for access.

02

A response path

Validate the alert. Investigate the real host, rotate exposed real credentials and assess the scope.

AI makes more paths easier to explore. Deception creates a detection opportunity in the search for access itself.

09 / THE NEXT SIGNAL

See the next attempt at access.

On-prem, your cloud/BYOC or hybrid. Zero outbound connections. Raw evidence stays in your environment.

Explore an evaluation

falsetrail.com

SOURCES

Binance / Initial findings and response update
03 November 2026

FalseTrail / Deployment
Customer-hosted / zero outbound

Google / Mandiant / ARTEX and Friends
AI attack context / 30 October 2026

Independent FalseTrail analysis of public initial findings, as of 4 November 2026. Binance's investigation is ongoing. The security team owns investigation and containment in real systems.