OneInfinity

February 22nd, 2025

OneInfinity Bybit February 2025 Incident Analysis

Disclaimer: The analysis provided in this incident analysis brief from OneInfinity is based on information available at the time of writing. The exact details of the incident and its causes may evolve as further investigations are conducted. The conclusions drawn in this brief are derived from currently available sources and may be subject to change.

Our R&D team conducted an in-depth analysis of the Bybit hack happeing on 21st February, drawing on publicly available data and on-chain information, along with key insights shared by Bybit leadership via social media and livestreams. We sincerely appreciate Bybit's transparency and accountability during this challenging incident, which serves as a valuable lesson for all Web3 participants

Incident Analysis

Below are the key findings from our investigation:

I. Basic Information

- The compromised Bybit multi-signature Safe wallet: 0x1Db92e2EeBC8E0c075a02BeA49a2935BcD2dFCF4

- Bybit Exploiter address: 0x47666Fab8bd0Ac7003bce3f5C3585383F09486E2

- Over 1.4 billion USD worth of assets drained, including the following:

asset	amount	transaction id	estimated value*
ETH	401,347	0xb61413c495fdad6114a7aa863a00b2e3c28945979a10885b12b30316ea9f072c	1,123,770,953
stETH	90,376	0xa284a1bc4c7e0379c924c73fcea1067068635507254b03ebbbd3f4e222c1fae0	253,051,540
mETH	8,000	0xbcf316f5835362b7f1586215173cc8b294f5499c60c029a3de6318bf25ca7b20	22,400,000
cmETH	15,000	0x847b8403e8a4816a4de1e63db321705cdb6f998fb01ab58f653b863fda988647	42,000,000
USDT	90	0x25800d105db4f21908d646a7a3db849343737c5fba0bc5701f782bf0e75217c9	90
		Est. total value	**1,441,222,583**

**assuming ETH, stETH, mETH and cmETH price at $2800 for simplicity*

- On-chain crime investigator ZachXBT submitted a definitive proof to Arkham Intelligence showing that the hack is done by the North Korean Hacker Group named Lazarus (source: Arkham Intelligence)

II. Attack Techniques

This attack is conducted through several steps:

1. Deployment of a malicious implementation contract
    a. The attacker deployed a new, malicious implementation contract (at address 0xbDd077f651EBe7f7b3cE16fe5F2b025BE2969516) on February 19, 2025, at 7:15:23 UTC.
    b. This contract was intentionally crafted to include hidden backdoor functions. Its code contains methods like sweepETH and sweepERC20, which allow anyone who can call them (in this case, the attacker) to transfer out ETH and ERC20 tokens from the wallet.

2. Replacing the Original Safe Implementation

    a. On February 21, 2025, at 14:13:35 UTC, the attacker initiated an upgrade transaction on the wallet. This transaction was approved with signatures from three signers —a requirement for the wallet's upgrade mechanism.

    The transaction (0x46deef0f52e3a983b67abf4714448a41dd7ffd6d32d32da69d62081c68ad7882) replaced the legitimate Safe implementation with the malicious contact in 1a.

    Since many modern wallets (like Gnosis Safe) use an upgradeable proxy pattern, the "logic" of the wallet can be swapped out if the necessary multisig conditions are met.

    b. Signature phishing through Safe App front end hack:
        i. The three signers from Bybit side see a legitimate transaction information on Safe App and they proceed to signing. What they actually signed is the transaction that upgrade the wallet to the malicious contract in point 1a.

        ii. Bybit CEO and co-founder Ben Zhou served as the last signer. He provided several important information in the live-streaming 2 hours after the hack:
            1. He checked the Safe App URL and claim that it should be correct
            2. The information shown on the interface looks like a legitimate transaction

3. Bybit team was performing a routine cold-to-warm wallet transaction
4. He was using Ledger device to sign the transaction. He revealed that a clear signing (you see what you sign) was a bit hard on the device he use as the transaction details display on the hardware is quite packed and hard to read through.

c. Exploiting DELEGATECALL to Embed Malicious Logic: After the contract is upgraded, the smart contract wallet doesn't execute its own code but instead uses DELEGATECALL to execute functions from the malicious contract while keeping its own storage context.

Imagine your wallet is like a smartphone that runs apps (its functions) using software installed on it. Normally, it runs safe, trusted apps. In this case, the wallet uses a method called DELEGATECALL, which is like saying, "Hey, run this app from another source, but treat it as if it's part of me."

What the attacker did was replace the trusted app with a malicious one. Because of DELEGATECALL, even though the harmful code comes from a different contract, it runs as if it were native to the wallet. This lets the bad app access and control the wallet's data and funds without the wallet noticing something is wrong.

3. Draining the Wallet via Backdoor Functions

Because the malicious functions were executed in the context of the wallet (thanks to DELEGATECALL), they had full access to the wallet's funds. This allowed the attacker to systematically drain all assets from the wallet.

The attacker then called the backdoor functions sweepETH and sweepERC20.

- sweepETH: This function transfers all ETH from the wallet to an address controlled by the attacker.
- sweepERC20: Similarly, this function transfers any ERC20 tokens held by the wallet.

III. Is SAFE app compromised?

How the hackers managed to compromise the SAFE app's front end and display legitimate transaction information remains unknown. The precise root cause can only be determined through further forensic analysis.

Bybit CEO and founder Ben Zhou suggested two potential root causes in his live-streaming:

- A compromise of Safe Wallet's infrastructure or back end.
- A compromise of Bybit signer's laptop or signing device.

According to Safe Wallet's latest official post (source: Safe.eth X thread), their internal investigation has not found any evidence of a codebase breach, malicious dependencies, unauthorized access, or any other impact on Safe Wallet.

Similar Incidents

Over the past 12 months, several incidents have exhibited similar patterns and attack techniques. Notable examples include:

- The DMM Crypto hack (approximately $305 million; involving a Ginco multi-sig wallet)
- The Radiant Capital hack (approximately $50 million; involving a SAFE multi-sig wallet)
- The WazirX hack (approximately $230 million; involving a SAFE multi-sig wallet)

Each of these incidents involved a compromise of the wallet interface's front end, which displayed legitimate transaction information to trick signers into approving malicious transactions.

Furthermore, these cases have shown that attackers often employ social engineering techniques to infiltrate the internal environments of exchanges, custody services, or wallet technology providers. By installing malicious code and applications, attackers can gather operational intelligence, move laterally within networks, and gain access to critical systems to stage the final attack. It is possible that similar techniques were used in the Bybit attack as well.

Security Advice – How to Prevent Similar Incidents from Happening

- **Verify Transactions Independently**
  Signers should not rely solely on what is displayed by the wallet's front-end interface. Instead, it is advisable to use hardware wallets or dedicated devices that independently verify and display transaction details. This ensures that the information being signed matches the intended transaction, reducing the risk of approving a malicious transaction that may have been altered by compromised front-end software.

- **Enhance Endpoint Security**

  Robust endpoint security is critical in defending against attacks that seek to install malicious software on devices. Enforcing software whitelisting, maintaining up-to-date antivirus and anti-malware solutions, and regularly patching software vulnerabilities can significantly reduce the risk of unauthorized code execution. These measures help ensure that even if an attacker attempts to breach a device, the environment is hardened against the installation and operation of malicious tools.

- **Strengthen Management & Employee Security Awareness:**
  Provide ongoing training for employees to recognize phishing attempts, social engineering tactics, and other common attack vectors. An informed team is better equipped to avoid actions that could inadvertently compromise security. Emphasize the importance of clear signing!