1. Getting Set Up

Before we start hunting, make sure your environment is ready.

1. **Deploy the Machine:** Get that host IP address (e.g., 10.10.247.19) and connect either via the AttackBox or VPN. It might take 3–5 minutes to start.

2. **Access Logs:** All the necessary logs are pre-ingested into Splunk under the index name **win_eventlogs**.

Context: The Network Segments

Keep these departments in mind, especially when we start filtering for HR users:

Department	Users
IT	James, Moin, Katrina
HR (Our Target)	Haroon, Chris, Diana
Marketing	Bell, Amelia, Deepak

2. Initial Log Reconnaissance

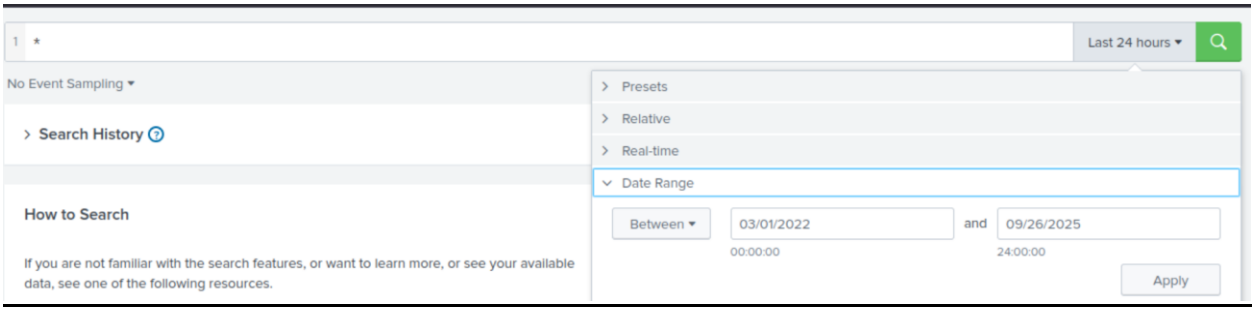
First things first: let's get a feel for the data volume and make sure we don't miss any obvious red flags.

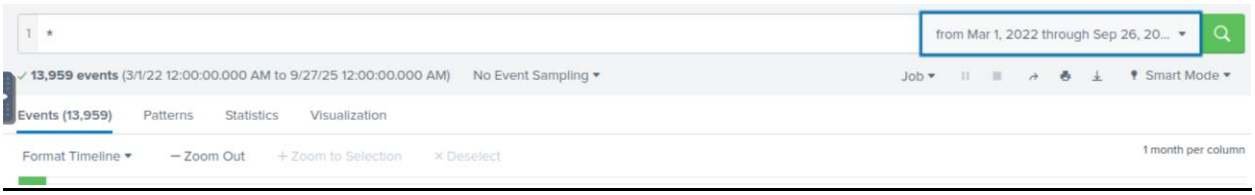
Query 1: Log Volume Check

It's always good practice to confirm the time frame we are analyzing.

• *Task:* How many logs did we actually ingest from March 2022?

• *Answer:* You should find **13959** logs from that month.



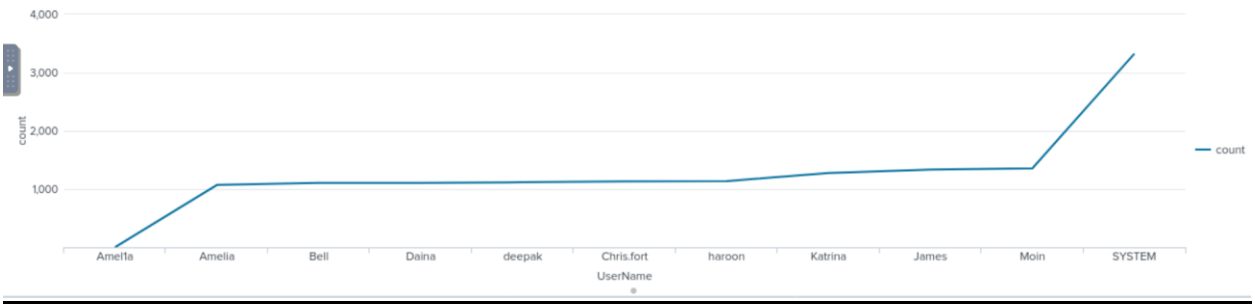


Query 2: Imposter Alert! 🚨

One common trick attackers use is creating accounts that look similar to legitimate ones. We need to check for a sneaky imposter account.

• *Task:* Find the name of the user that looks suspicious.

• *How to find the imposter (Pro Tip!):* Select the UserName field to see who the usernames are. You will find that there are about 11 unique user names in the logs, but the imposter is probably not in the top 10 most frequent values. To spot them quickly, click on **Rare Values** to find the user **Amel1a**.



---

3. Investigating the HR Compromise

We've confirmed the initial suspicion that a host in the HR department was potentially compromised, and tools related to scheduled tasks or network gathering were run.

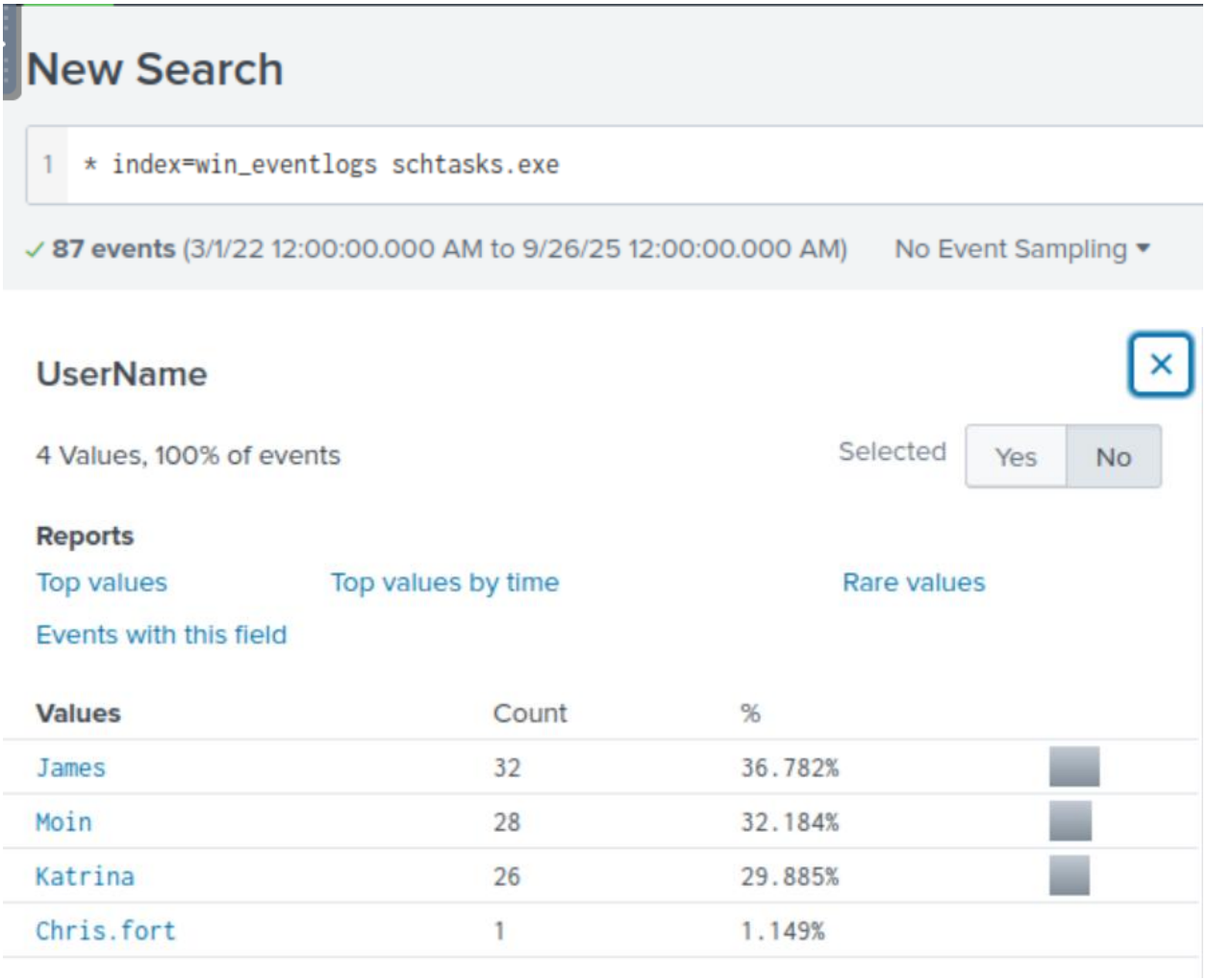
Query 3: Scheduled Task Persistence

Attackers love persistence, and running scheduled tasks is a classic method. The application name for this is **schtasks.exe**.

• *Task:* Which user from the HR department was running scheduled tasks?

• *Approach:* Filter the logs for Process Application Name="schtasks.exe". While four people used this app, only one belongs to the HR department.

• *Answer:* The HR user running scheduled tasks is **Chris.fort**.



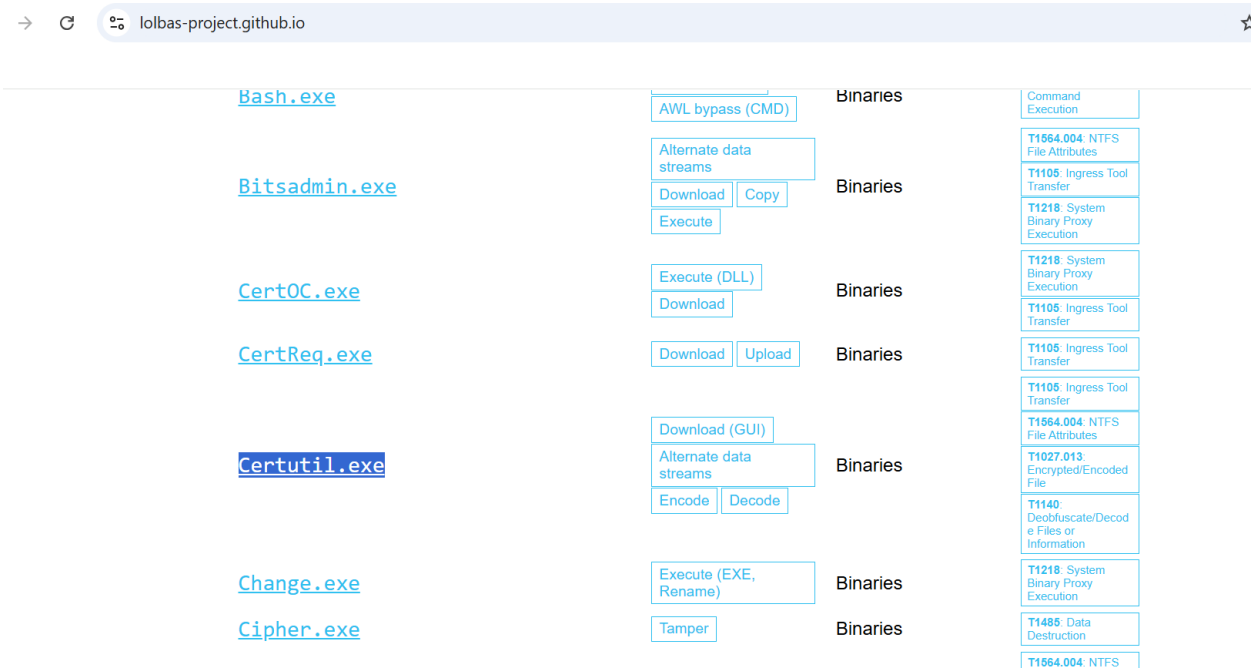
4. Identifying the Payload Download (The LOLBIN)

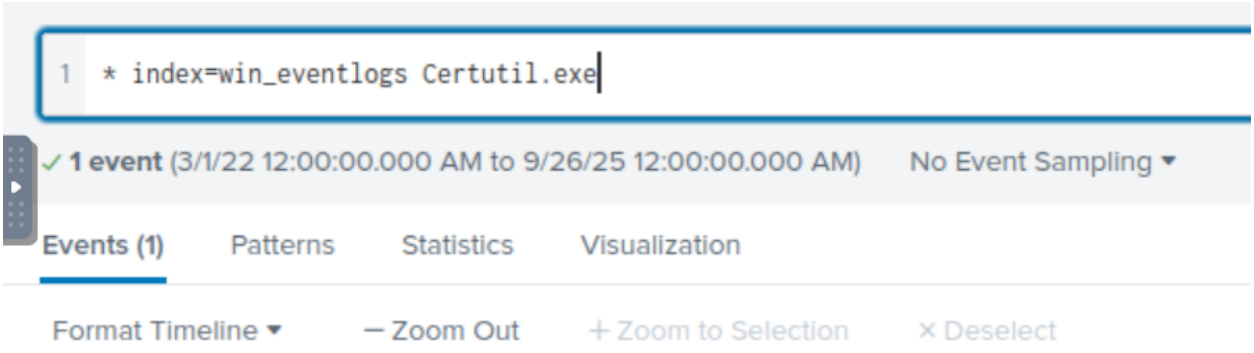
This is where the real action happened. We are looking for an HR user who utilized a system process—a **LOLBIN (Living Off The Land Binary)**—to download a payload, likely bypassing security controls.

• *If you need to brush up on LOLBINs,* check out resources like lolbas-project.github.io/.

Query 4: Who and What?

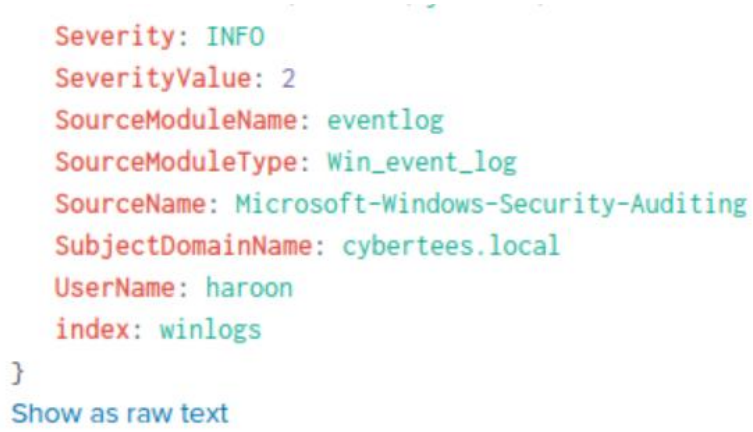
We need to figure out who initiated the download and which built-in binary they used.





• *Task (User):* Which user from HR executed a system process to download a payload from a file-sharing host?

• *Answer (User):* The user observed executing this was **Haroon**.



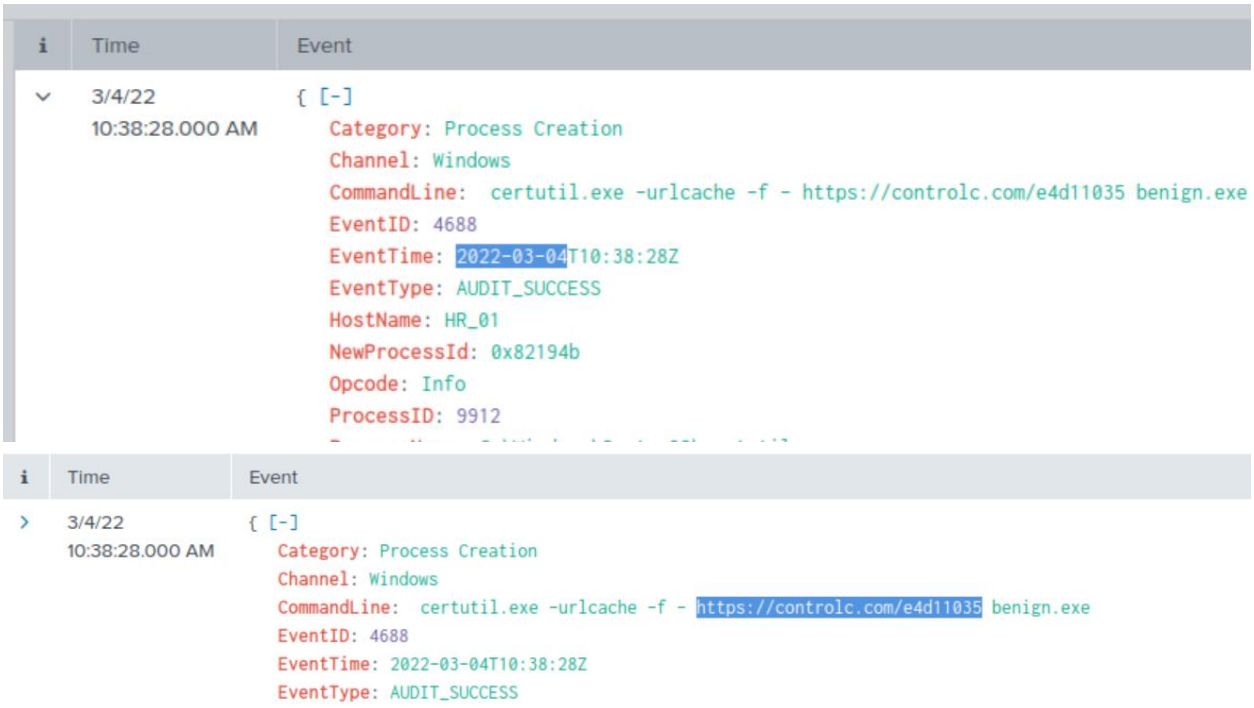
• *Task (Binary):* Which system process (LOLBIN) was used to bypass security controls and download the file?

• *Answer (Binary):* After filtering potential LOLBINs, we hit a match: **certutil.exe**.

Query 5, 6, & 7: Extracting the Attack Details

Now that we have the log entry showing Haroon running certutil.exe, we can extract all the critical details.

Detail	Task	Answer
Date	When was the binary executed? (YYYY-MM-DD)	**2022-03-04**
C2 Site	Which third-party site provided the payload?	**controlc.com**
Filename	What was the name of the file saved on the host machine?	**benign.exe**



---

5. Finalizing the Connection

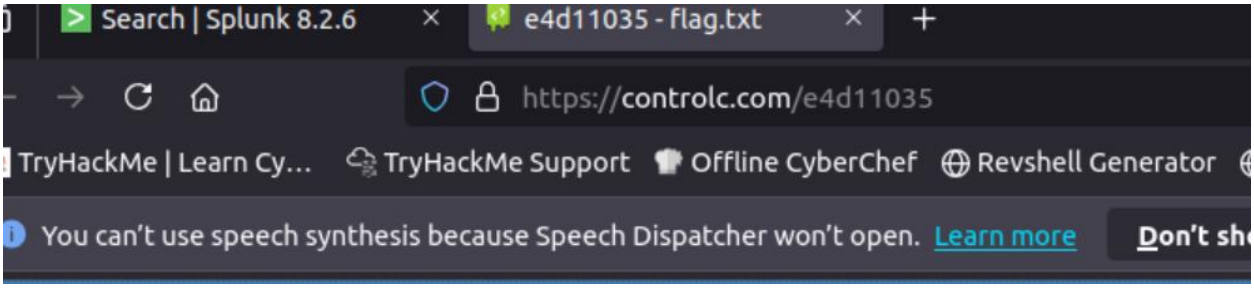
We've tracked the execution, the date, and the filename. Now for the last piece of the puzzle: the exact connection details and the content itself.

Query 8 & 9: URL and Flag

The downloaded file came from a Command and Control (C2) server, and it contained a specific pattern we need to capture.

• *Task (URL):* What is the full URL that the infected host connected to?

• *Answer (URL):* The C2 server URL is **https://controlc.com/e4d11035**.



• *Task (Flag):* The file downloaded from the C2 server contained a malicious pattern: THM{.....}. What is that pattern?

• *Approach:* You'd typically access this C2 server in a safe environment (sandbox) to retrieve the content.

• *Answer (Flag):* The pattern is **THM{KJ&*H^B0}*