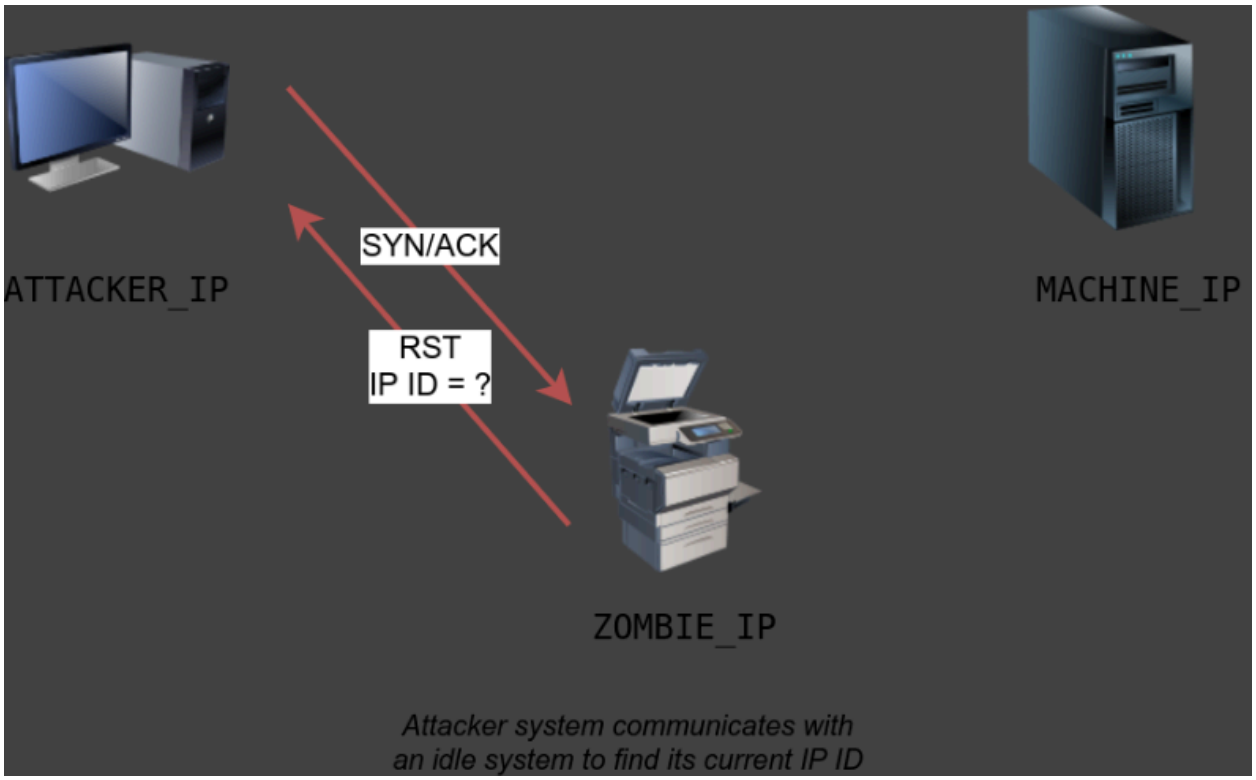
Nmap Idle/Zombie Scan

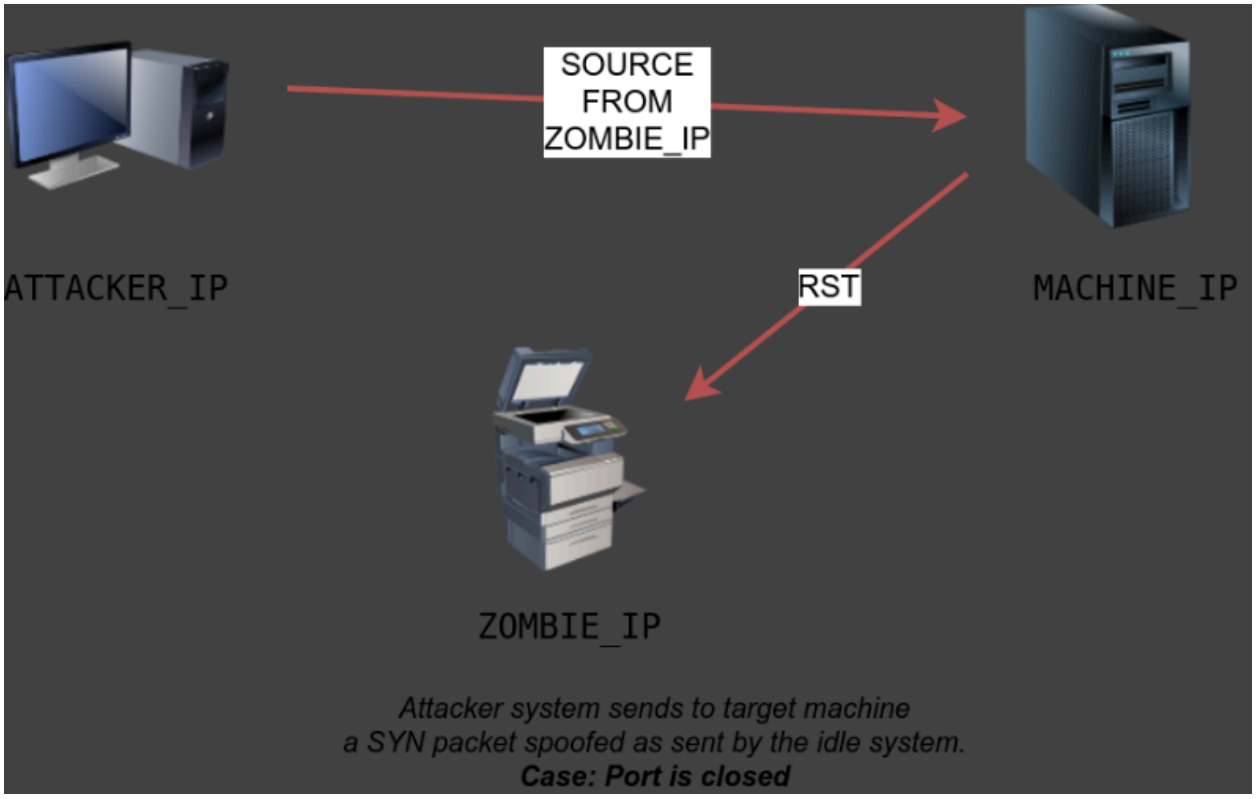
Spoofing the source IP address can be a great approach to scanning stealthy, but spoofily will ONLY work in specific network setups. Requires the attacker to be in a position where they can monitor traffic. Considering these limitations, spoofing your IP address can have little use, however using an Idle scan can build on these limitations.

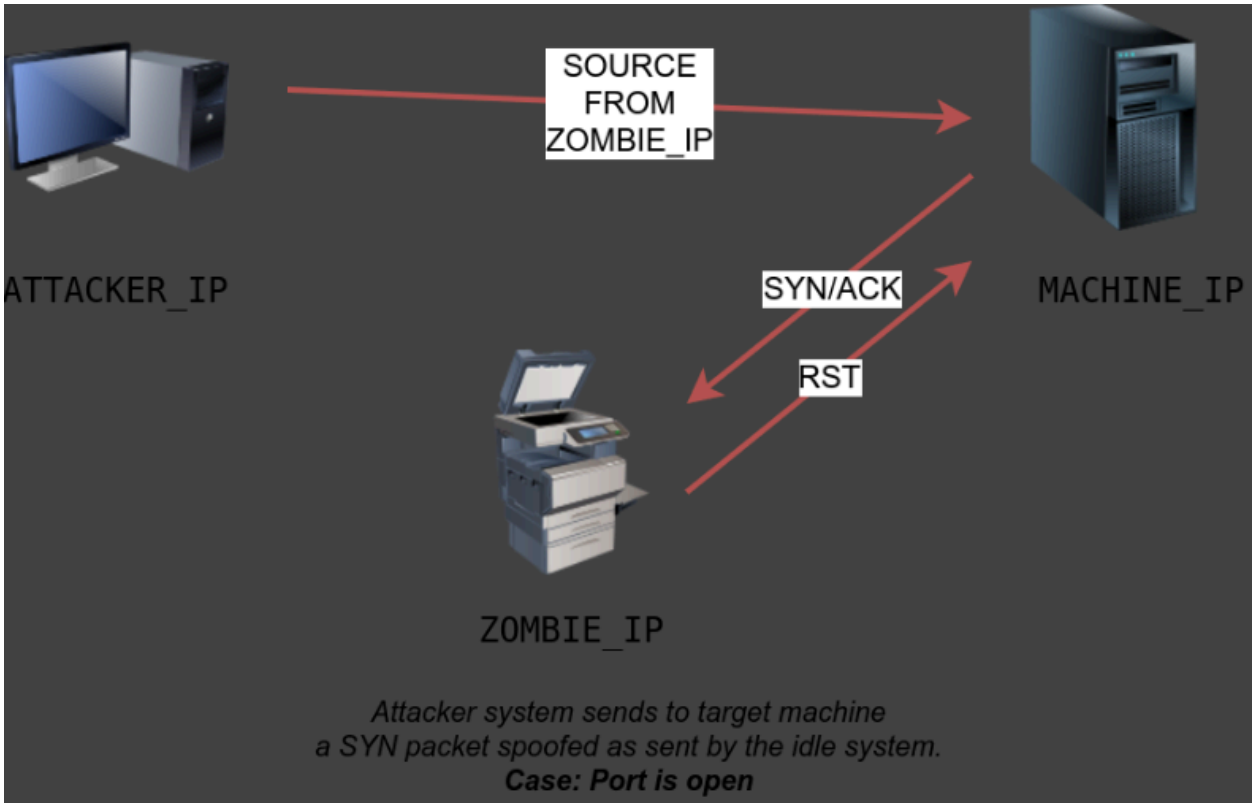
The Idle scan **also known as the zombie scan** requires an idle system connected to the network that you can communicate with. Practically Nmap will make each probe appear as if it is coming from the Zombie host, then it will check for indicators whether the zombie is receiving any response to the spoofed probe. This can be accomplished by checking the IP ID value in the IP header.
Run idle scan using: **nmap -sI ZOMBIE_IP 10.10.149.1** where zombie_ip is the IP of the idle host.

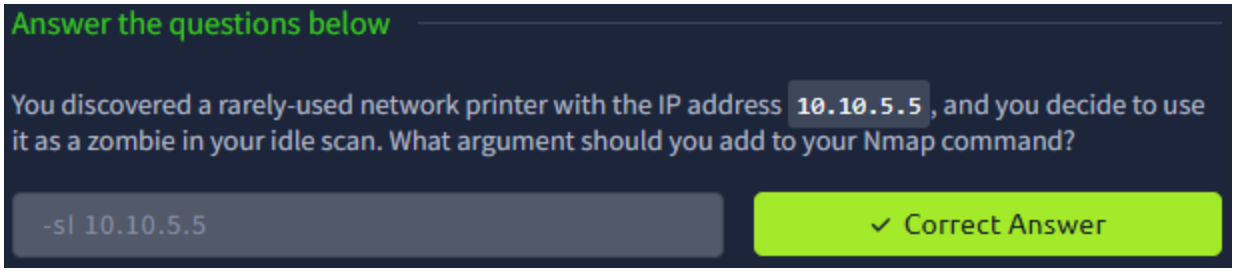
3 Steps:

1. **Trigger idle host to respond so that you can record the current IP ID of idle host.**
2. **Send SYN packet to TCP port on the target, packet should be spoofed to appear as if it was coming from the idle host (zombie) IP address.**
3. **Trigger idle machine again to respond so that you can compare the new IP ID with the older one received earlier.**




SOURCE
FROM
ZOMBIE_IP
ATTACKER_IP
RST
MACHINE_IP
ZOMBIE_IP
Attacker system sends to target machine
a SYN packet spoofed as sent by the idle system.
Case: Port is closed


SOURCE
FROM
ZOMBIE_IP
ATTACKER_IP
SYN/ACK
RST
MACHINE_IP
ZOMBIE_IP
Attacker system sends to target machine
a SYN packet spoofed as sent by the idle system.
Case: Port is open



Conclusion:
Zombie IPs are an excellent way at being quiet or not sending any traffic of its own such as a printer. Has a predictable IP ID sequence and doesn't respond aggressively or suspiciously to unusual traffic. So for stealth attacks it is an excellent way of doing it as well as making it difficult to trace back to the original attacker.
What I could do is a MITM attack on the printer to intercept the traffic sent between the destination and the printer for instance, so I can see the traffic. Intercepting SYN/ACK or similar traffic to learn about the target port state.