Threat Actor Selling New Atomic macOS (AMOS) Stealer on Telegram

blog.cyble.com/2023/04/26/threat-actor-selling-new-atomic-macos-amos-stealer-on-telegram/

April 26, 2023

Undetected Golang-Based Stealer Emerges and Baffles Security Vendors

In recent years, macOS has become increasingly popular among users, largely due to its user-friendly interface, which is often commended for its simplicity and ease of use.

macOS is also often perceived as being more secure than other operating systems. Despite this, Threat Actors (TAs) have continued to target macOS platforms. Previously, there have been several cases where Threat Actors have targeted macOS users with various families of malware, including MacStealer, RustBucket, DazzleSpy, etc.

Cyble Research and Intelligence Labs (CRIL) recently discovered a Telegram channel advertising a new information-stealing malware called **Atomic macOS Stealer (AMOS)**. The malware is specifically designed to target macOS and can steal sensitive information from the victim's machine.

The TA behind this stealer is constantly improving this malware and adding new capabilities to make it more effective. The most recent update to the malware was highlighted in the Telegram post on April 25th, showcasing its latest features.

The Atomic macOS Stealer can steal various types of information from the victim's machine, including keychain passwords, complete system information, files from the desktop and documents folder, and even the macOS password. The stealer is designed to target multiple browsers and can extract auto-fills, passwords, cookies, wallets, and credit card information. Specifically, AMOS can target cryptowallets such as Electrum, Binance, Exodus, Atomic, and Coinomi.

The TA also provides additional services such as a web panel for managing victims, meta mask brute-forcing for stealing seed and private keys, crypto checker, and dmg installer, after which it shares the logs via Telegram. These services are offered at a price of $1000 per month.

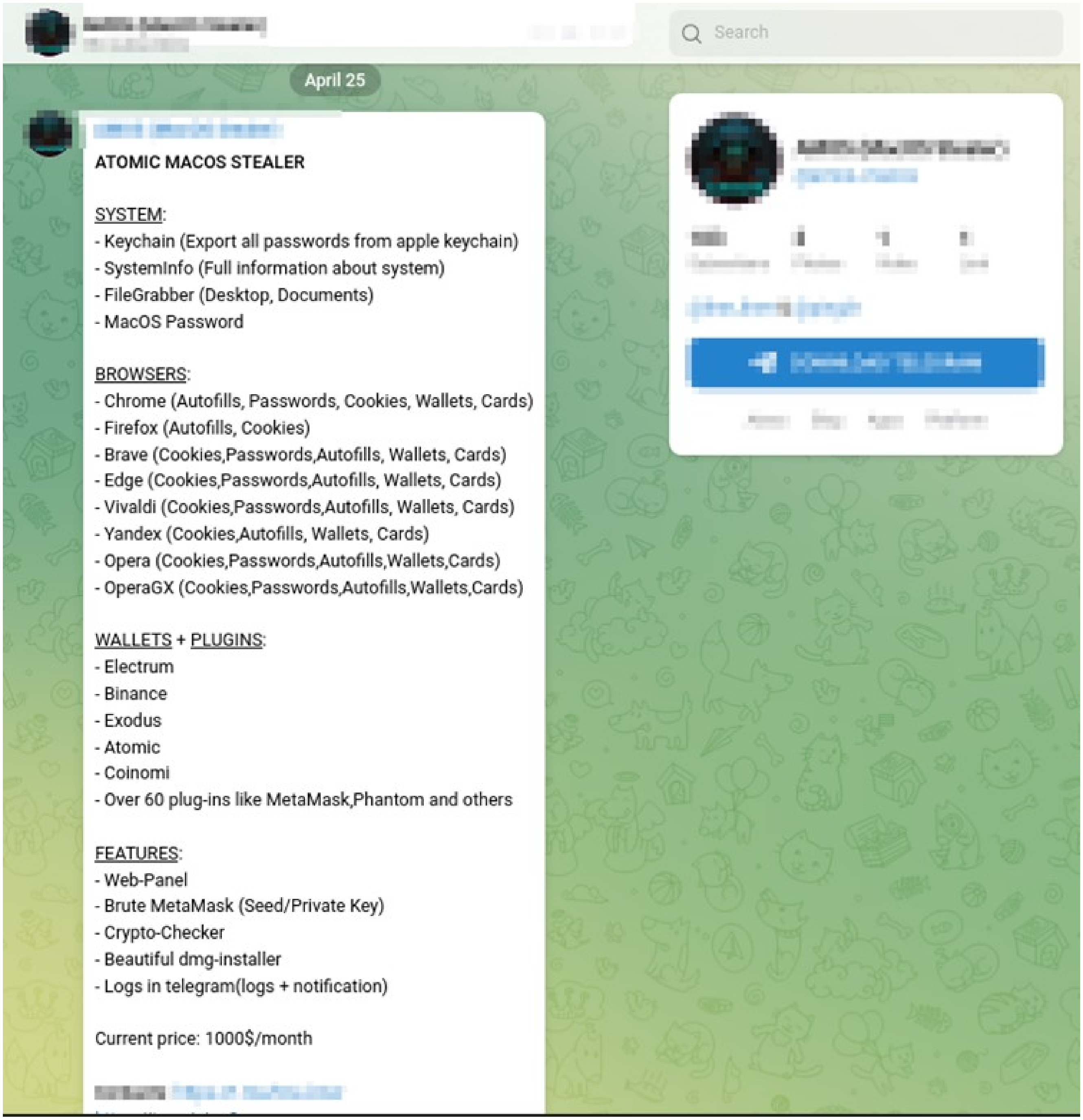

Figure 1 – Telegram Post by Malware Developer

Technical Analysis

For our analysis, we have taken the sample hash (SHA256) of “Setup.dmg” as *15f39e53a2b4fa01f2c39ad29c7fe4c2fef6f24eff6fa46b8e77add58e7ac709*, which is FUD (stands for “Fully Undetectable”) on Virustotal at the time of writing this analysis.

The TAs use a ‘.dmg’ file to disseminate this malware, including a Mac OS X executable, located at “/Setup.app/Contents/macOS/My Go Application.app” and is a 64-bit Golang executable file.

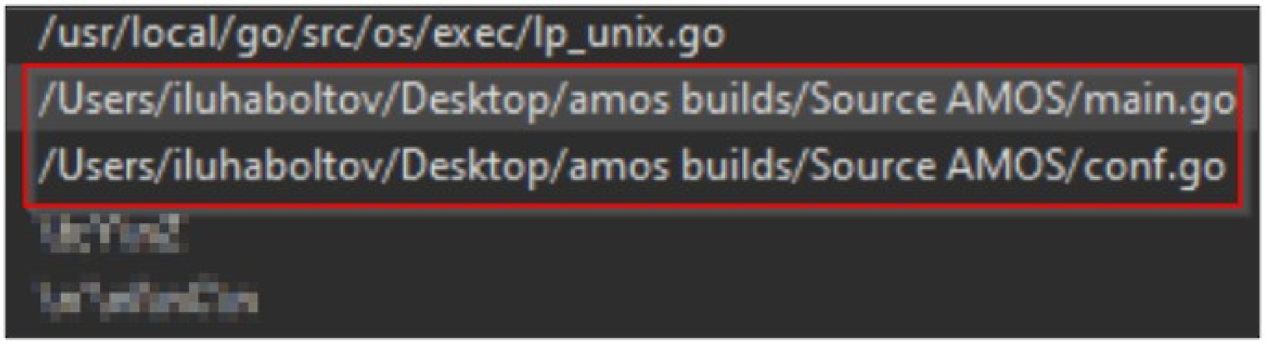

Figure 2 – Strings related to Go Source Files of Stealer

The Atomic macOS Stealer’s primary function encompasses all of its capabilities, including keychain extraction, crypto wallet theft, stealing browser details, grabbing user files, collecting system information, and sending all the stolen data to the remote C&C server.

The main functions of the stealer are depicted in the figure below.

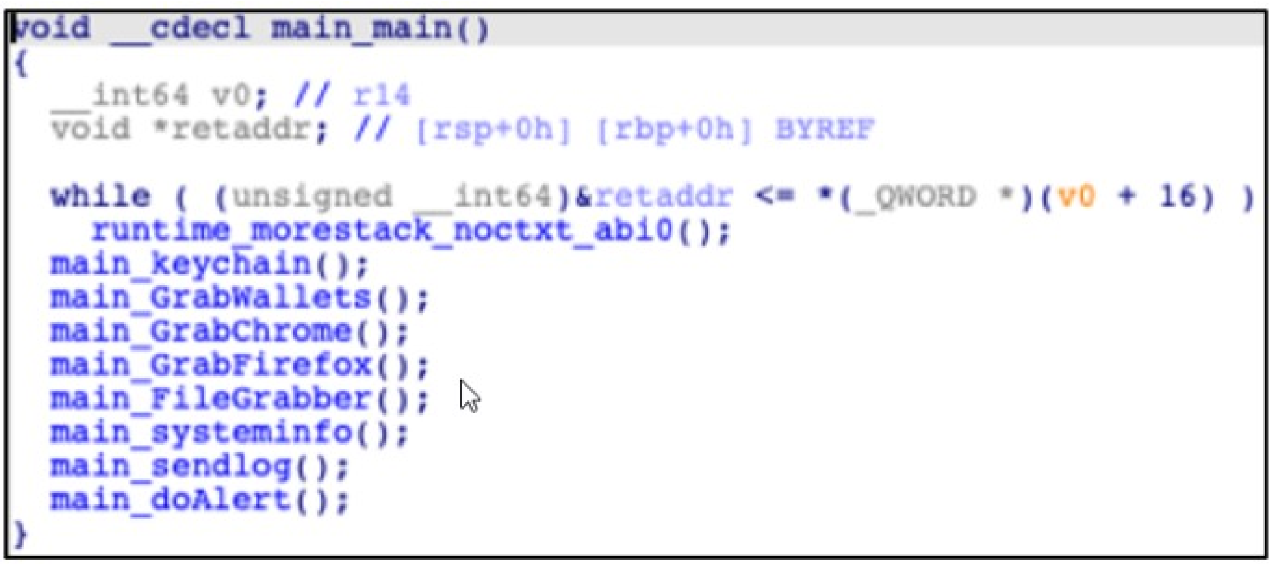

Figure 3 – Stealer’s main function

Once a user executes the file, it displays a fake password prompt to obtain the system password, as shown in the figure below.

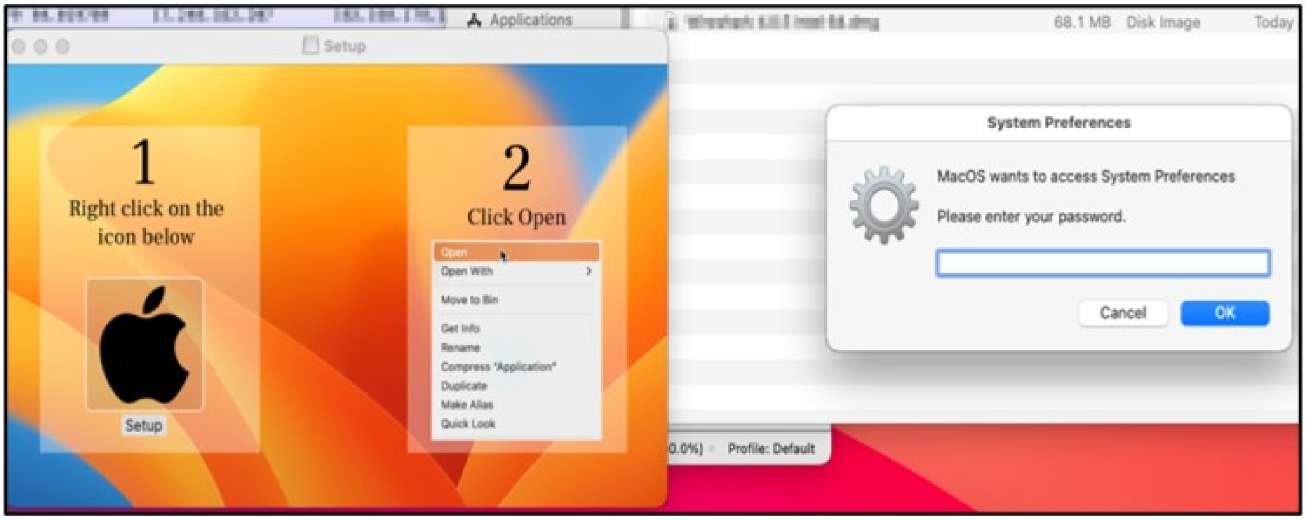

Figure 4 – Fake password prompt

Keychain Password Extraction

In addition to obtaining the system password, the malware also targets the password management tool by utilizing the *main_keychain()* function to extract sensitive information from the victim's machine. Keychain is a macOS password management system that enables users to safely store sensitive data such as website logins, Wi-Fi passwords, credit card details, and more.

The code snippet depicted in the figure below exhibits the *main_keychain()* function, implemented to gather the user's credentials.

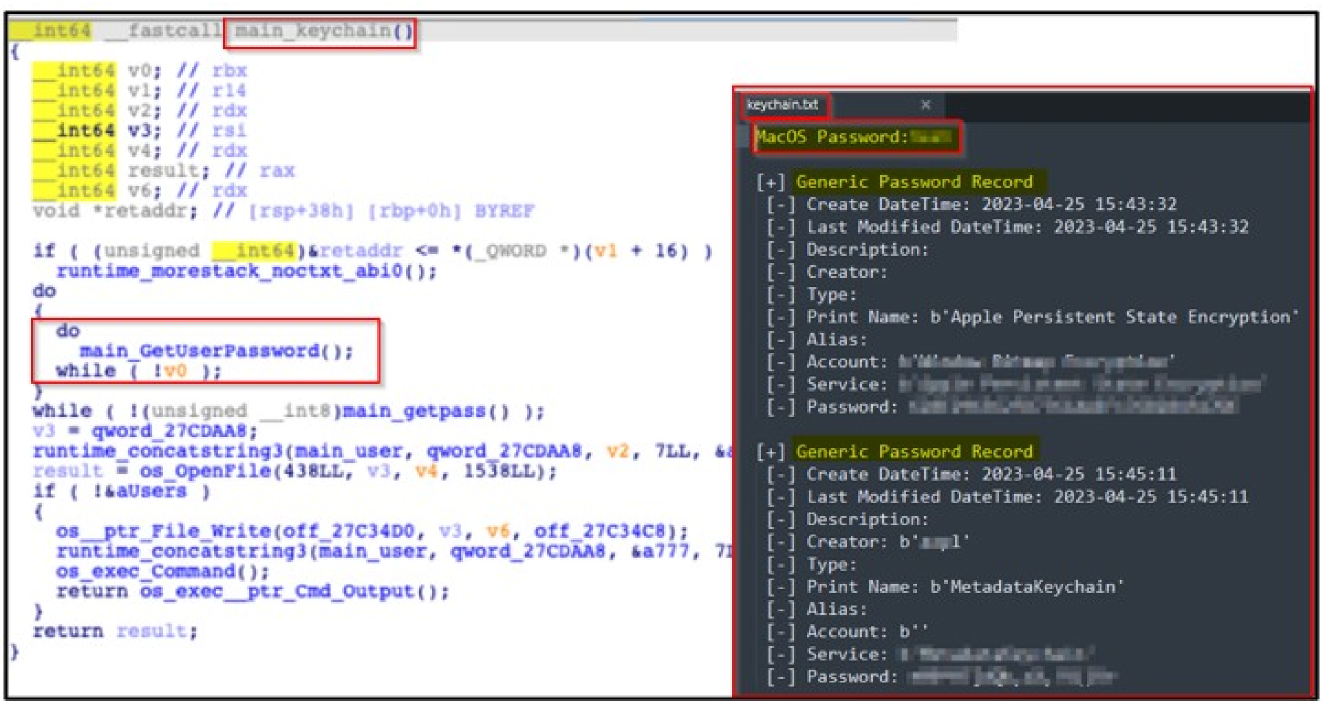

Figure 5 – Keychain password extraction

Stealing Crypto Wallets

After that, the stealer begins to extract information related to crypto-wallets by querying and reading files from specific directories using the function *main_GrabWallets().* The stealer targets crypto wallets such as Electrum, Binance, Exodus, and Atomic, as shown below.

```
if ( runtime_writeBarrier )
  runtime_gcWriteBarrier(&main_telegram);
else
  main_telegram = v43;
v45 = runtime_concatstring3(main_user, qword_27CDAA8, v44, 7LL, "/.electrum/wallets/", 19LL);
qword_27CEE48 = (__int64)"/Users/";
if ( runtime_writeBarrier )
  runtime_gcWriteBarrier(&qword_27CEE40);
else
  qword_27CEE40 = v45;
v47 = main_library;
v48 = runtime_concatstring2("Coinomi/wallets/", 16LL, v46, qword_27CDA68);
qword_27CEE58 = v47;
if ( runtime_writeBarrier )
  runtime_gcWriteBarrier(&qword_27CEE50);
else
  qword_27CEE50 = v48;
v50 = main_library;
v51 = runtime_concatstring2("Exodus/", 7LL, v49, qword_27CDA68);
qword_27CEE68 = v50;
if ( runtime_writeBarrier )
  runtime_gcWriteBarrier(&qword_27CEE60);
else
  qword_27CEE60 = v51;
v53 = main_library;
result = runtime_concatstring2("atomic/Local Storage/leveldb/", 29LL, v52, qword_27CDA68);
qword_27CEE78 = v53;
if ( runtime_writeBarrier )
  return runtime_gcWriteBarrier(&qword_27CEE70);
qword_27CEE70 = result;
return result;
```

Figure 6 – Targeted Crypto-wallets

Crypto Wallet Extension

The Atomic macOS stealer can also extract information from crypto wallet browser extensions. These extensions are integrated into the stealer binary via hard coding, with over 50 extensions being targeted thus far.

The table below highlights some crypto wallets with respective browser extension IDs targeted by the malware.

acmacodkjbdgmoleebolmdjonilkdbch	Rabby Wallet
aeachknmefphepccionboohckonoeemg	Coin98 Wallet
afbcbjpbpfadlkmhmclhkeeodmamcflc	Math Wallet
aholpfdialjgjfhomihkjbmgjidlcdno	Exodus Web3 Wallet
aiifbnbfobpmeekipheeijimdpnlpgpp	Station Wallet
amkmjjmmflddogmhpjloimipbofnfjih	Wombat – Gaming Wallet for Ethereum & EOS
apnehcjmnengpnmccpaibjmhhoadaico	CWallet
bcopgchhojmggmffilplmbdicgaihlkp	Hycon Lite Client

bfnaelmomeimhlpmgjnjophhpkkoljpa	Phantom
bocpokimicclpaiekenaeelehdjllofo	XDCPay
cgeeodpfagjceefieflmdfphplkenlfk	EVER Wallet
cihmoadaighcejopammfbmddcmdekcje	LeafWallet
cjelfplplebdjjenllpjcblmjkfcffne	Jaxx Liberty
cjmkndjhnagcfbpiemnkdpomccnjblmj	Finnie
cmndjbecilbocjfkibfbifhngkdmjgog	Swash
cnmamaachppnkjgnildpdmkaakejnhae	Auro
copjnifcecdedocejpaapepagaodgpbh	Freaks Axie
cphhlgmgameodnhkjdmkpanlelnlohao	NeoLine
dhgnlgphgchebgoemcjekedjjbifijid	Crypto Airdrops & Bounties
dkdedlpgdmmkkfjabffeganieamfklkm	Cyano
dmkamcknogkgcdfhhbddcghachkejeap	Keplr
efbglgofoippbgcjepnhiblaibcnclgk	Martian Wallet for Sui & Aptos
egjidjbpglichdcondbcbdnbeeppgdph	Trust Wallet
ffnbelfdoeiohenkjibnmadjiehjhajb	Yoroi
fhbohimaelbohpjbbldcngcnapndodjp	BinanceChain
fhilaheimglignddkjgofkcbgekhenbh	Oxygen
flpiciilemghbmfalicajoolhkkenfel	ICONex
fnjhmkhhmkbjkkabndcnnogagogbneec	Ronin
fnnegphlobjdpkhecapkijjdkgcjhkib	Harmony Wallet
hcflpincpppdclinealmandijcmnkbgn	KHC
hmeobnfnfcmdkdcmlblgagmfpfboieaf	XDEFI
hnfanknocfeofbddgcijnmhnfnkdnaad	Coinbase
hnhobjmcibchnmglfbldbfabcgaknlkj	Flint Wallet
hpglfhgfnhbgpjdenjgmdgoeiappafln	Guarda
ibnejdfjmmkpcnlpebklmnkoeoihofec	TronLink
imloifkgjagghnncjkhggdhalmcnfklk	Trezor Password Manager
jojhfeoedkpkglbfimdfabpdfjaoolaf	Polymesh

klnaejjgbibmhlephnhpmaofohgkpgkd	ZilPay
kncchdigobghenbbaddojjnnaogfppfj	iWallet
kpfopkelmapcoipemfendmdcghnegimn	Liquality
lodccjjbdhfakaekdiahmedfbieldgik	DAppPlay
mfhbebgoclkghebffdldpobeajmbecfk	Starcoin
mnfifefkajgofkcjkemidiaecocnkjeh	TezBox
nhnkbkgjikgcigadomkphalanndcapjk	CLW
nkbihfbeogaeaoehlefnkodbefgpgknn	Metamask
nknhiehlklippafakaeklbeglecifhad	Nabox
nlbmnnijcnlegkjjpcfjclmcfggfefdm	MewCx
nlgbhdfgdhgbiamfdfmbikcdghidoadd	Byone
nphplpgoakhhjchkkhmiggakijnkhfnd	Ton
ookjlbkiijinhpmnjffcofjonbfbgaoc	Temple
pdadjkfkgcafgbceimcpbkalnfnepbnk	KardiaChain
pnndplcbkakcplkjnolgbkdgjikjednm	Tron Wallet & Explorer – Tronium
pocmplpaccanhmnllbbkpgfliimjljgo	Slope
ppdadbejkmjnefldpcdjhnkpbjkikoip	Oasis

Extracting Browser Information

After collecting wallet details, the malware queries the installed browsers' directories on the victim's device and searches for particular browser-related files to extract confidential data, such as:

- Autofills
- Passwords
- Cookies
- Credit Cards

As depicted below, the malware can steal files from various browsers, including Mozilla Firefox, Google Chrome, Microsoft Edge, Yandex, Opera, and Vivaldi.

```
v20 = runtime_concatstring2("Firefox/Profiles/", 17LL, v18, qword_27CDA68);
qword_27CDA58 = v19;
if ( runtime_writeBarrier )
  runtime_gcWriteBarrier(&main_firefox);
else
  main_firefox = v20;
v22 = main_library;
v23 = runtime_concatstring2("Google/Chrome/", 14LL, v21, qword_27CDA68);
qword_27CDA18 = v22;
if ( runtime_writeBarrier )
  runtime_gcWriteBarrier(&main_chrome);
else
  main_chrome = v23;
v25 = main_library;
v26 = runtime_concatstring2("BraveSoftware/Brave-Browser/", 28LL, v24, qword_27CDA68);
qword_27CDA08 = v25;
if ( runtime_writeBarrier )
  runtime_gcWriteBarrier(&main_brave);
else
  main_brave = v26;
v28 = main_library;
v29 = runtime_concatstring2("Microsoft Edge/", 15LL, v27, qword_27CDA68);
qword_27CDA48 = v28;
if ( runtime_writeBarrier )
  runtime_gcWriteBarrier(&main_edge);
else
  main_edge = v29;
v31 = main_library;
v32 = runtime_concatstring2("Vivaldi/", 8LL, v30, qword_27CDA68);
qword_27CDAB8 = v31;
if ( runtime_writeBarrier )
  runtime_gcWriteBarrier(&main_vivaldi);
else
  main_vivaldi = v32;
v34 = main_library;
v35 = runtime_concatstring2("Yandex/YandexBrowser/", 21LL, v33, qword_27CDA68);
qword_27CDAC8 = v34;
if ( runtime_writeBarrier )
  runtime_gcWriteBarrier(&main_yandex);
else
  main_yandex = v35;
v37 = main_library;
v38 = runtime_concatstring2("com.operasoftware.Opera/", 24LL, v36, qword_27CDA68);
qword_27CDA78 = v37;
if ( runtime_writeBarrier )
  runtime_gcWriteBarrier(&main_opera);
else
  main_opera = v38;
v40 = main_library;
v41 = runtime_concatstring2("com.operasoftware.OperaGX/", 26LL, v39, qword_27CDA68);
qword_27CDA88 = v40;
```

Figure 7 – Targeted web browsers

File Grabber

The stealer now steals the victim's files from directories such as *Desktop* and *Documents* using the *main_FileGrabber()* function. The figure below shows the malware requesting permission to access files within the specified directories.

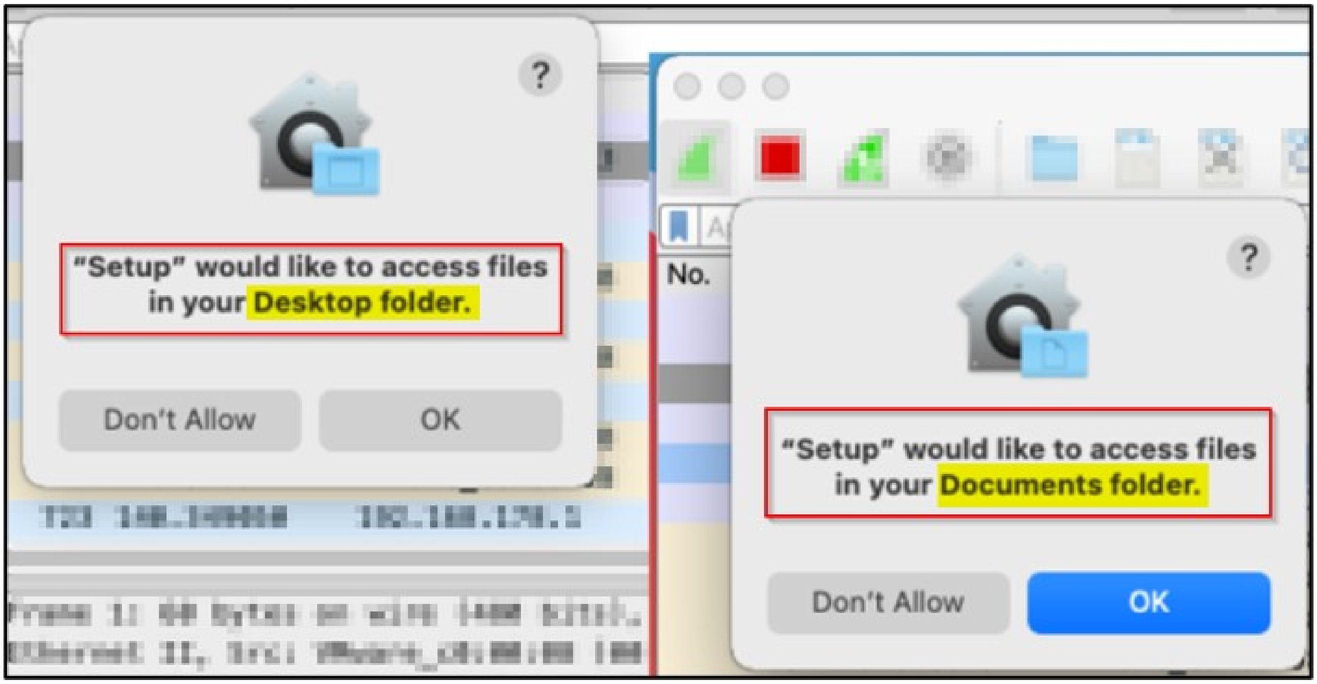

Figure 8 – Stealer requesting permission to access files

The code snippet in the figure below displays the *main_FileGrabber()* function, which is implemented to grab files from the victim's system.

```
if ( runtime_writeBarrier )
  runtime_gcWriteBarrier(&main_library);
else
  main_library = v13;
v15 = runtime_concatstring3(main_user, qword_27CDAA8, v14, 7LL, "/Desktop/", 9LL);
qword_27CDA28 = (__int64)"/Users/";
if ( runtime_writeBarrier )
  runtime_gcWriteBarrier(&main_desktop);
else
  main_desktop = v15;
v17 = runtime_concatstring3(main_user, qword_27CDAA8, v16, 7LL, "/Documents/", 11LL);
qword_27CDA38 = (__int64)"/Users/";
if ( runtime_writeBarrier )
  runtime_gcWriteBarrier(&main_documents);
```

Figure 9 – File grabber

Collecting System Information

Subsequently, the malware starts the process of obtaining further hardware-related information regarding the system, such as the Model name, Hardware UUID, RAM size, the number of cores, and serial number, among other information. This is illustrated in the figure below.

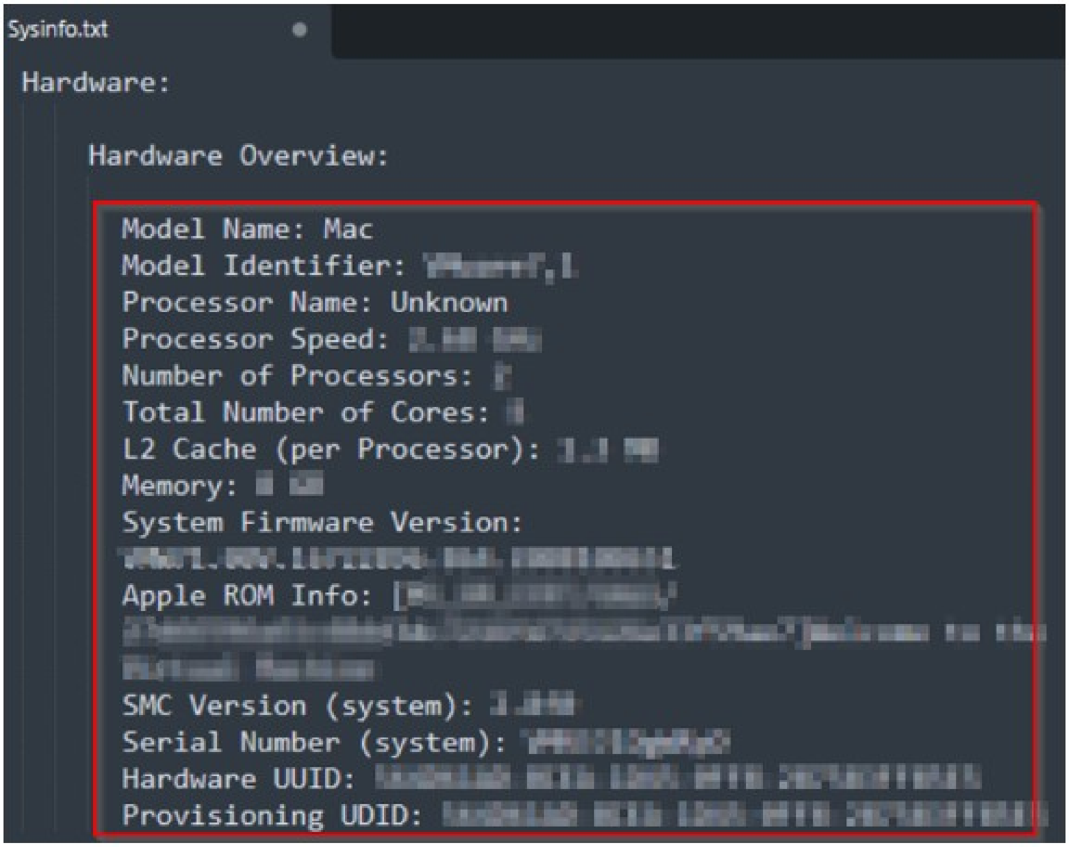

Figure 10 – Collected system information

Command and Control (C&C)

Finally, the Atomic macOS stealer processes the stolen information by compressing into ZIP and encoding it using Base64 format for exfiltration.

The stealer communicates with the below C&C server URL and sends the stolen information.

> *hxxp[:]//amos-malware[.]ru/sendlog*

The figure below shows the network communication of data exfiltration from the victim’s machine.

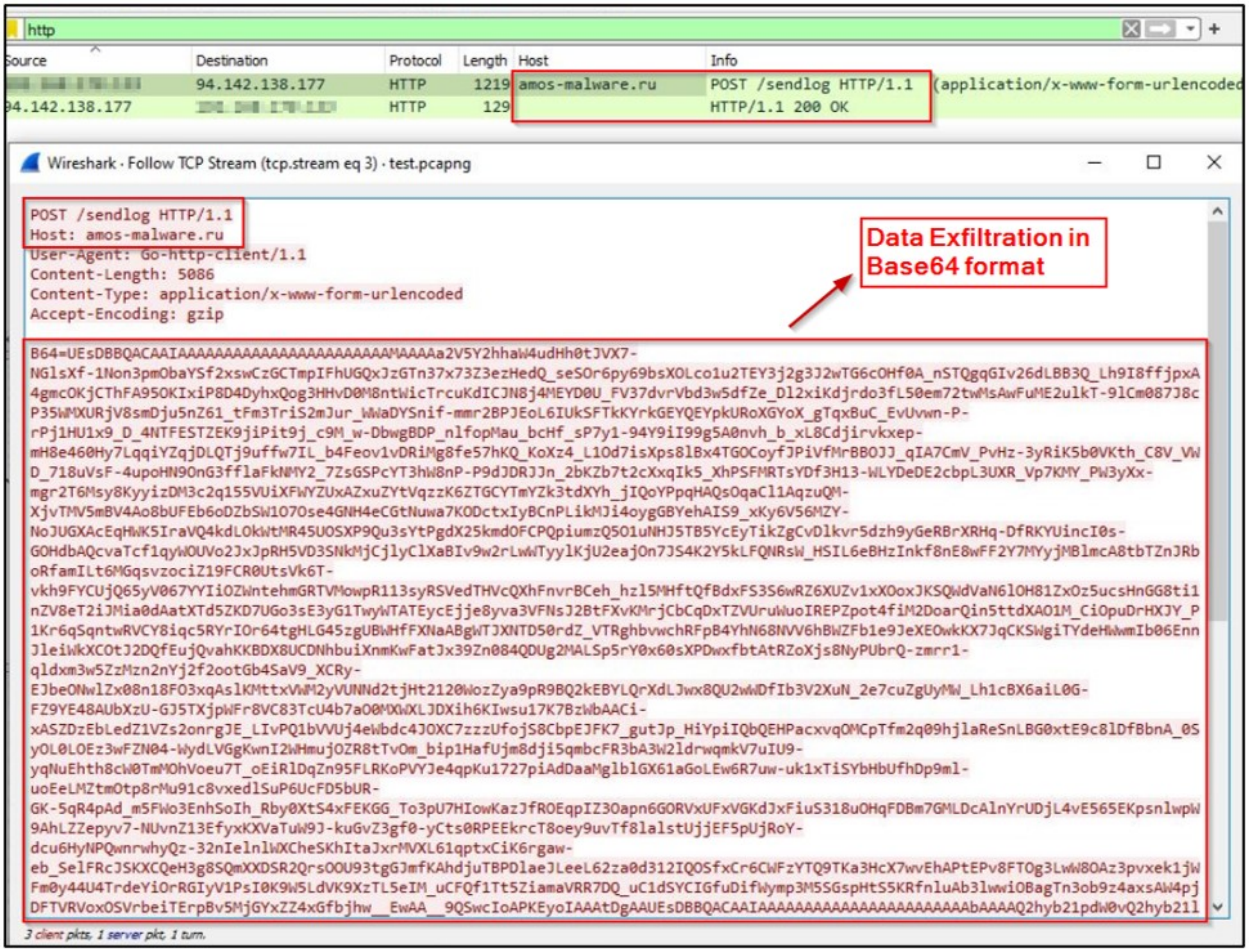

Figure 11 – Exfiltrated data

Concurrently, the Atomic macOS stealer sends selected information to Telegram channels along with the compiled ZIP file, as shown below.

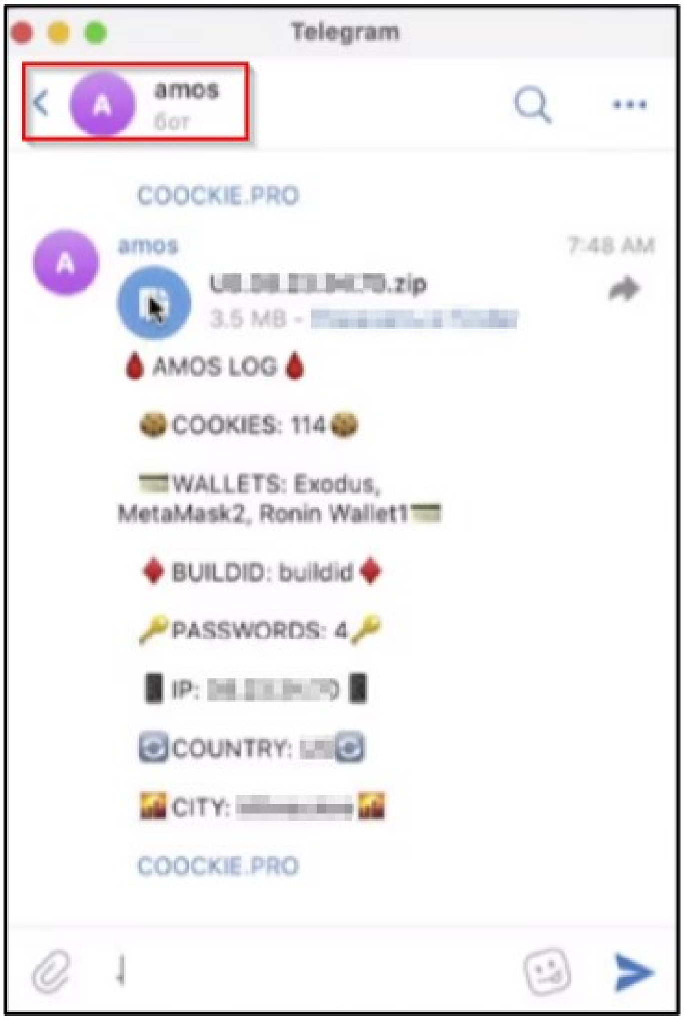

Figure 12 – Sending ZIP file to Telegram channel

C&C Panel

The below figure shows Atomic macOS stealer’s active C&C panel.

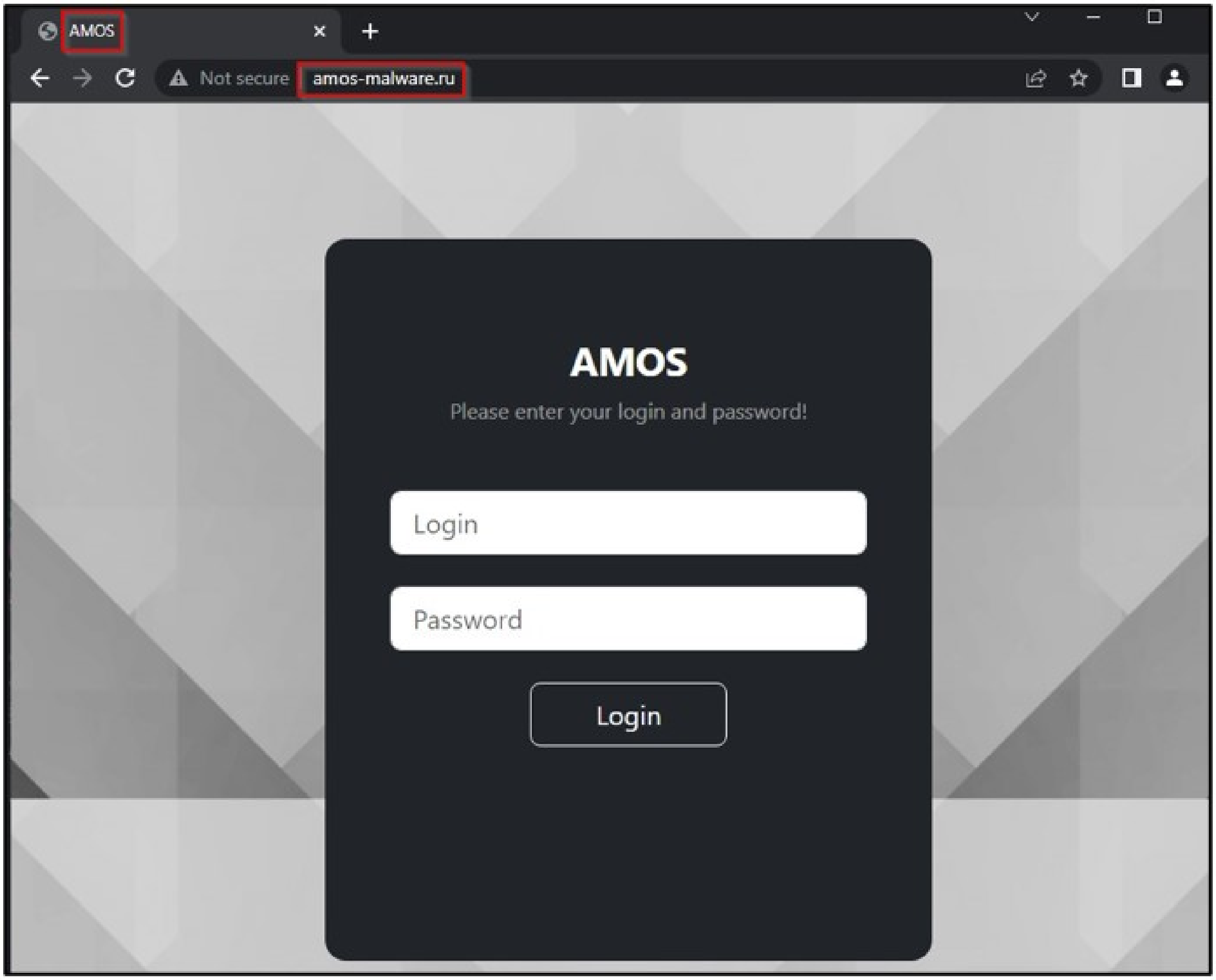

Figure 13 – AMOS C&C panel

Conclusion

Due to its robust security features, macOS is the preferred operating system for numerous high-profile individuals. Targeting macOS is not a novel trend, and various malware families exist that specifically aim to infiltrate this operating system.

Malware such as the Atomic macOS Stealer could be installed by exploiting vulnerabilities or hosting on phishing websites. Threat Actors can use the stolen data for espionage or financial gain. While not commonplace, macOS malwares can have devastating impacts on victims.

Our Recommendations

We have listed some essential cybersecurity best practices that create the first line of control against attackers. We recommend that our readers follow the best practices given below:

- Download and install software only from the official Apple App Store.
- Use a reputed antivirus and internet security software package on your system.
- Use strong passwords and enforce multi-factor authentication wherever possible.

- Enable biometric security features such as fingerprint or facial recognition for unlocking the device wherever possible.
- Be wary of opening any links received via emails delivered to you.
- Be careful while enabling any permissions.
- Keep your devices, operating systems, and applications updated.

MITRE ATT&CK® Techniques

Tactic	Technique ID	Technique Name
Execution	T1204.002	User Execution: Malicious File
Credential Access	T1110	Brute Force
Credential Access	T1555.001	Keychain
Credential Access	T1555.003	Credentials from Web Browsers
Discovery	T1083	File and Directory Discovery
Command and Control	T1132.001	Data Encoding: Standard Encoding
Exfiltration	T1041	Exfiltration Over C&C Channel

Indicators of Compromise (IoC)

Indicators	Indicators Type	Description
5e0226adbe5d85852a6d0b1ce90b2308 0a87b12b2d12526c8ba287f0fb0b2f7b7e23ab4a 15f39e53a2b4fa01f2c39ad29c7fe4c2fef6f24eff6fa46b8e77add58e7ac709	MD5 SHA1 SHA256	Setup.dmg
amos-malware[.]ru	Domain	C&C
hxxp[:]//amos-malware[.]ru/sendlog	URL	C&C