Text

Description automatically generated

*#!/usr/bin/env python*

*from* mininet.net *import* Mininet

*from* mininet.node *import* Controller, RemoteController, OVSController

*from* mininet.node *import* CPULimitedHost, Host, Node

*from* mininet.node *import* OVSKernelSwitch, UserSwitch

*from* mininet.node *import* IVSSwitch

*from* mininet.cli *import* CLI

*from* mininet.log *import* setLogLevel, info

*from* mininet.link *import* TCLink, Intf

*from* subprocess *import* call

def myNetwork():

    net = Mininet( *topo*=None,

*build*=False,

*ipBase*='10.0.0.0/8')

    info( '\*\*\* Adding controller\n' )

    c0=net.addController(*name*='c0',

*controller*=RemoteController,

*ip*='192.168.10.200',

*protocol*='tcp',

*port*=6633)

    net = Mininet( *topo*=None,

*build*=False,

*ipBase*='10.0.0.0/8')

    info( '\*\*\* Adding controller\n' )

    c0=net.addController(*name*='c0',

*controller*=RemoteController,

*ip*='192.168.10.200',

*protocol*='tcp',

*port*=6633)

    info( '\*\*\* Add switches\n')

    s1 = net.addSwitch('s1', *cls*=OVSKernelSwitch)

    info( '\*\*\* Add hosts\n')

    h1 = net.addHost('h1', *cls*=Host, *ip*='10.0.0.1', *defaultRoute*=None)

    h3 = net.addHost('h3', *cls*=Host, *ip*='10.0.0.3', *defaultRoute*=None)

    h2 = net.addHost('h2', *cls*=Host, *ip*='10.0.0.2', *defaultRoute*=None)

    h4 = net.addHost('h4', *cls*=Host, *ip*='10.0.0.4', *defaultRoute*=None)

    info( '\*\*\* Add links\n')

    net.addLink(h4, s1)

    net.addLink(h3, s1)

    net.addLink(h2, s1)

    net.addLink(h1, s1)

    info( '\*\*\* Starting network\n')

    net.build()

    info( '\*\*\* Starting controllers\n')

*for* controller *in* net.controllers:

        controller.start()

    info( '\*\*\* Starting switches\n')

    net.get('s1').start([c0])

    info( '\*\*\* Post configure switches and hosts\n')

    info( '\*\*\* Starting network\n')

    net.build()

    info( '\*\*\* Starting controllers\n')

*for* controller *in* net.controllers:

        controller.start()

    info( '\*\*\* Starting switches\n')

    net.get('s1').start([c0])

    info( '\*\*\* Post configure switches and hosts\n')

    CLI(net)

    net.stop()

*if* \_\_name\_\_ == '\_\_main\_\_':

    setLogLevel( 'info' )

    myNetwork()