Apple Crimeware | Massive Rust Infostealer

Among those, we discovered some samples are already targeting Apple's forthcoming OS release, macOS 14 Sonoma. Earlier this month, security researcher iamdeadlyz detailed an interesting campaign that uses Rust to craft a new malware, which we refer to as Realst. Building on this previous analysis, we identified the variant families A, B, and C, which we detail below.

### Realst Variant Family A

Realst Variant Family A appears to be the least sophisticated of the three variants, as they are more of a proof of concept. Variant A contains easily identifiable strings related to password scraping and it uses several network URLs to exfiltrate stolen data. In many versions, the malware also uses the scraped password to elevate privileges with the `sudo` command or the `osascript` command.

### Realst Variant Family B

Variant B uses similar artifacts to Variant A, but it breaks up the strings related to password scraping and elevating privileges. In addition to the network URLs, Variant B also includes a command to display a dialog box with the “hidden answer” option. This prevents the user from seeing the characters they type by replacing them with bullet points, similar to a real password safe.

### Realst Variant Family C

Variant C contains more complex string artifacts related to beta Apple software, such as `hw.model` and `osascript`. In some versions, the malware also uses the scraped password to elevate privileges with the `sudo` command.

### Realst Infostealer Prepares for macOS 14 Sonoma

In Family D, which accounted for 16/59 samples, there are no static artifacts for versioning. Dependencies, from a telemetry point of view the key to all these infostealers is the access and exfiltration of browser data, crypto wallets, and keychains.

The samples we analyzed reach out to one of two hardcoded URLs to exfiltrate stolen data. The first URL, `http://77[.].91[.].84[.].110:5000/send_analytics`, tries to exfiltrate all relevant system data. The second URL, `http://77[.].91[.].84[.].110:8010/hash`, tries to exfiltrate a hash of the data.

Other versions of realst stealer are distributed as applications via `/~/.config/realst`, such as a `backstory` file and `super.keychain`. We explore these differences in detail in the static analysis section below.

### Realst Infostealer In Apple's New macOS 14 Sonoma

As reported by iamdeadlyz, the campaign appears to have links to the earlier macOS stealers and may have been designed to take advantage of earlier macOS vulnerabilities. The Realst malware contains string artifacts related to beta Apple software, such as `AppleScript spoong` and `AppleScript tun`.

### Realst Distribution

Realst uses `AppleScript spoong` and `AppleScript tun` to access keychain data and store the data in a .pkg file. The `install` script is simply a barebones uninstall script with no malicious behavior. The `uninstall` script attempts to delete the `.pkg` installer containing a malicious Mach-O and three related scripts.

### SentinelOne Protects Against Realst Infostealer

It is not clear at this point how differences between Sonoma and Ventura would affect execution of the malware – a question it seems the malware developers were also asking themselves. Observing MITRE TTPs, we can see that the threat actor has invested serious effort in order to target macOS users for data and keychain.

The number of Realst samples and their variation shows that the threat actor has invested serious effort in order to target macOS users for data and keychain. In the case of macOS, the infostealer turned out to be a new malware written in Rust, dubbed “realst”. Building on this previous analysis, we identify some unique techniques used by Realst in its effort to exfiltrate data from infected systems.