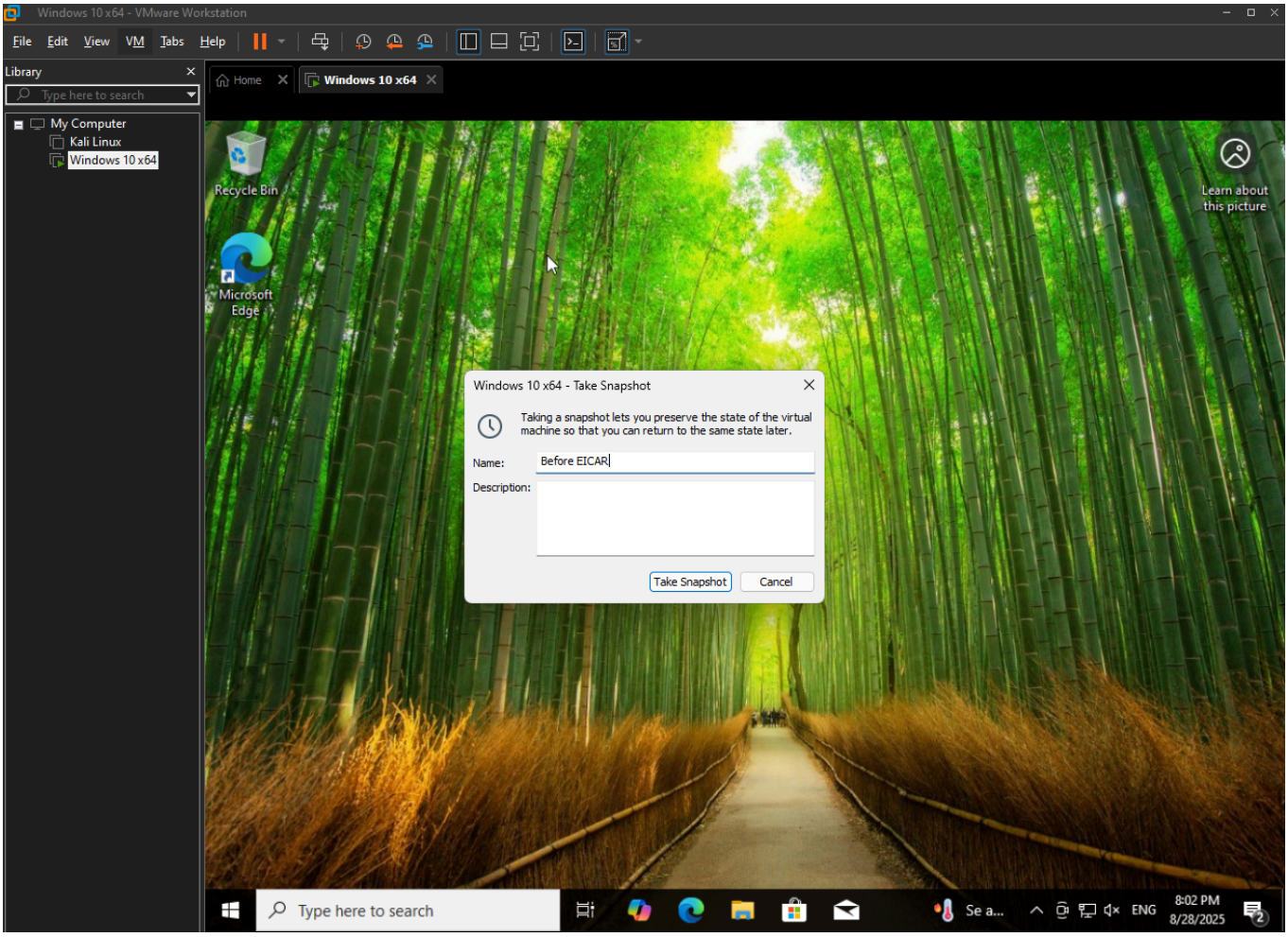
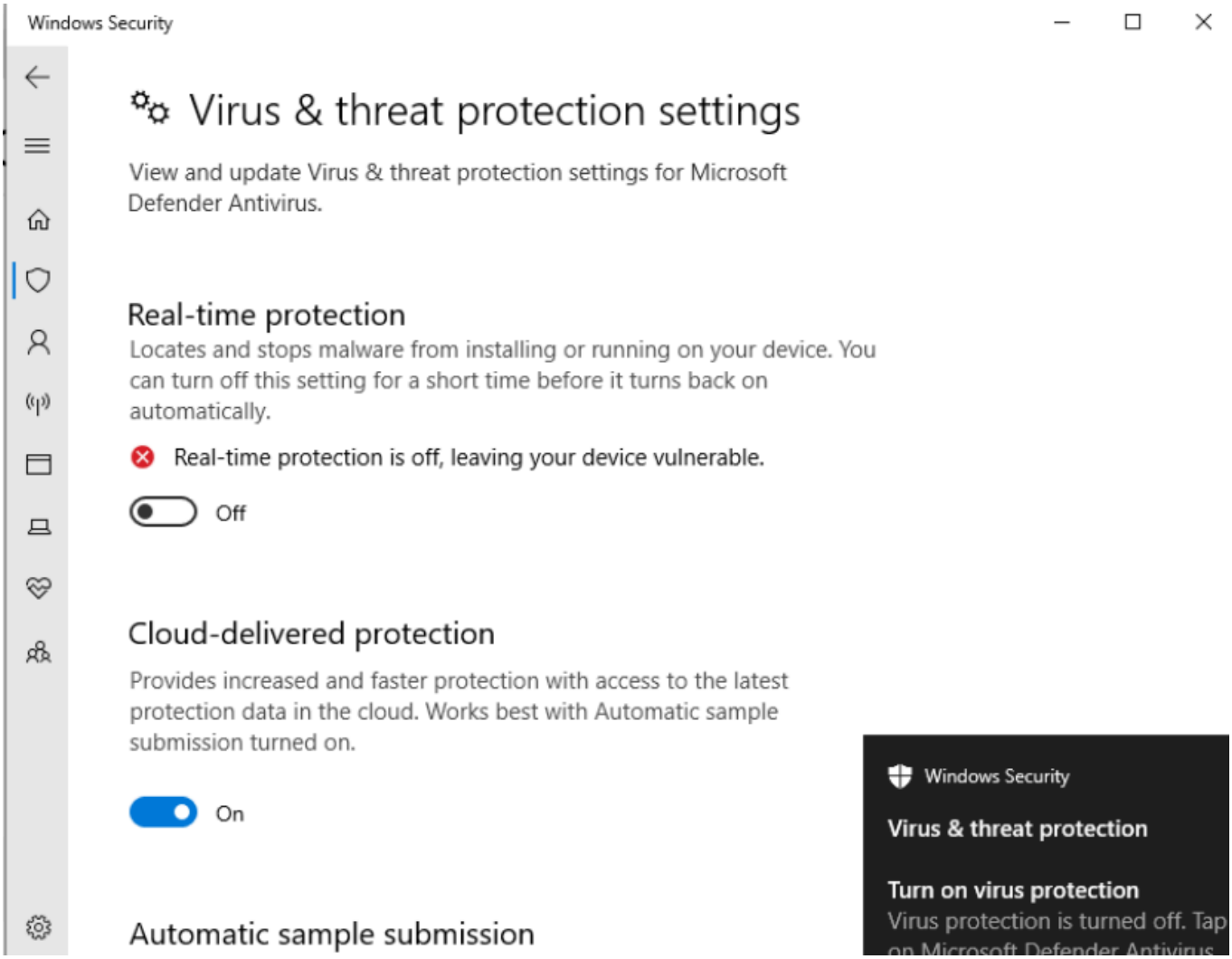
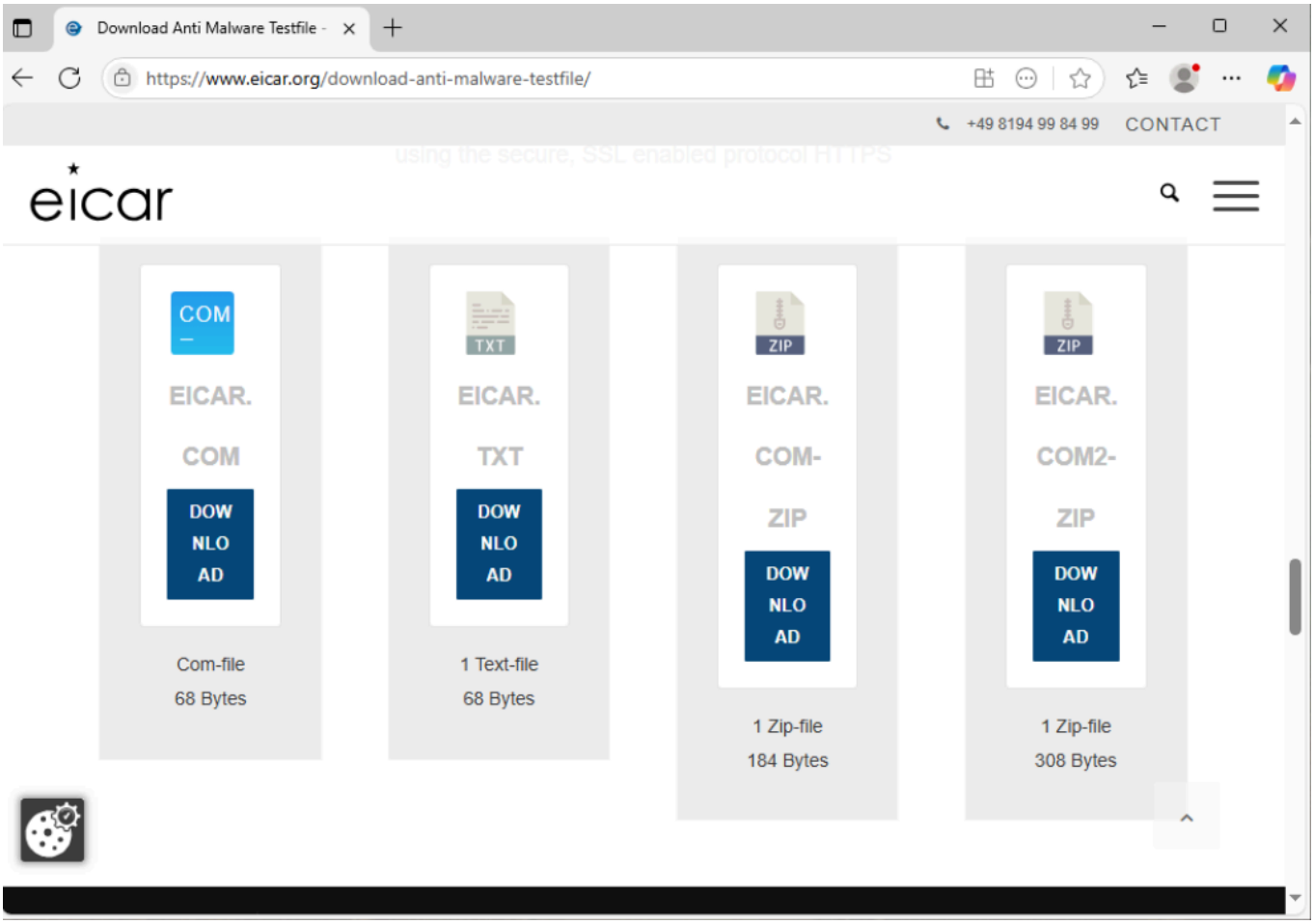
Firstly, we configure the Windows machine with a NAT network adapter and establish the IP address to be dynamically obtained.
We take a snapshot of the current configuration



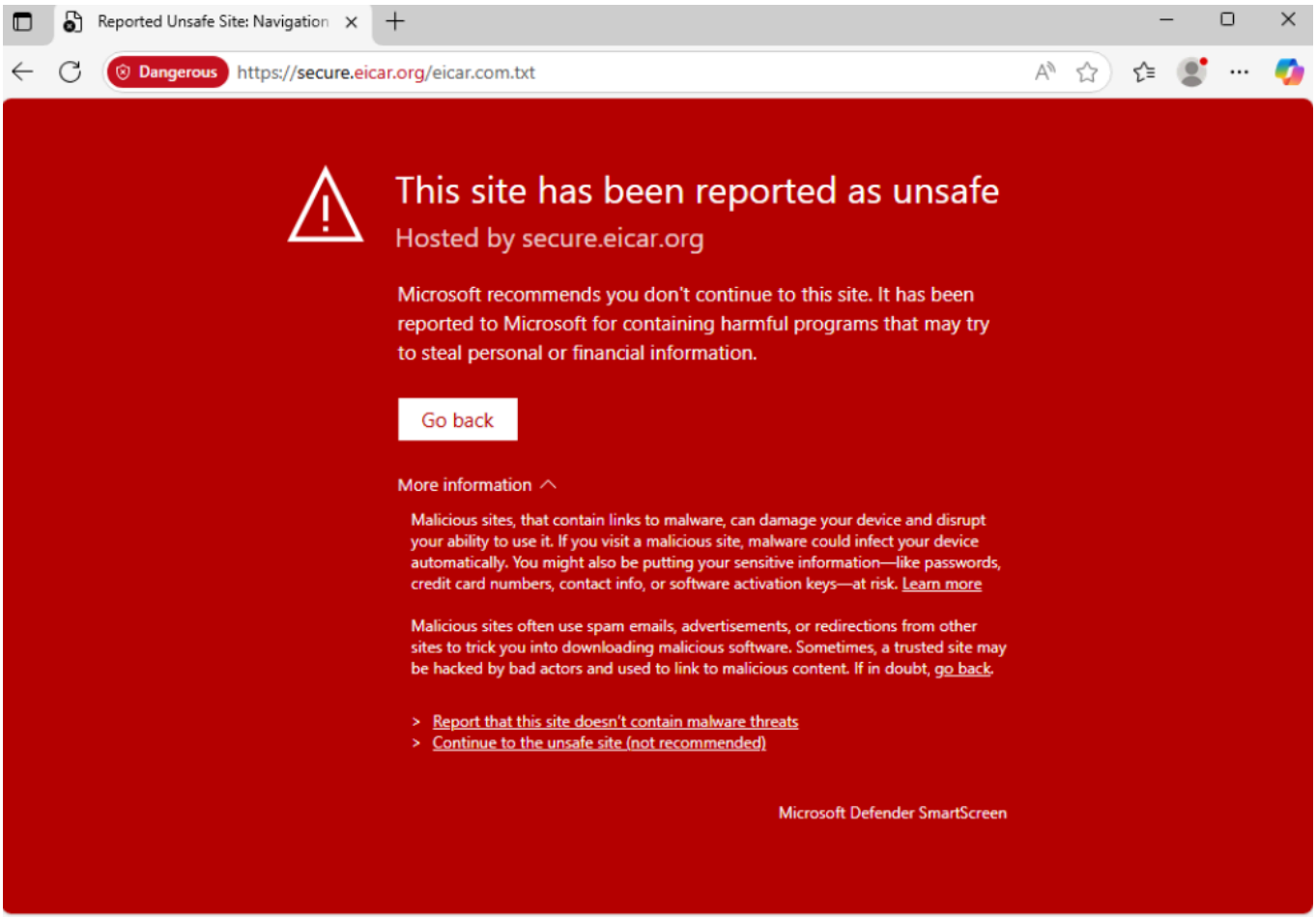
We turn off real time protection



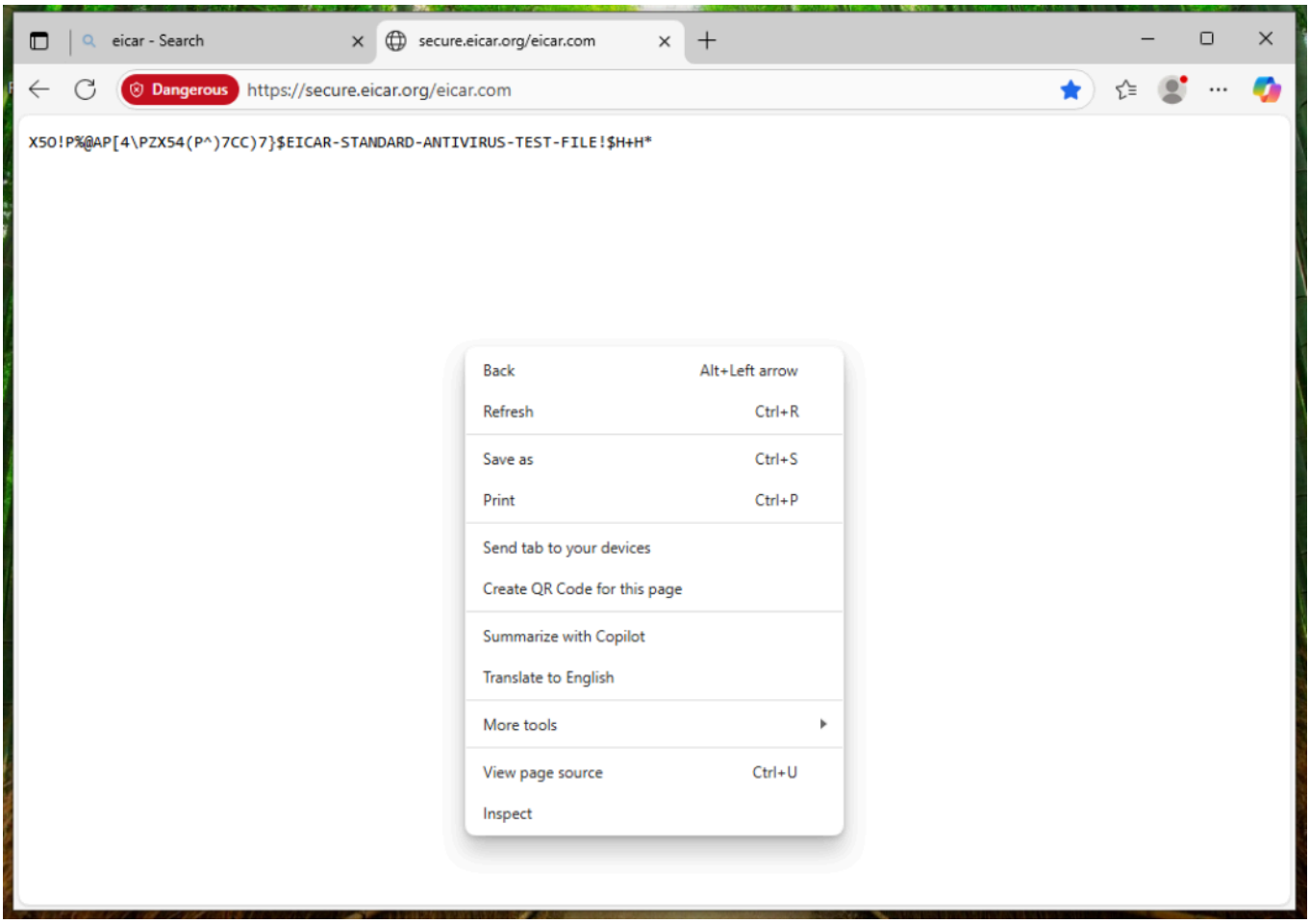
We click the .com file on the eicar website



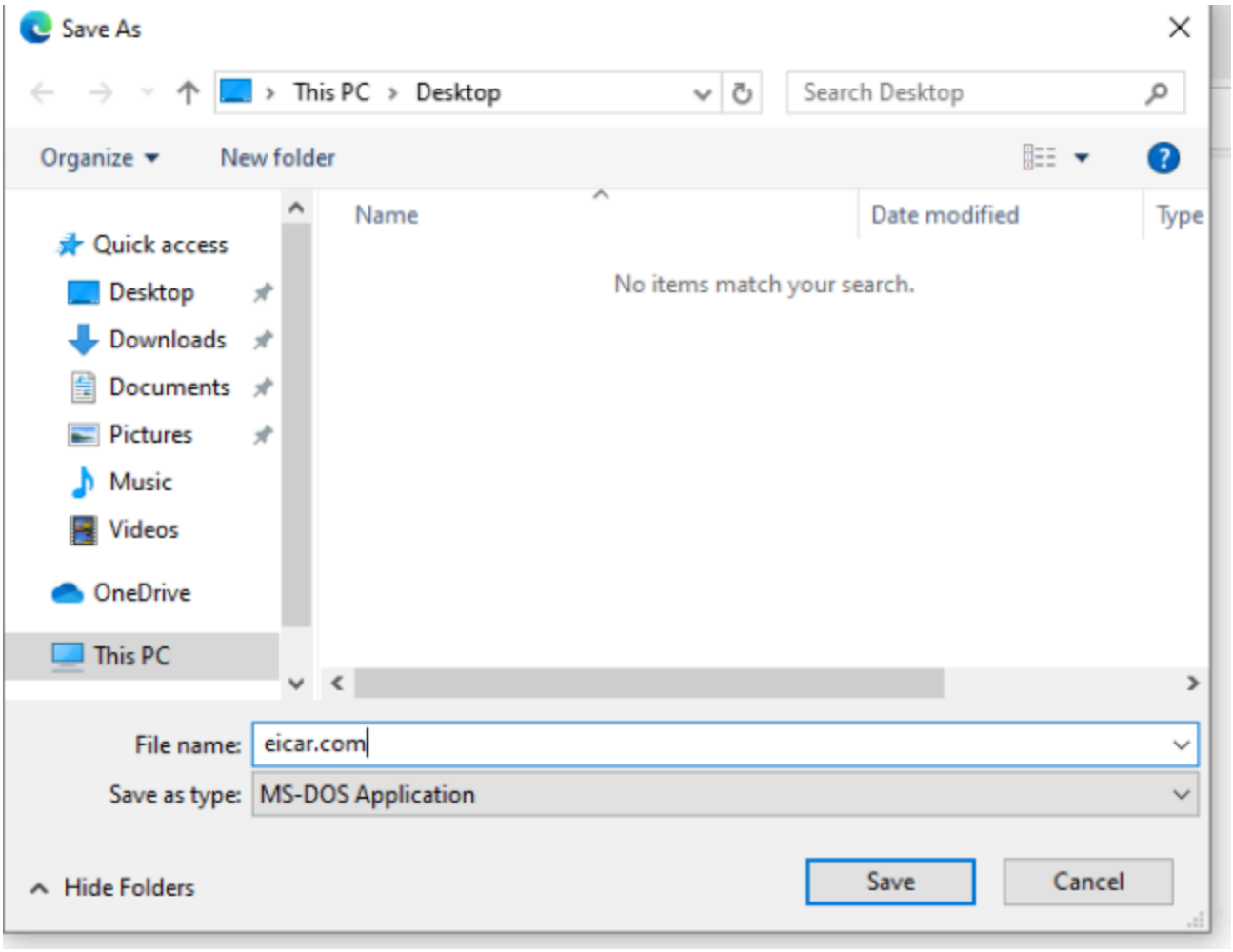
We may receive warnings when trying to download it but we do it anyway, this file wont harm the machine.


Reported Unsafe Site: Navigation
Dangerous
https://secure.eicar.org/eicar.com.txt
This site has been reported as unsafe
Hosted by secure.eicar.org
Microsoft recommends you don't continue to this site. It has been reported to Microsoft for containing harmful programs that may try to steal personal or financial information.
Go back
More information
Malicious sites, that contain links to malware, can damage your device and disrupt your ability to use it. If you visit a malicious site, malware could infect your device automatically. You might also be putting your sensitive information—like passwords, credit card numbers, contact info, or software activation keys—at risk. Learn more
Malicious sites often use spam emails, advertisements, or redirections from other sites to trick you into downloading malicious software. Sometimes, a trusted site may be hacked by bad actors and used to link to malicious content. If in doubt, go back.
> Report that this site doesn't contain malware threats
> Continue to the unsafe site (not recommended)
Microsoft Defender SmartScreen

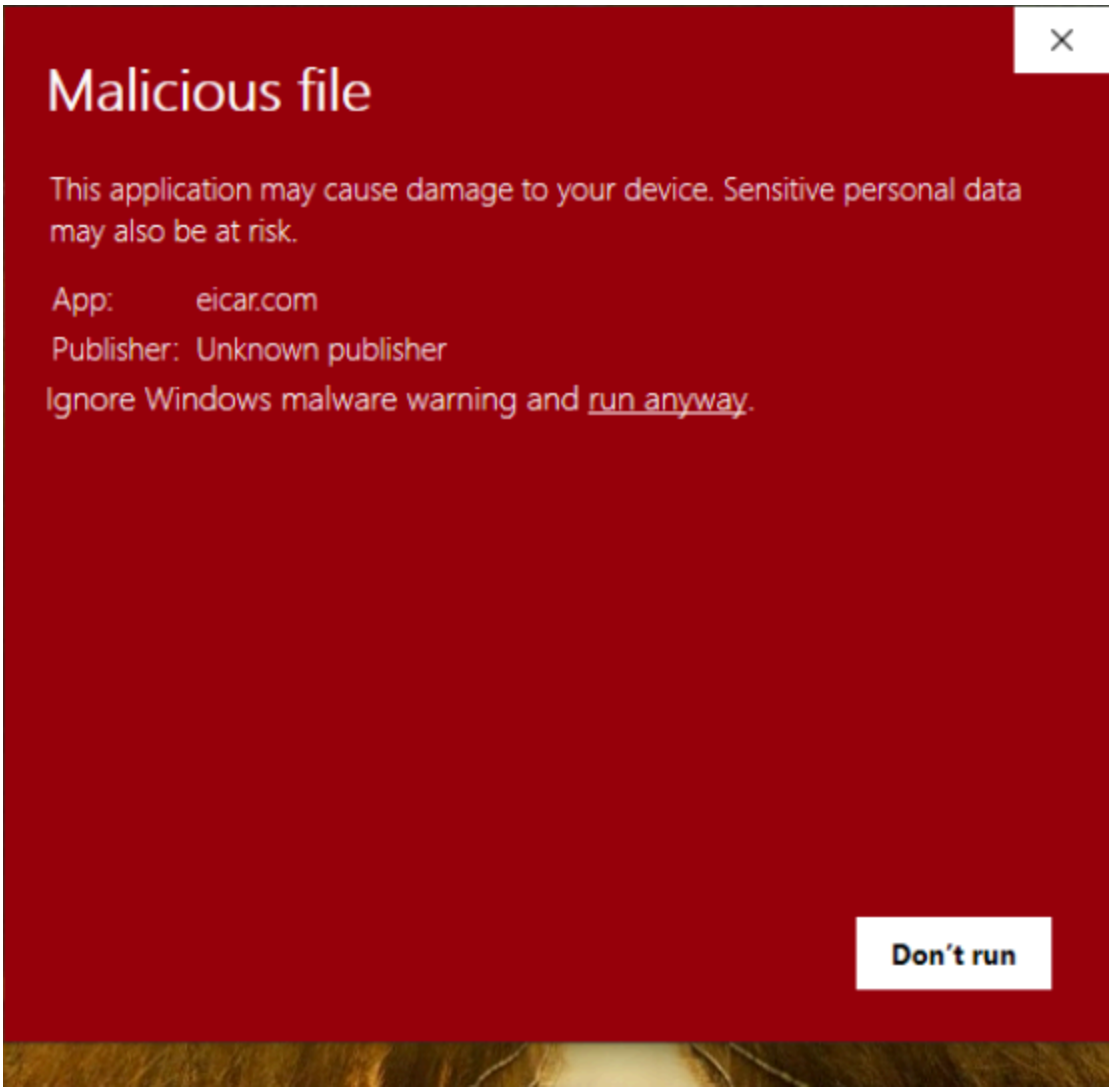
We right-click and hit save as



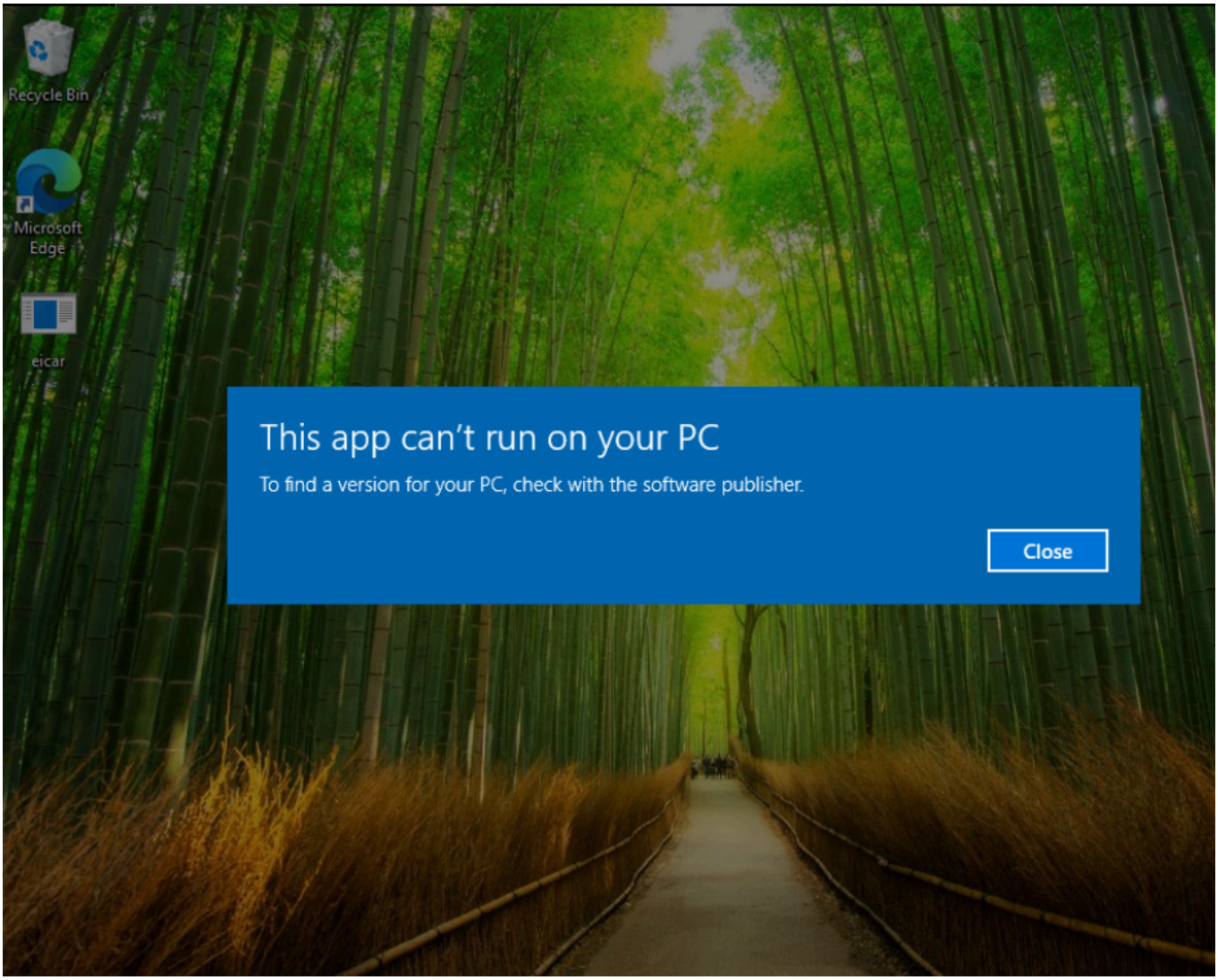
We save it as eicar.com:



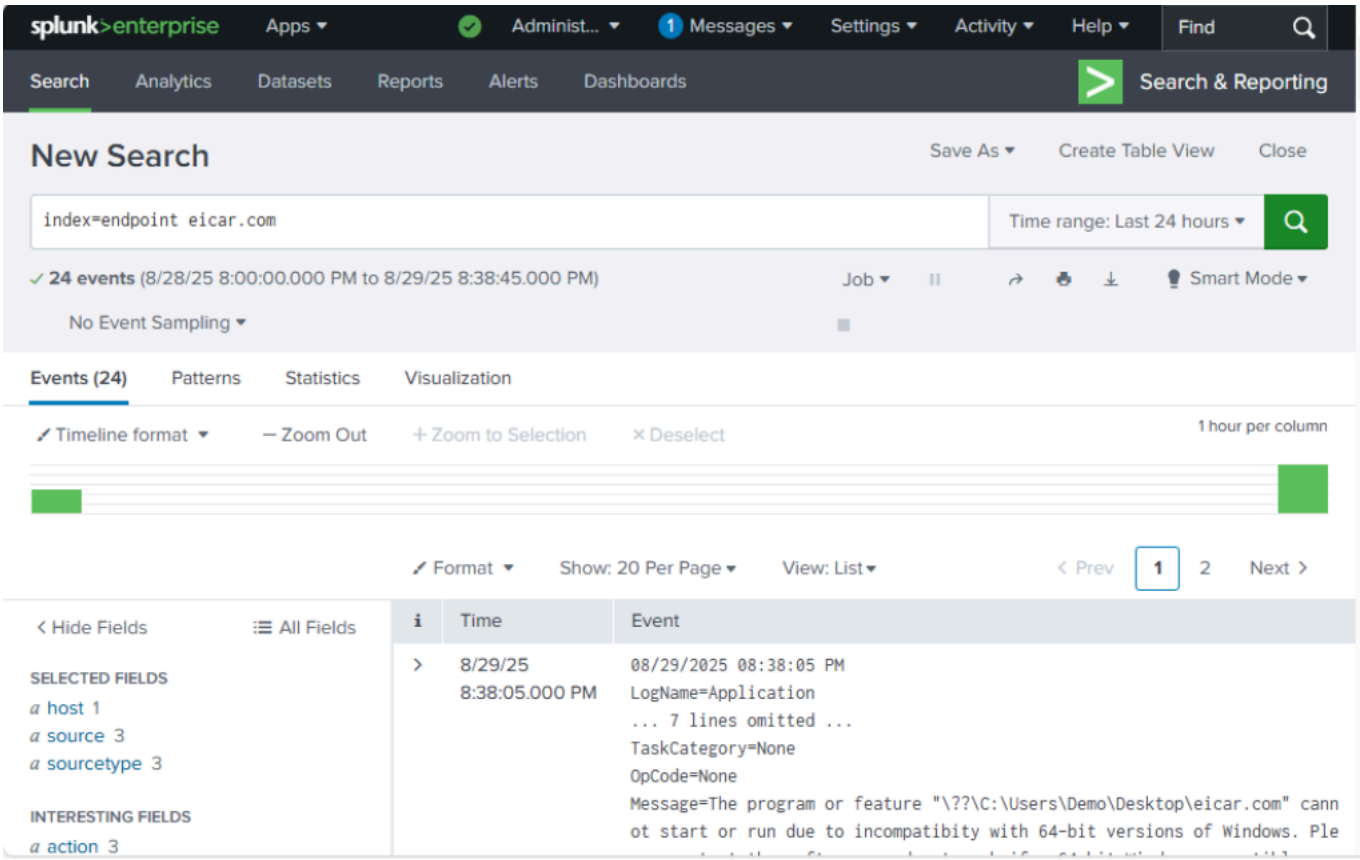
We know execute it, it will throw the following warning, we click run anyway:


Malicious file
This application may cause damage to your device. Sensitive personal data may also be at risk.
App: eicar.com
Publisher: Unknown publisher
Ignore Windows malware warning and run anyway.
Don't run

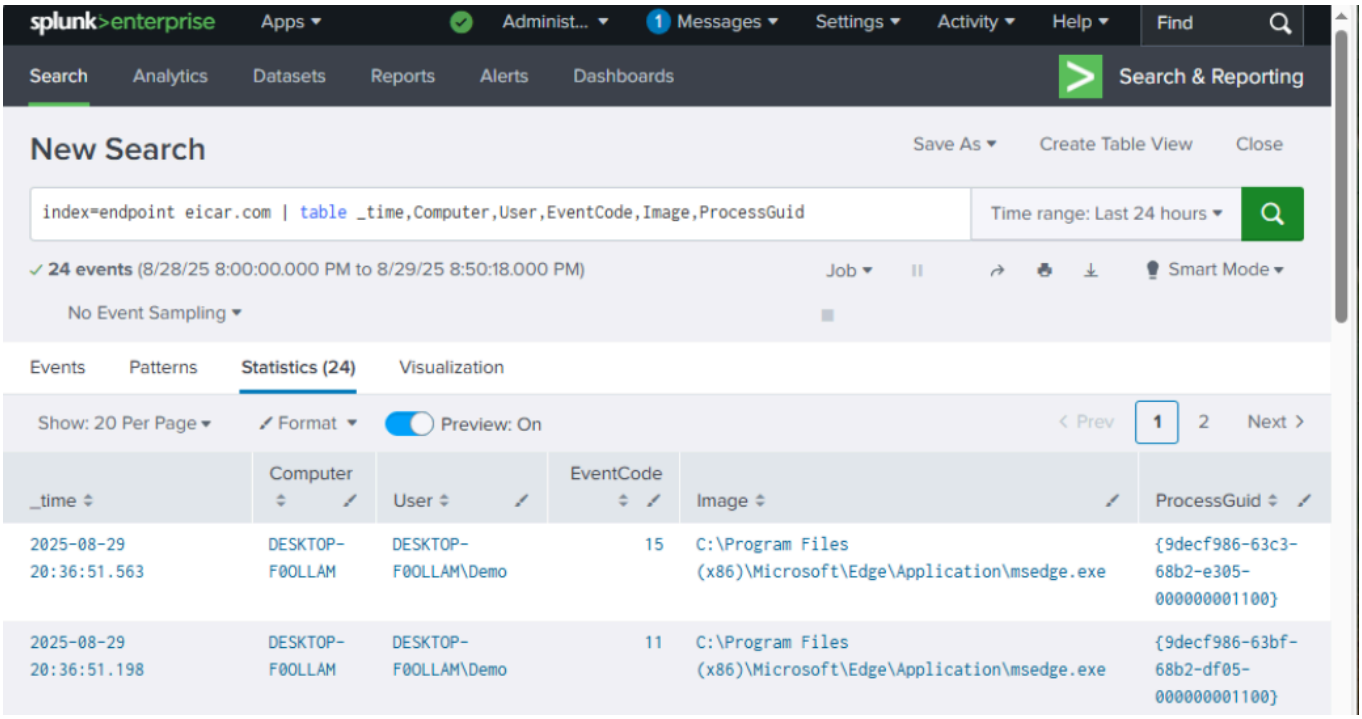
We will find the following pop up:



Now, we go to Splunk and we search for the following (you may have less events registered as I have run it several times):



We now can create a table to visualize general information about this malware file:

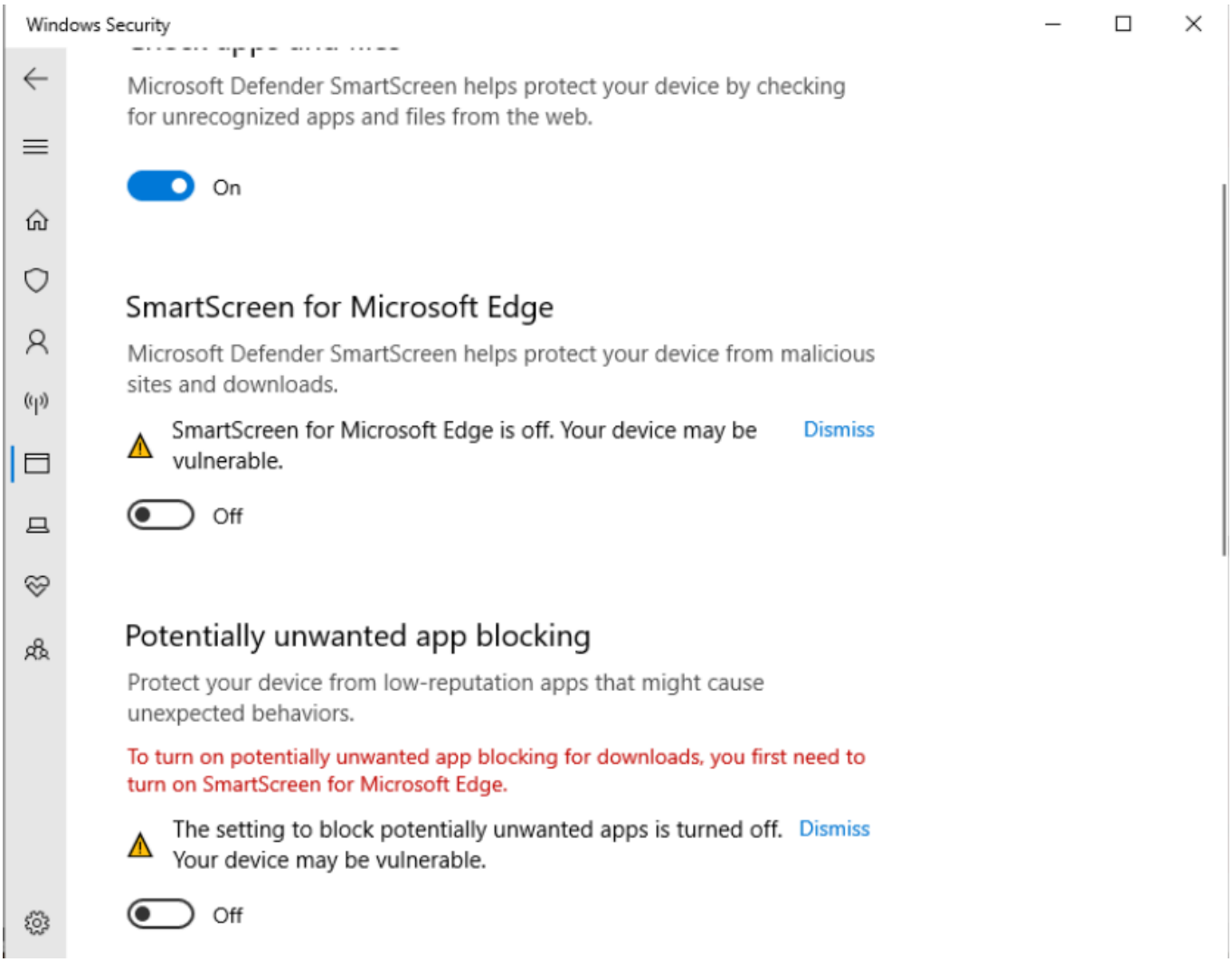


While this malware file is a test file and doesn't really affect our equipment, it is a good way to test both sysmon and our antivirus.

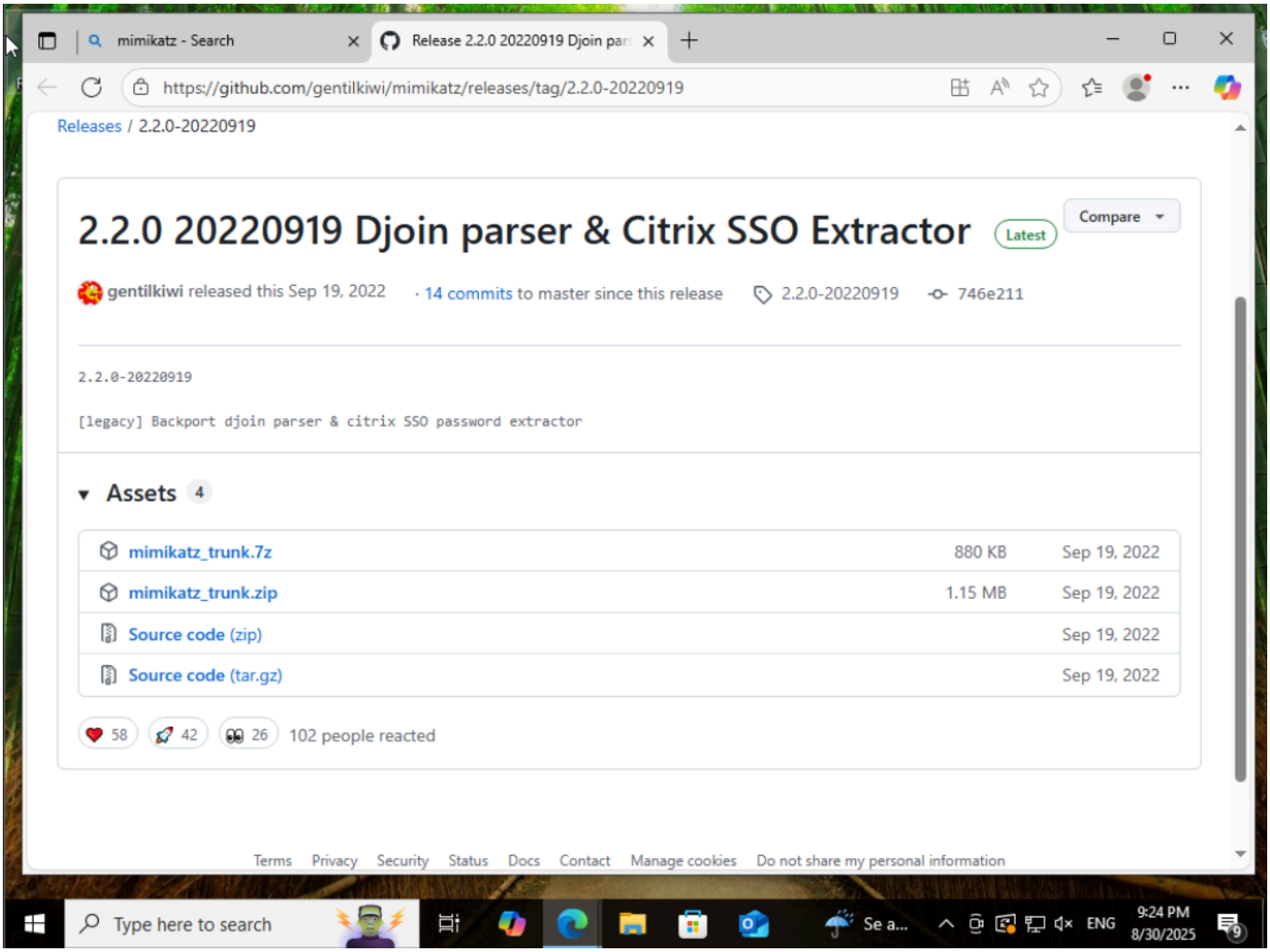
Now, we will work with Mimikatz. Mimikatz is a widely-used post-exploitation tool designed to extract sensitive information, such as plaintext passwords, hashes , and Kerberos tickets, from system memory. In this lab, we will be using Mimikatz to exploit LSASS (Local Security

Authority Subsystem Service). This is the Windows process in charge of holding hashes, Kerberos tickets and passwords.

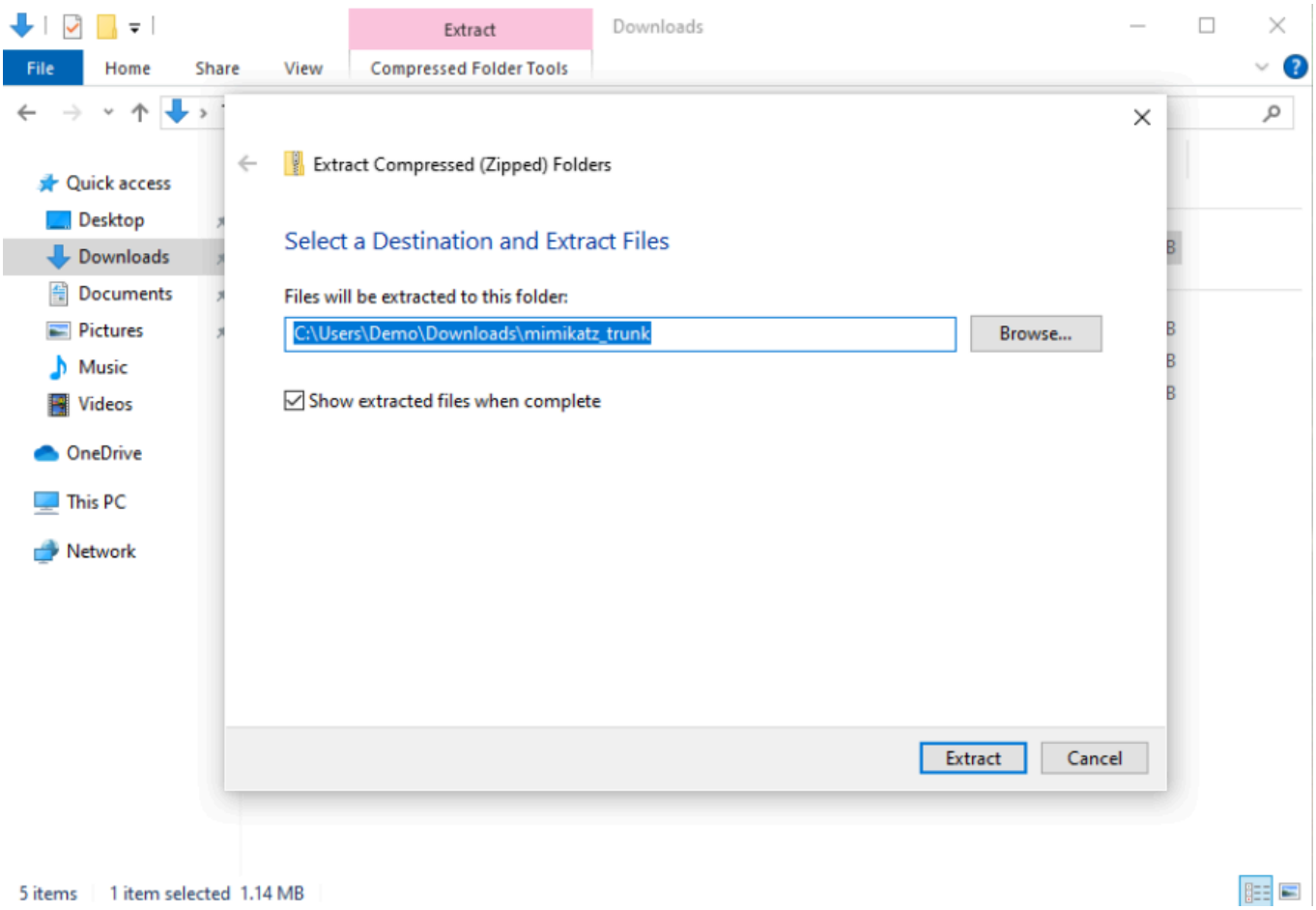
We first turn down smartscreen on our device:



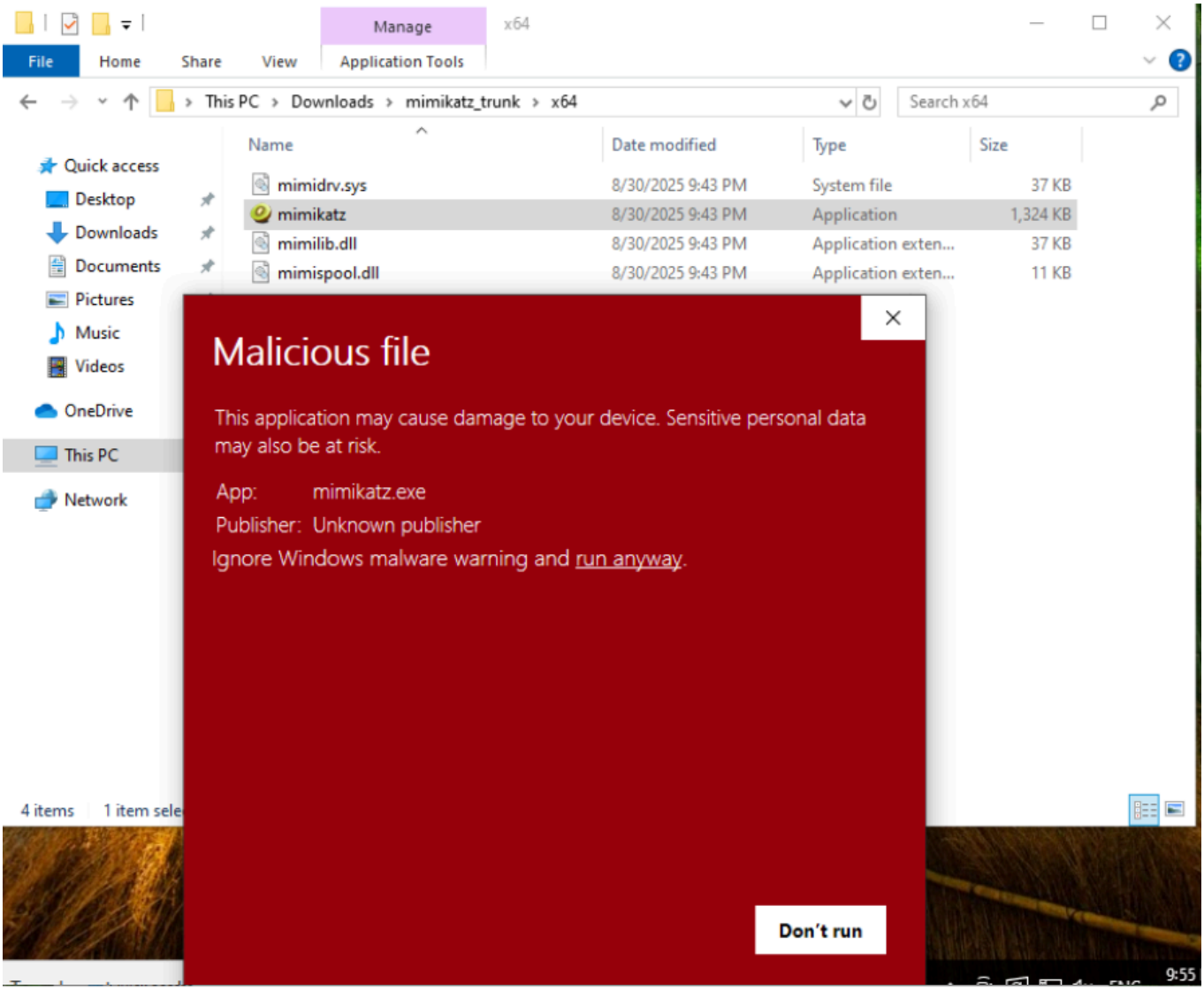
Now, we download the Mimikatz file:

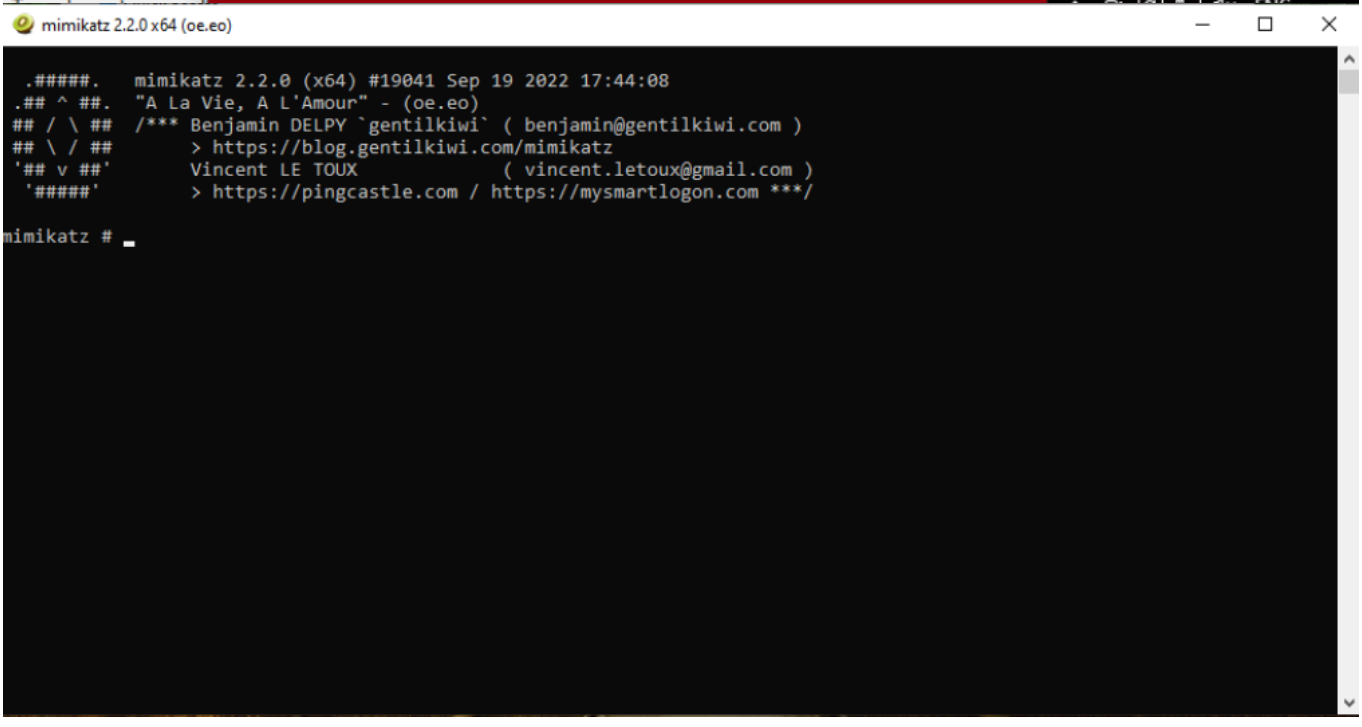


Then, we extract the file:



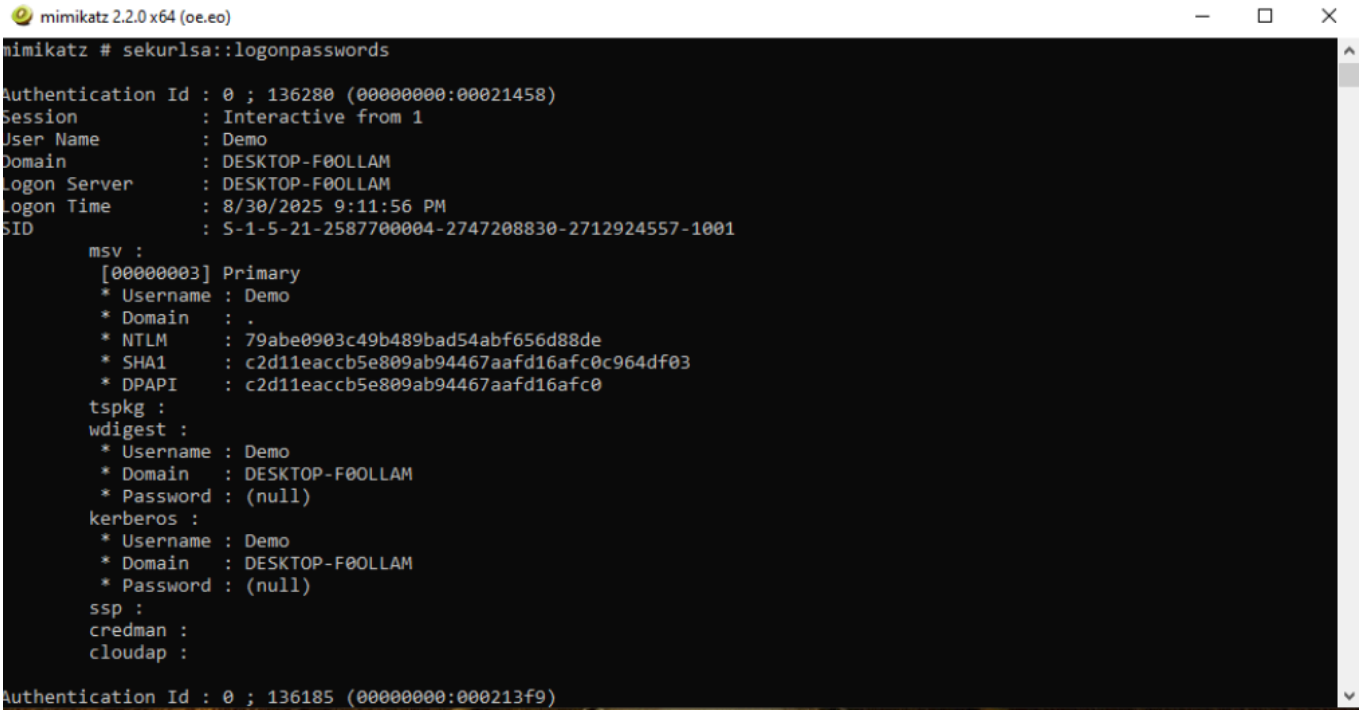
We go to the x64 folder and run mimikatz as administrator



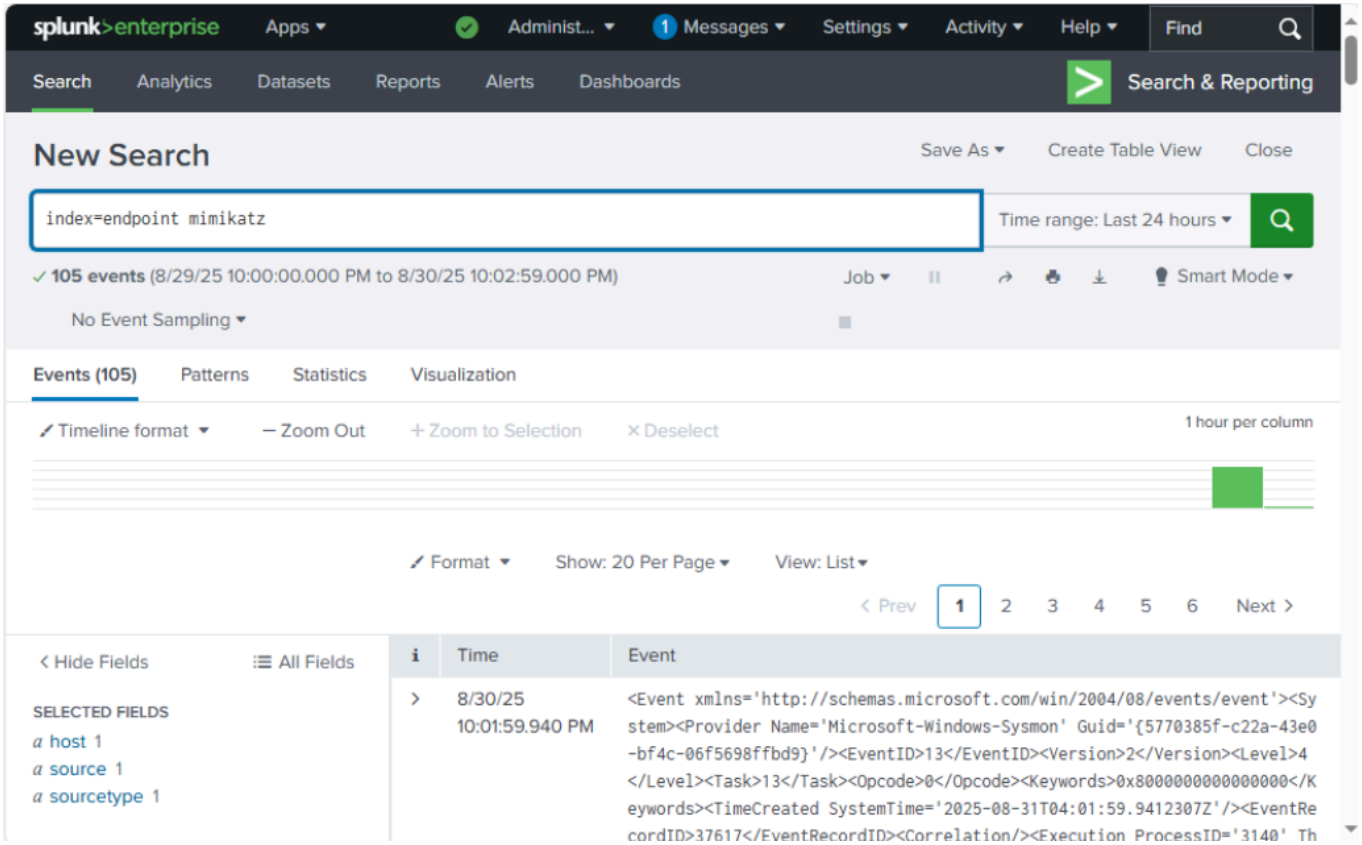


We first run the following command, which gives Mimikatz the rights to peek inside LSASS:

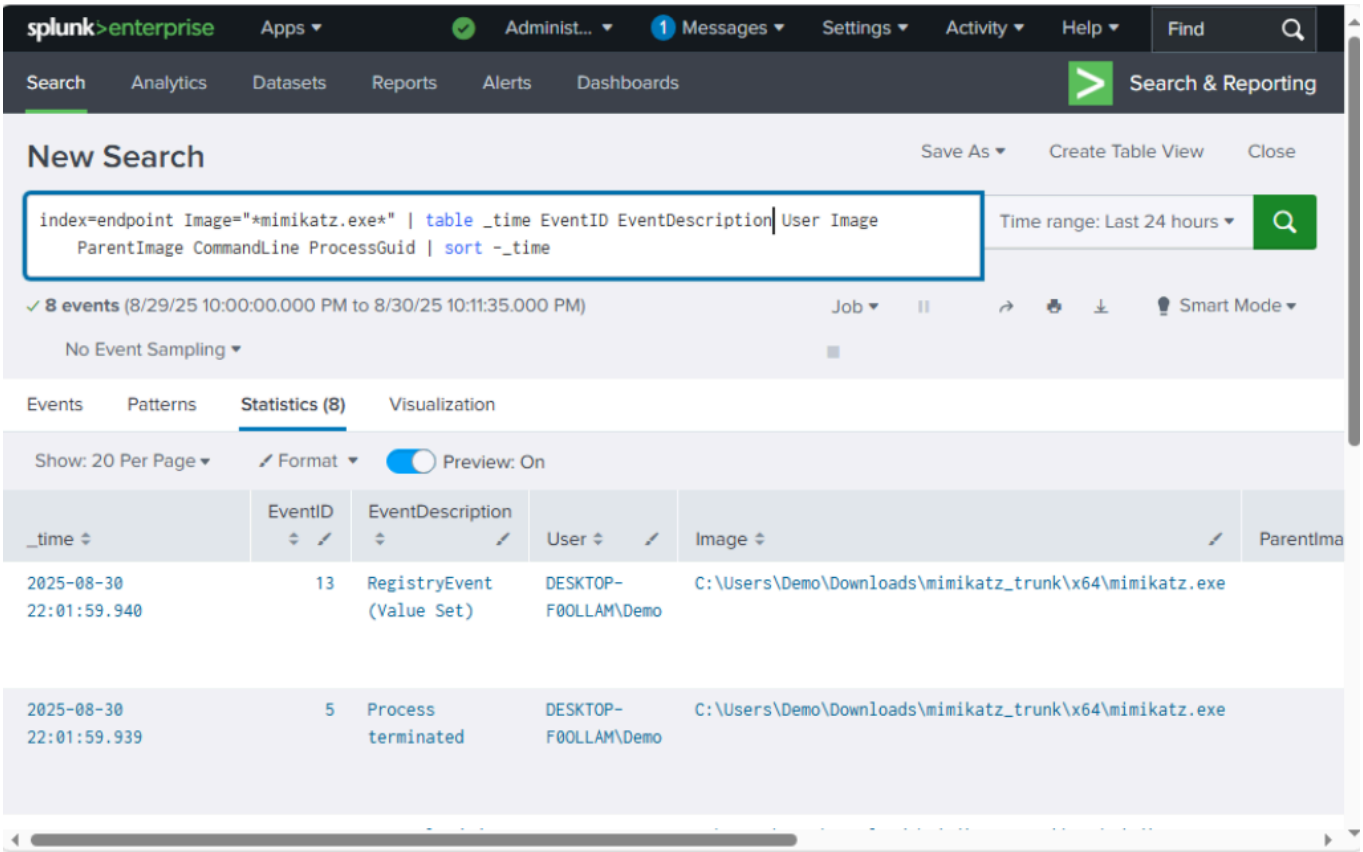
We then run the following command, which extracts all credentials currently in LSASS memory:



We now exit Mimikatz and head on to Splunk where we will review if this activity was recorded. As we can see, Mimikatz activity was in fact logged onto our Splunk:



Now we can visualize all Mimikatz related events with this table, which will tell us which process launched Mimikatz.exe, and the different events related to it:



Now, we can analyze each event and see its details. And we have finalized testing different malware files and programs and analyzing them in Splunk.