```
netv.data(data)
                                                         const layout = new NetV.Layouts.RandomLayout(netv)
const simulation = d3.forceSimulation(data.nodes)
                                                         layout.time(1000)
    .force(
                                                         layout.onStart(() => {
        'link',
                                                             console.log('random layout start')
        d3.forceLink(data.links).id((d) => d.id)
                                                         })
                                                        layout.onTick(() => {
    .force(
                                                             console.log('iteration...')
        'collide',
        d3.forceCollide((d) => {
                                                        layout.onStop(() => {
            return 2.4 * radius(d)
                                                             console.log('random layout done')
        })
                                                        })
                                                         layout.start()
    .force('center', d3.forceCenter(width / 2, height
 / 2))
    .force('x', d3.forceX(width / 2).strength(0.13))
    .force('y', d3.forceY(height / 2).strength(0.13))
simulation.on('tick', () => {
    data.nodes.forEach((n) => {
        const node = netv.getNodeById(n.id)
        node.x(n.x)
        node.y(n.y)
    })
    netv.draw()
                                                                     (b)
                       (a)
```