Z E R O T R A C E L A B S

A T T A C K C A S E S T U D Y

GreedyBear Malicious Firefox Extensions Campaign: A Supply-Chain Phishing Operation That Drained $1M Across Multiple Chains

Over 150 counterfeit browser extensions impersonating MetaMask and Coinbase Wallet weaponized the Firefox Add-ons ecosystem to harvest seed phrases at scale.

Attack Snapshot

victim	Individual crypto users across multiple chains (Ethereum, Bitcoin, Solana, and others) who installed malicious Firefox browser extensions
date	2025-08-08
loss	$1,000,000+
attack Type	Phishing via malicious browser extensions (credential and seed phrase exfiltration)
chain	Multi-chain (Ethereum, Bitcoin, Solana, and other EVM/non-EVM networks)
attribution	GreedyBear threat actor group — assessed as an evolution of the earlier 'Foxy Wallet' campaign that targeted Firefox crypto users

Executive Summary

The GreedyBear campaign deployed over 150 malicious Firefox browser extensions masquerading as legitimate crypto wallet interfaces — primarily MetaMask and Coinbase Wallet — to phish seed phrases, private keys, and login credentials from unsuspecting users. Victims who installed these extensions were presented with convincing wallet setup and import flows that captured sensitive material and exfiltrated it to attacker-controlled infrastructure, enabling systematic multi-chain account drainage totaling over $1 million. This campaign represents a direct evolution of the Foxy Wallet

operation, demonstrating increased sophistication in scale, evasion of Mozilla's Add-ons review process, and operational tempo. The incident underscores that browser extension marketplaces remain a critically underdefended attack surface in the crypto security stack.

What Happened

The GreedyBear campaign did not emerge from a vacuum. It is the operational successor to the Foxy Wallet campaign, an earlier effort that used a smaller set of malicious Firefox extensions to target crypto wallet users. The threat actors behind Foxy Wallet evidently learned from that campaign's eventual detection and takedown, refining their tactics for GreedyBear: broader extension coverage, more convincing UI clones, improved infrastructure rotation, and a significant increase in the number of extensions deployed simultaneously — over 150 across the Firefox Add-ons marketplace.

The attack chain was straightforward but effective. The threat actors created Firefox extensions with names, icons, and descriptions closely mimicking MetaMask, Coinbase Wallet, and other popular crypto wallet extensions. Some extensions used near-identical naming with subtle character substitutions or appended version numbers to evade basic string-matching filters. The extensions were listed on the official Mozilla Add-ons store, lending them a veneer of legitimacy that most users would not question. SEO poisoning and social media promotion — including posts on X (Twitter), Reddit, and Telegram crypto groups — drove traffic to these listings.

Once installed, the malicious extension presented the user with a polished wallet onboarding interface. Users were prompted to either create a new wallet or — critically — import an existing wallet by entering their seed phrase (BIP-39 mnemonic). Some variants also presented fake login screens for Coinbase Wallet, capturing email and password credentials. All captured data was immediately exfiltrated via HTTPS POST requests to attacker-controlled servers. Infrastructure analysis suggests the attackers used rotating domains and Cloudflare-fronted endpoints to complicate takedown efforts and C2 attribution.

With seed phrases in hand, the attackers executed automated sweeps of victim wallets across multiple chains. On-chain analysis indicates drainage scripts that enumerated token balances on Ethereum, BSC, Polygon, Arbitrum, Solana, and Bitcoin, then transferred all assets to consolidation wallets. The speed of drainage — often within minutes of seed phrase submission — indicates a high degree of automation. Consolidated funds were subsequently laundered through DEX swaps, cross-chain bridges, and mixing services, complicating recovery.

Mozilla removed the identified extensions following reports from security researchers, but the response lag was significant. Many of the 150+ extensions had been live for days to weeks before removal, and the threat actors demonstrated the ability to re-upload new variants faster than they were being taken down. This whack-a-mole dynamic is a systemic weakness across all browser extension marketplaces and is not unique to Firefox. The total confirmed losses exceeded $1 million, though the true figure

is likely higher given that many victims may not have reported their losses or attributed them to the extensions.

The GreedyBear campaign is a case study in how low-sophistication, high-volume phishing operations can achieve substantial financial impact in the crypto space. No smart contracts were exploited. No zero-days were burned. The attackers simply abused user trust in a distribution platform and the habitual willingness of crypto users to enter seed phrases into software they have not verified.

Kill Chain

1. **Staging & Distribution**
   Threat actors created 150+ malicious Firefox extensions impersonating MetaMask, Coinbase Wallet, and other crypto wallets. Extensions were uploaded to the official Mozilla Add-ons store with convincing names, icons, descriptions, and screenshots. Supplementary distribution via SEO poisoning, social media posts, and crypto forum promotion drove installation volume.

2. **Credential Harvesting**
   Upon installation, extensions presented pixel-perfect wallet onboarding UIs. Users were prompted to import existing wallets by entering their BIP-39 seed phrases or, for Coinbase Wallet clones, to enter email/password credentials. The UI was designed to mimic the legitimate extension experience with no visible anomalies.

3. **Exfiltration**
   Captured seed phrases and credentials were exfiltrated in real-time via HTTPS POST to attacker-controlled servers. Infrastructure used rotating domains and CDN fronting (Cloudflare) to resist takedown and obscure C2 endpoints. Data was transmitted immediately upon user submission — no local staging or delayed exfil.

4. **Multi-Chain Wallet Drainage**
   Automated scripts derived private keys from stolen seed phrases and swept all assets across Ethereum, BSC, Polygon, Arbitrum, Solana, Bitcoin, and other supported chains. Drainage typically occurred within minutes of seed phrase capture. Funds were consolidated into staging wallets.

5. **Laundering & Obfuscation**
   Stolen assets were laundered through DEX swaps, cross-chain bridges (likely including Thorchain and similar permissionless bridges), and mixing services. Rapid asset rotation and chain-hopping were used to frustrate on-chain tracing and asset recovery efforts.

Where Users Failed Themselves

— Installed wallet extensions without verifying the developer identity, extension ID, or official distribution link from the wallet provider's own website (e.g., metamask.io, coinbase.com).

— Entered seed phrases into a browser extension without confirming its authenticity — the single most catastrophic action a crypto user can take. Seed phrases should never be entered into any software that has not been rigorously verified.

— Relied on the Mozilla Add-ons store listing as a trust signal, assuming that presence on an official marketplace equates to security vetting. Browser extension stores do not perform deep security audits on submissions.

— Failed to use hardware wallets, which would have rendered stolen seed phrases insufficient for transaction signing in many configurations.

— Did not cross-reference the extension's download count, review history, or publication date — newly published extensions with zero reviews impersonating major wallets are an obvious red flag.

Prevention Checklist

FOR INDIVIDUAL USERS

☐ ALWAYS navigate to the official wallet provider's website (e.g., metamask.io) and use only the direct install link provided there. Never search for wallet extensions in the browser store directly.

☐ Use a hardware wallet (Ledger, Trezor, Keystone) as the primary signing mechanism. Even if a seed phrase is compromised, hardware wallet-protected accounts require physical device confirmation for transactions.

☐ Never enter a seed phrase into any browser extension, website, or application unless you are performing a deliberate wallet recovery on verified, air-gapped, or hardware-secured software.

☐ Verify the extension ID and developer name against the wallet provider's official documentation before installation. Bookmark the legitimate extension page for future reference.

☐ Enable browser-level extension installation restrictions: Firefox supports policies that restrict add-on installation to a pre-approved allowlist.

FOR PROTOCOLS & PROJECTS

☐ Wallet providers (MetaMask, Coinbase) should implement signed extension manifests and publish their official extension IDs prominently on their websites, in their documentation, and in their mobile apps.

☐ Wallet providers should actively monitor browser extension marketplaces for impersonators using automated trademark and code-similarity scanning, and maintain rapid takedown pipelines with Mozilla, Google, and other store operators.

☐ Implement canary mechanisms: wallet providers could embed unique identifiers in legitimate extensions that allow users to programmatically verify authenticity (e.g., a verification endpoint the extension calls on first launch).

FOR THE ECOSYSTEM

☐ Mozilla must dramatically improve its Add-ons review pipeline for extensions requesting sensitive permissions (access to all website data, clipboard access, etc.), particularly those impersonating known financial or crypto brands.

☐ Browser vendors should implement mandatory developer identity verification (KYC-equivalent) for extensions in financial/crypto categories and enforce minimum publication-age or review thresholds before extensions appear in search results.

☐ The broader crypto security community should maintain and distribute public blocklists of known malicious extension IDs and C2 domains, integrated into wallet software and browser security extensions (e.g., Blockaid, Pocket Universe, Wallet Guard).

Key Takeaway

Your seed phrase is your entire portfolio. Any software asking for it that you have not verified through the wallet provider's official website is an adversary. Browser extension stores are distribution platforms, not security guarantees. The GreedyBear campaign stole $1M with nothing more than convincing UI clones and users who trusted the wrong install button.

S O U R C E S

— ZeroTraceLabs internal threat intelligence analysis — GreedyBear campaign tracking (August 2025)

— Mozilla Add-ons abuse reports and extension takedown records (August 2025)

— On-chain fund flow analysis of GreedyBear consolidation wallets across Ethereum, BSC, Solana, and Bitcoin

— Prior reporting on the Foxy Wallet campaign — predecessor operation targeting Firefox crypto extension users

— Community incident reports aggregated from crypto security forums, Reddit r/CryptoCurrency, and X (Twitter) disclosures