Level 2.2 - Detect a TCP scan (service enumeration)

Attackers can use a TCP scan / service enumeration to discover open ports on a server.

Password

`muzlppvv5q`

Instructions

1. Use your generic `Threshold` class to detect TCP scans
2. Create a new `Threshold` object in the main module, this time with different `group_key` and `unique_key` arguments
3. Think how to set these to correctly detect a TCP scan
4. Feed this object all TCP SYN packets and alert when a threshold is exceeded
5. Adjust the threshold parameters to catch the malicious trigger and not alert the non-malicious trigger

Notes

GUIDING QUESTIONS

Use the guiding questions below to identify the malicious trigger and structure your code's detection:

- What do I need to extract from the TCP packet?
    - Are there specific flags I should look for?
    - How do I group the relevant packets in a window, e.g. IP source, IP destination, etc?

- How can I implement the sliding window algorithm?
    - Can I reuse any existing implementations?
    - What do I need to keep track of and log inside my window?
    - What data structures (e.g. lists, dictionaries, classes) can I use to implement my window so that it is easy to read and use?
    - When should I check if the window has exceeded?
    - When should I check if the logged packets are outdated?
- What should the values of my `count` and `window` be?

To submit

Submit a ZIP `nids.zip` containing all `.py` files.