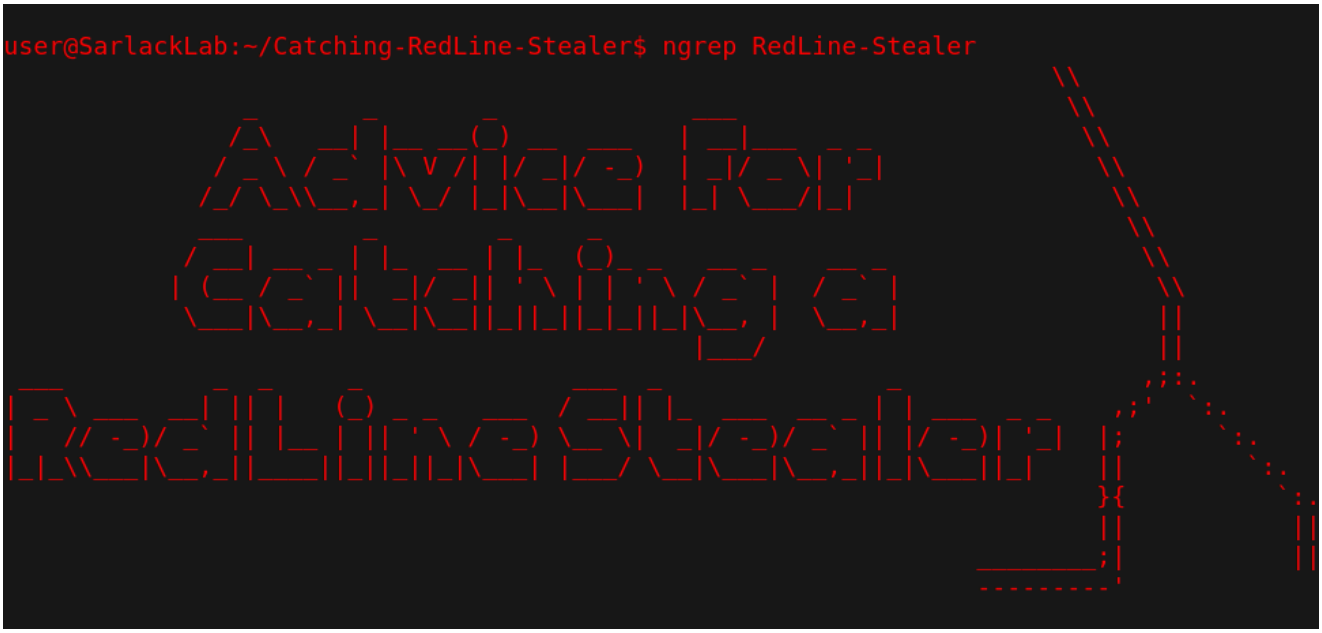
Advice For Catching a RedLine Stealer

medium.com/@the_abjuri5t/advice-for-catching-a-redline-stealer-dca126867193

John F October 23, 2023



John F

--

RedLine Stealer is an infamous malware strain that provides cyber-criminals with a reliable payload for stealing sensitive information from an infected computer. Both MalwareBazaar statistics and ANY.RUN trends consistently track RedLine as the most common payload on their platforms. Redline Stealer is classified by malware taxonomy as an “information stealer” (infostealer). Like many infostealers, RedLine is leveraged by cyber-criminals for reconnaissance and initial access. The information that RedLine steals includes:

- browser autofill details (login credentials, payment card information, contact information, etc.)
- session cookies
- cryptocurrency keys
- certain files (dependent on configuration)
- and system details of the infected computer

RedLine Stealer is maintained by professional malware developers who sell access to the infostealer on Malware-as-a-Service (MaaS) markets. The purchasing 'clients' specialize in infecting victims with RedLine and profiting from that stolen information. RedLine has grown in popularity among cyber-criminals — becoming responsible for a large number of account compromises and resulting damages. A well-known account compromise, attributed to a RedLine Stealer infection, was the 2023 "hack" of *Linus Tech Tips* — a technology communications YouTube channel with 15+ Million subscribers. In the modern (October 2023) threat landscape, many network defenders consider RedLine Stealer infamous due to the malware's prevalence and stealing capabilities.

Table of Contents

War Story — Late Night Data Exfil… over cleartext?

> It was around 11 p.m. and I felt worn-out as I was nearing the end of my night shift. After I finished addressing a few minor alerts, I figured I'd reward myself by running a brief series of threat hunting queries for malware (yes, this is what SOC analysts do for fun). Sifting through the results of communications to suspicious servers, I noticed that one of the connections had an interesting pattern… the timings and amounts of data[1] almost looked like some amount of data exfiltration to this already untrustworthy IP address.
>
> Less-groggy (and admittedly excited about a potential incident), I downloaded a full PCAP of the connection and ran some searches on the external IP address. Viewing the TCP stream in Wireshark, I was thrilled to see that the protocol was in cleartext. "Oh wicked!" I initially thought as I could now easily read the network traffic with my own eyes. "Wait... are those file locations… and usernames… and oh $#|7 it's got passwords too!"
>
> I immediately alerted the client and we began taking steps toward remediation (scrambling accounts, isolating the host, blocking the C2's IP, etc). We were able to prioritize accounts to be locked-down, as the cleartext in the PCAP conveniently acted as a log of compromised credentials. My threat hunting and searches for suspicious servers that night was essential for detecting the initial stages of this attack so that we could respond long before it became a larger incident.
>
> As part of our follow-up investigation, I checked the reputation of the IP to see what community threat intelligence said: RedLine. "Wait. As in that infamous RedLine Stealer?" I thought. "Sure, this is definitely some infostealer — but **RedLine** has got to be more advanced than some cleartext data exfil... right?"

Data Exfiltration Protocol

During its detonation, a RedLine Stealer payload will initiate communications with a hard-coded C2 (command-and-control) server. RedLine's C2 channels leverage direct TCP sockets[2] and communicate over a custom cleartext protocol. After a payload authenticates with its C2 and receives an updated configuration (if applicable) it will gather sensitive data from the infected computer and exfiltrate the stolen information to its C2. The custom C2 protocol RedLine implements has every individual communication follow a strict format of:

1. initiating the connection
2. stating the call's classification
3. providing a 32-character authorization key
4. and sending information

All communications over RedLine's protocol originate with a 'call' from the infected computer, which is followed by a corresponding 'response' from the C2 server. The protocol classifies these call-and-response pairs with specific 'Id numbers' that range from "Id1" to Id24". Most of the Id numbers classify the sending of data — though some are used to request information from the C2 server (and a few… well I confess, I have not been able to confirm what they are used for — that's a point of future research).

"Id6" classifies stolen information regarding installed anti-virus/EDR (referred to as "defenders" by RedLine)
The string "http://tempuri.org/" appears just prior to an entity's classification. tempuri.org is **not** an IOC but is actually a leftover artifact of improperly implemented ASP.Net web services. We can leverage a combination of such leftover strings — along with control characters, Id classifications, and the format of stolen data — to write Snort/Suricata signatures that will detect RedLine Stealer data exfiltration.

Formatted Suricata rules in and at on GitHub.

Command-and-Control Meta-Analysis

```
Data for meta-analysis of C2 infrastructure sourced from: - abuse.ch's threat
intelligence community platform  - C2 servers uncovered by
```

Domain Name Infrastructure:

Many of the RedLine Stealer payloads I've investigated leverage DNS resolution of hard-coded domain names in order to identify and connect to their C2 servers. Approximately 2/3rds of the domain names are registered under the top-level domains ***.xyz** and ***.top**. There are no significant patterns of the RedLine Stealer second-level domains under .xyz and .top. The subdomains are mainly just high-entropy domain names (pseudo-random characters or words meshed together in a non-nonsensical manner). Similar to the C2

patterns of many commodity RATs, the cyber-criminals who deploy RedLine Stealer also leverage Dynamic DNS (DDNS) services[4] such as ***.duckdns[.]org** and ***.ddns[.]org** (see NanoCore Hunting Guide).

showing patterns among RedLine Stealer C2 domains.

```
The second-level domains makelogs[.]org and tuktuk[.]ug appear to be part ofsome
temporary campaign and/or leveraged hosting infrastructure. As noted in a, a threat
actor had been registering similarsubdomains under tuktuk[.]ug every day in early
September. You can investigatemany curious patterns among C2 infrastructure by
checking-out the .
```

IP Address Hosting:

Based on the payloads I've investigated, however, approximately 4/5ths of RedLine payloads **do not** leverage DNS to connect to their C2 server. Instead of querying their C2's domain, the majority of payloads just have hard-coded IP addresses[4] which they connected to directly. Many of the IP addresses used by RedLine C2 servers are hosted on a variety of amorphous bulletproof hosting providers and/or compromised infrastructure. Patterns among the hosting infrastructure and ASN ranges for these IPs are rather sparse. Overall, the leveraged IPv4 infrastructure appears relatively similar to other commodity malware[5].

The majority of RedLine Stealer payloads have the IP address of their C2 server hard-coded. The IP itself is stored inside Base64 encoding after being XORed with a hard-coded key.

Catching the RedLine

Cybercriminals commonly leverage infostealers for reconnaissance and initial access stages of their attacks — collecting target information and stealing authentication secrets. RedLine Stealer, in particular, is a prevalent threat in the year 2023. I was able to stop the attack that night in its initial stages because I caught RedLine's data exfiltration and could initiate timely incident response[6].

Protect: DNS sinkholes can be leveraged to redirect domain name resolution to a benign server (see previous post). I recommend sinkholing subdomains of *.xyz, *.top, *.duckdns.org, *.ddns.net, and related meta-IOCs (assuming connections to those domains are not required in your environment). Sinkholing DNS requests for commonly abused domain infrastructure will proactively block connection attempts to many C2 servers of RedLine Stealer and related commodity malware (as well as some intrusive advertisements). Lists of specific IP addresses, such as the CSV data dumps on ThreatFox, can also be implemented to detect and block RedLine C2 connections. IP lists are of course limited as they require prior knowledge and constant updating. Additionally, infostealer C2 servers are often short-lived. Most RedLine IP addresses are only active for a day-or-two after public discovery by malware researchers.

```
I spend much of my research focused on tracking patterns among the
hostinginfrastructure of C2 servers. The SarlackLab server posts daily updates
ofpatterns to  and . I will be writing a future blog post onthese patterns (which I
refer to as "C2 hotspots") - but in the meantime, youcan find some of the tooling in
on GitHub.
```

Detect: It is important to layer defenses with detection capabilities so that response can be initiated **when** an attack bypasses existing protections. In addition to network protections, domain name patterns and IP lists can also be leveraged by network metadata and SIEM solutions to alert on potential RedLine communications to known servers. Network signatures can directly detected RedLine's C2 communications by exploiting patterns in the infostealer's custom protocol. I have written several Suricata signatures to detect RedLine Stealer and I am sharing a few TLP:CLEAR signatures regarding data exfiltration for on GitHub.

Due to the fact that RedLine Stealer is actively maintained and updated by MaaS cybercrime developers, I did skimp on certain details and defensive techniques in this blog post. Please message me on Twitter, if you have further questions on RedLine or would like access to some TLP:AMBER signatures. In the meantime, I hope the above advice helps you catch RedLine and related infostealers that threaten your environment… or at least that you have more luck than me and that Red Line train I just missed.

Sorry. As a Bostonian, I needed to include some reference to our local subway. And yes, I did miss the pictured Red Line train while writing part of this blog's conclusion.

Footnotes

[1] Security researchers and data scientists at Vectra AI classify such patterns among network metadata as C2 beaconing. Essentially, an apparent client and server have a role reversal where the "client" on an infected host begins behaving like a server — receiving requests from the C2 and returning data.

[2] I have heard rumors of RedLine proxying and encrypting its communications via Telegram bots — however **none** of the malware samples nor sandbox reports I investigated uncovered any such capabilities. I suspect that there may be some confusion with other information stealers or perhaps Redline's Telegram marketplace. Regardless, there are many command-and-control channels which **do** leverage Telegram (see LOTS Project) and I recommend closely investigating all apparent beacon communications with subdomains of telegram[.]org.

[3] TLP:CLEAR Suricata signatures matching RedLine Stealer's protocol for data exfiltration

```
alert tcp any any -> any any (msg:"Id7.languages"; flags:PA; content:"|06|";
startswith; content:"|01|"; within:3; content:"|1d|http|3a 2f
2f|tempuri|2e|org|2f|Entity|2f|Id7|09|languagesV|02 0b 01 73 04 0b 01 61 06 56 08 44
0a 1e 00 82 ab|"; within:60; content:"|40 0d|Authorization|08 03|ns1|99 20|";
within:23; content:"|44 1a ad|"; within:35; content:"|44 2c 44 2a ab 14 01 44 0c 1e
00 82 ab 03 01 56 0e 42 2b 0a 07 42|"; within:38; content:"|0b 01 62 2d 0b 01 69 15
45|"; within:10; content:"|29 01 01 01 01|"; endswith; sid:65001107;)alert tcp any
any -> any any (msg:"Id9.processes"; flags:PA; content:"|06|"; startswith;
content:"|1d|http|3a 2f 2f|tempuri|2e|org|2f|Entity|2f|Id9|09|processes"; within:43;
content:"|56 02 0b 01 73 04 0b 01 61 06 56 08 44 0a 1e 00 82 ab|"; within:25;
content:"|40 0d 41 75 74 68 6f 72 69 7a 61 74 69 6f 6e 08 03 6e 73 31 99 20|";
within:23; content:"|44 1a ad|"; within:35; content:"|44 2c 44 2a ab 14 01 44 0c 1e
00 82 ab 03 01 56 0e 42 33 0a 07 42|"; within:38; content:"|0b 01 62 2d 0b 01 69 15
45|"; within:10; content:"|99 27|ID|3a 20|"; within:7; content:"|01 01 01 01|";
endswith; sid:65001109;)alert tcp any any -> any any (msg:"Id13.installedBrowsers";
flags:PA; content:"|06|"; startswith; content:"|39 1e|http|3a 2f
2f|tempuri|2e|org|2f|Entity|2f|Id13|11|installedBrowsers|07|Entity4V|02 0b 01 73 04
0b 01 61 06 56 08 44 0a 1e 00 82 ab|"; within:78; content:"|40 0d|Authorization|08
03|ns1|99 20|"; within:23; content:"|44 1a ad|"; within:35; content:"|44 2c 44 2a ab
14 01 44 0c 1e 00 82 ab 03 01 56 0e 42 1f 0a 07 42|"; within:38; content:"|0b 01 62
13 0b 01 69 15 45|"; within:10; content:"|45 05 99|"; within:4; content:"|2e|exe|01
01 01 01 01|"; endswith; sid:65001113;)
```

[4] Cybercriminals deploying RATs will commonly leverage domain names and Dynamic DNS services at a much higher rate than those deploying infostealers. The infection of a Remote Access Trojan is expected to last longer than an infostealer due to the goals of the attack (ie exploiting remote access of a victim's computer V.S. quickly stealing specific information). My hypothesis is that RAT maintainers commonly leverage DDNS services in an attempt to make their C2 infrastructure resistant to take-downs — as the domain names can be rapidly shifted to new servers. In contrast, those who deploy infostealers are less-concerned with long lasting C2s and therefore will use normal DNS services or simply direct IP addresses. The idea is interesting to note for those of us who hunt C2 servers and is a point of future research.

[5] IPv4 comparison

IPv4 heatmaps comparing the similarity of RedLine Stealer C2 servers to ALL C2 servers tracked by SarlackLab.

[6] Thankfully, the IR and remediation efforts were not severe. I'd almost hesitate to call it an "incident" as there was no lost sleep and no systems encrypted — but technically it counts… I wish that all the IR engagements I got called for were as limited in destruction as this had been (though I imagine the attack would have progressed into something worse had it not been stopped).

Links

RedLine Stealer IOCs:

- ThreatFox —
- SarlackLab —
- (and of course *.xyz, *.top, *.duckdns.org, and *.ddns.net as well as other meta-IOCs)

See Also

- First public research on RedLine Stealer (early 2020) —
- How RedLine compromises web browser storage vault —
- RedLine Stealer MaaS documentation —

References

- Malware prevalence trends — , ,
- 2023 hack of Linus Tech Tips YouTube channel — ,
- tempuri.org documentation —
- NanoCore RAT research and DNS sinkholing —
- Vectra AI documentation of C2 beacons —
- Living Off Trusted Sites documentation of api[.]telegram[.]org abuse —