W10 — Reflect: Standard ACLs & Wildcard Masks

Week: W10
Date: 2025-11-09
Student: Allison "AJ" Buffam
Ops-only policy: Use operational commands (device prompt + command). No `show running-config`. Packet Tracer friendly outputs.

Reflect Questions

1. Why does the “Protect PC Network” ACL inspect **source** instead of **destination** addresses?

The ACL is basically a gatekeeper, super good at checking who's trying to send traffic the way of the router. It doesn't really care where it's headed, it just wants to make sure that the only trusted sources are allowed through. Kind of like a club bouncer, like this guy down below. If he sees someone he doesn't trust or recognize, he won't let them in and just blocks the door.

2. What would happen if the ACL were applied **outbound** on EDGE instead of **inbound** on CORE?

Relating back to the image above, we'll use the left side as the EDGE outbound ACL and the right side as the CORE inbound ACL.

Left side, EDGE outbound: dude is blocking the wrong side of the door, he only stops things leaving the building. Bad traffic already walked through the network, and now the guard is like "you're not allowed to leave" DESPITE already being let in. It's inefficient.

Right side, CORE inbound:

Dude will open the door for only the approved guests coming in, so no unwanted packets will even step foot inside the network (or club). It's smart protection.

3. Show two `show access-lists` lines proving both a permit and a deny match occurred.

From BUFF0039-CORE: `show ip access-lists PROTECT-PC`

```
Standard IP access list PROTECT-PC
    10 deny   198.18.116.128 0.0.0.63 log (10 matches)
    20 permit any (58 matches)
```

So, `10 deny` blocked 10 packets from my PC + Mgmt subnet `(198.18.116.128/26)` blocked it ten times, showing that the deny statement worked as intended. Then, my `20 permit` allowed 58 packets to flow through, showing that communication wasn't interrupted. Together, they demonstrated that the Protect-PC ACL filtered traffic successfully, allowing packets through the door but then blocking the restricted ones.

4. What part of ACL behaviour confirms the “first-match wins” rule?

From how I understand it, the "first-match wins" rule works like the bouncer reading the list from the top to the bottom. If he sees your name on the list, he stops reading (even if it says permit or deny). He doesn't need to keep reading, because he's already determined whether you're in or out. That's why the order matters so much in ACLs.

5. Describe one improvement you made to your ACL order or mask logic during testing.

Not so much of an ACL issue, but more of a "I forgot to disable the firewall and only realized after an hour like a clown 🤡". However, I realized while doing the quiz that I probably messed up with the addressing because you're supposed to start with the network and then account for how many addresses you need to block. Eventually after tweaking (and turning the firewall off), I was able to get the results I wanted, with the ACL working as how I believe you wanted it to.

Time & Confidence

- **Time spent (hh:mm):** 8:00
- **Confidence (0–5):** 3

I understand some of it, but I am unsure if I'm getting it right! :D

Appendix — Your best one-liner

```
BUFF0039-CORE# show access-lists PROTECT-PC | include matches
    10 deny   198.18.116.128 0.0.0.63 log (10 matches)
    20 permit any (58 matches)
```