* > admin.nodeInfo.enode
* >admin.addPeer()
* >net.peerCount
* [::] will be parsed as localhost (127.0.0.1)
* > net.listening

### Setup bootnode

The first time a node connects to the network it uses one of the predefined [bootnodes](https://github.com/ethereum/go-ethereum/blob/master/params/bootnodes.go). Through these bootnodes a node can join the network and find other nodes. In the case of a private cluster these predefined bootnodes are not of much use. Therefore go-ethereum offers a bootnode implementation that can be configured and run in your private network.

It can be run through the command.

> bootnodeFatal: Use -nodekey or -nodekeyhex to specify a private key

As can be seen the bootnode asks for a key. Each ethereum node, including a bootnode is identified by an enode identifier. These identifiers are derived from a key. Therefore you will need to give the bootnode such key. Since we currently don't have one we can instruct the bootnode to generate a key (and store it in a file) before it starts.

On command line not on console

> bootnode -genkey bootnode.key

I0216 09:53:08.076155 p2p/discover/udp.go:227] Listening, enode://890b6b5367ef6072455fedbd7a24ebac239d442b18c5ab9d26f58a349dad35ee5783a0dd543e4f454fed22db9772efe28a3ed6f21e75674ef6203e47803da682@[::]:30301

(exit with CTRL-C)

The stored key can be seen with on command line:

> cat bootnode.keye6eaa407c614eccdc33d938d9a05307442b7fdf15b2f5748ab1f19ea507002c9

To instruct geth nodes to use our own bootnode(s) use the

--bootnodes flag. This is a comma separated list of bootnode enode identifiers.

geth --bootnodes "enode://8688b8b59cf0ea39cb6864967b8188a8071ebeff0008447064e497d25aedf1a5a43462ad1a9e255f000b434219fe75c138a3ae43a9a5507b95d34bcfb7ca2452@69.12.23.105:30301"

To get the bootnode:

bootnode -nodekey bootnode.key

INFO [08-28|12:47:33] UDP listener up self=enode://8688b8b59cf0ea39cb6864967b8188a8071ebeff0008447064e497d25aedf1a5a43462ad1a9e255f000b434219fe75c138a3ae43a9a5507b95d34bcfb7ca2452@[::]:30301

To monitor node on test net: https://github.com/ethereum/wiki/wiki/Network-Status