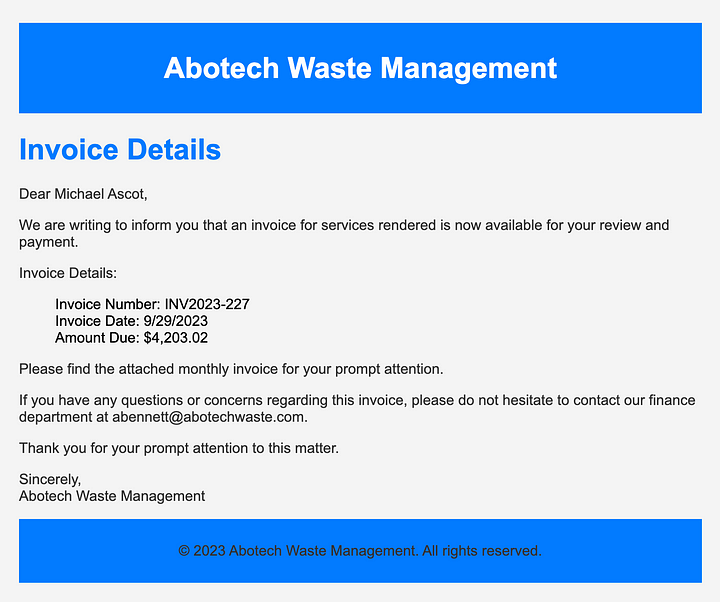
### **Hunt Me I: Payment Collectors**

Scenario: On **Friday, September 15, 2023**, Michael Ascot, a Senior Finance Director from SwiftSpend, was checking his emails in **Outlook** and came across an email appearing to be from Abotech Waste Management regarding a monthly invoice for their services. Michael actioned this email and downloaded the attachment to his workstation without thinking.



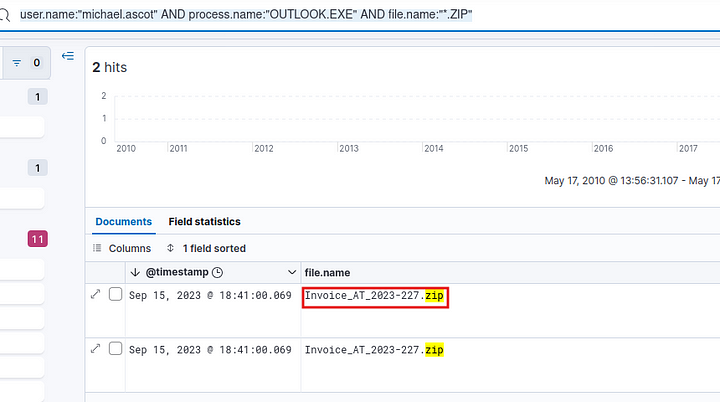
The following week, Michael received another email from his contact at Abotech claiming they were recently hacked and to carefully review any attachments sent by their employees. However, the damage has already been done. Use the attached Elastic instance to hunt for malicious activity on Michael’s workstation and within the SwiftSpend domain!

#### **TASKS**

1. **What was the name of the ZIP attachment that Michael downloaded?**

From the scenario we know that the user was Michael Ascot, the Mail User Agent was outlook. So we can filter the logs based on these details.

user.name:"michael.ascot" AND process.name:"OUTLOOK.EXE" AND file.name:"\*.ZIP"



Answer: Invoice\_AT\_2023–227.zip

**2. What was the contained file that Michael extracted from the attachment?**

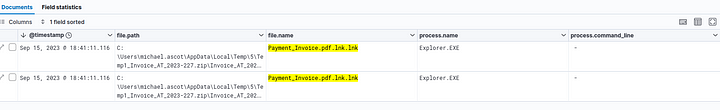
Let us search for logs with this zip file.



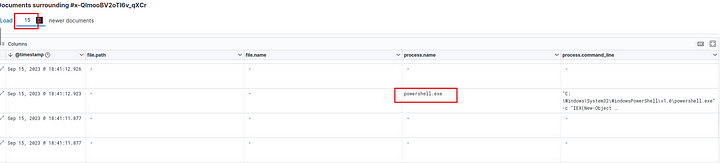
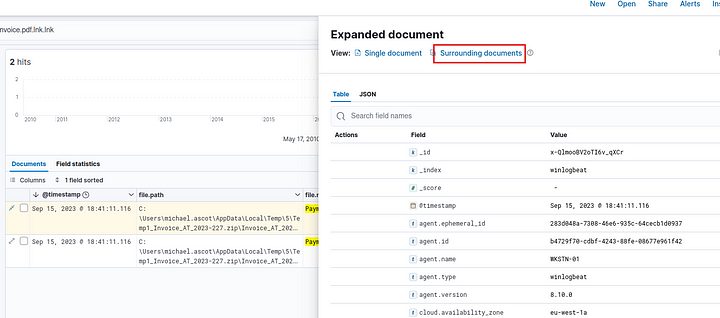
**Answer:** Payment\_Invoice.pdf.lnk.lnk

**3. What was the name of the command-line process that spawned from the extracted file attachment?**

Let us search for events with this file. Then view the surrounding documents. This will help us understand the flow of data.



As we can see above, we got two events with the file name. Now let us expand the first one and look for surrounding documents.



As we can see PowerShell was spawned from the extracted file and a command was executed.

**Answer:** PowerShell.exe

**4. What URL did the attacker use to download a tool to establish a reverse shell connection?**

In the previous question, we saw powershell was spawned and a command was executed. Let us look at this command closely.



**Answer:** https://raw.githubusercontent.com/besimorhino/powercat/master/powercat.ps1

**5. What port did the workstation connect to the attacker on?**

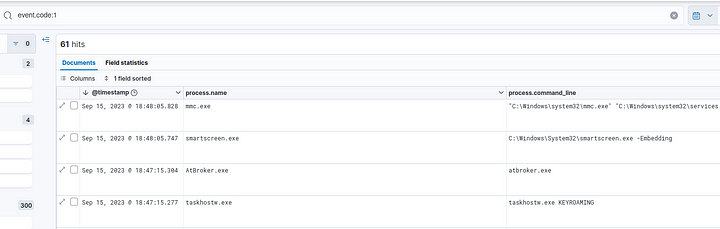
****

**Answer:** 19282

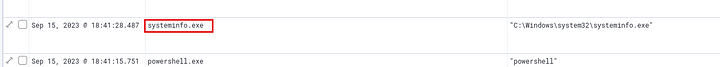
**6. What was the first native Windows binary the attacker ran for system enumeration after obtaining remote access?**

Look for process creation evetns with event id 1.

event.code:1



Now, we can look for any binaries that are commonly used for enumeration.



Answer: systeminfo.exe

**7. What is the URL of the script that the attacker downloads to enumerate the domain?**

Let us filter on events with process name powershell.exe

process.name:powershell.exe



While looking through the logs we find a script named powerview.ps1



Now let us expand this and view the surrounding documents.

In the surrounding documents, we find a web request.

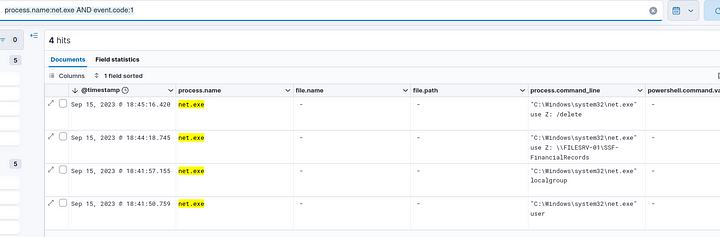


**Answer:** https://raw.githubusercontent.com/PowerShellEmpire/PowerTools/master/PowerView/powerview.ps1

**8. What was the name of the file share that the attacker mapped to Michael’s workstation?**

While looking for the answer on question 6, I found that the process net.exe was also used. This is commonly used to manage shared resources. so let us filter on this process.

process.name:net.exe AND event.code:1



In the second record, we can see the command used to map the network share to a drive.



**Answer:** SSF-FinancialRecords

**9. What directory did the attacker copy the contents of the file share to?**

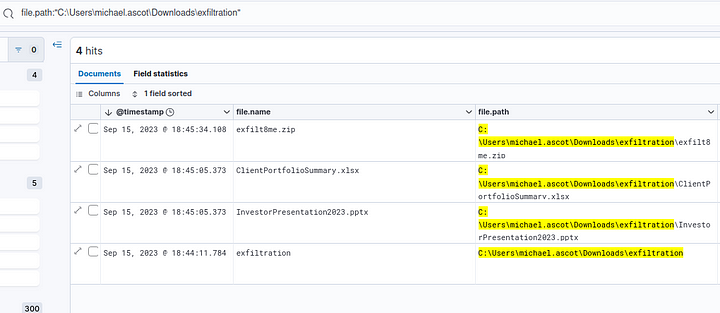
Filter on file creation events.

event.code:11



We can see that a directory called exfiltration was created. Now, let us filter on this path.

file.path:"C:\Users\michael.ascot\Downloads\exfiltration"



We got 4 hits. Let us examine the first file created in this folder, which is InvestorPresentation2023.pptx. Expand the record and view details.



We can see that robocopy.exe was used, which is a command line tool used to effortlessly copy files without worrying the permissions. So we can conclude that this is the path that the attacker used to copy the contents of the file share.

**Answer:** C:\Users\michael.ascot\Downloads\exfiltration

**10. What was the name of the Excel file the attacker extracted from the file share?**

We can see the name of the file in the last question.

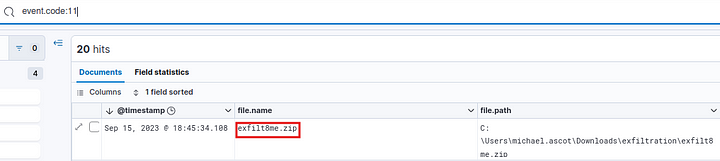


**Answer:** ClientPortfolioSummary.xlsx

**11. What was the name of the archive file that the attacker created to prepare for exfiltration?**

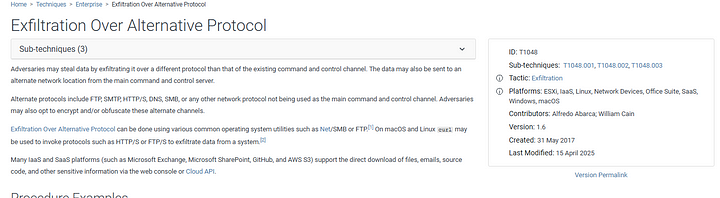
Back to the file creation records.

event.code:11



**Answer:** exfilt8me.zip

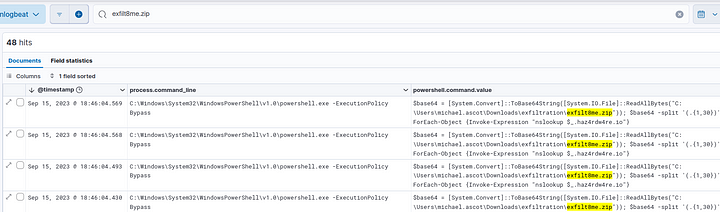
**12. What is the MITRE ID of the technique that the attacker used to exfiltrate the data?**

****

**Answer:** T1048

**13. What was the domain of the attacker’s server that retrieved the exfiltrated data?**

Filter records with the above identified archive file.



We can see the command that is sending data over and the attacker’s domain is clear from the command.

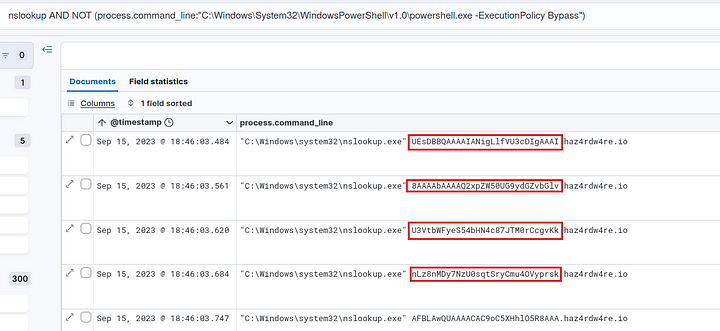


**Answer:** haz4rdw4re.io

**13. The attacker exfiltrated an additional file from the victim’s workstation. What is the flag you receive after reconstructing the file?**

Since we already saw that the attacker used nslookup to exfiltrate ata, we can filter for events with this.

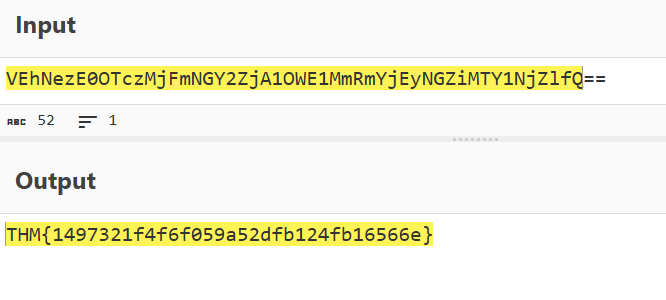
nslookup AND NOT (process.command\_line:"C:\Windows\System32\WindowsPowerShell\v1.0\powershell.exe -ExecutionPolicy Bypass")



We can see the base64 strings, we need to decode them to find the flag.



The last two records had the flag.



**Answer:** THM{1497321f4f6f059a52dfb124fb16566e}

This is the end of this walkthrough.