# User Stories

## Playing with EVE for the first time to try it out

EVE-OS (edge node, no user interface) <==> EVE Controller (provides all operational configuration of EVE-OS nodes, communicated over TLS using open EVE API)

Option 1 - no edge hardware, just want to explore architecture of EVE-OS and open source EVE Controller (OpenEVC) – run both on personal computer (PC)

Option 2a - RPi4 running EVE-OS, PC running OpenEVC, same local area network

Option 2b - any x86 (amd64) hardware running EVE-OS, PC running OpenEVC, same local area network

Option 3 - Equinix Metal iPXE install of EVE-OS, ZEDCloud as commercial EVE Controller, connected over the Internet

Option 4 - two virtual machines running EVE-OS and OpenEVC on personal computer

## OpenEVC controller autodiscover via mdns

This scenario can cover all options except for Option 3 (obviously because it does not involve OpenEVC).

1. User launches controller, which registers itself in mdns as **openevc.local.**
2. Default EVE build shall have **eve_install_server=openevc.local**. As long as EVE and Controller have access to the same mdns daemon (e.g. Avahi), Eve instance will be able too resolve that address
3. Whenever new node knocks controller, corresponding message is printed in the logs with the hint on what to run to onboard it, or notification pops up in the web-ui (if we ever decided to implement one for the OpenECV)
4. On the controller side user can list all the not-onboarded nodes detected on the network and issue command "**onboard 1**", where 1 is sequtial number of the node in the mentioned list.
5. Node shall be onboarded using serial number. For simplicity, serial number can be just assigned from the controller side. This way it will be possible to onboard multiple instances into the same controller