**How to deploy blockchain in a secure way - Sentry Node with Kubernetes.**

Today blockchain industry is gigantic and a lot of companies are adopting the blockchain to make one's way in decentralizing fashion and to secure the data. Our team were also keen to take a leap in the blockchain race. We built the first govern model with KYC enable blockchain called MAXONROW. We are happy to announce that we have successfully built and launch [Maxonrow](https://github.com/maxonrow/maxonrow-go" \t "_blank) main-net.

When our team started to search for blockchain-related information, we used to get a lot of blockchain-related data/information through sources eg(Medium, Reddit, Blockchain articles), were all data which we found was related to blockchain setup or some of the tools used for blockchain and how to run the blockchain. But when it comes deployment of blockchain in a secure way or as a security concern we didn’t find much of resources. The data, which we use to found, was related to running blockchain in a local environment or a bit of explanation on how to secure blockchain that’s why we created this article to share these piece of information how [Maxonrow](https://github.com/maxonrow/maxonrow-go" \t "_blank) was deployed in a secure way.

When it comes to deployment, the main concern is about security. How we can secure blockchain and especially validator node. Validators play an important key role in maxonrow blockchain; maxonrow blockchain adopts POS consensus protocol. What is the difference between bitcoin, ethereum and our blockchain consensus protocol is

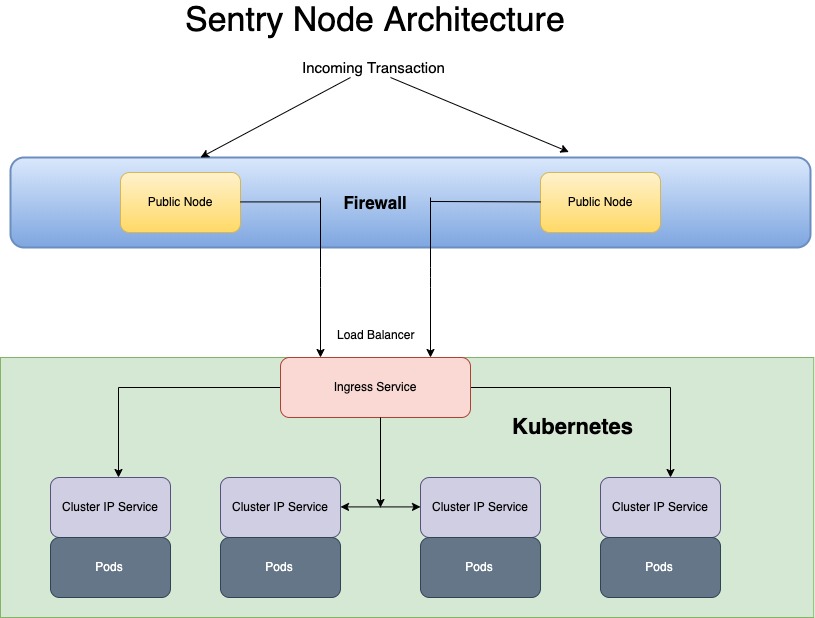
PoW (Proof of work) is adopted by Bitcoin, Ethereum, etc. PoW selects one node to create a new block in each round of consensus by computational power competition. The node who first addresses the puzzle can have a right to create a new block called miners. For more info [proof-of-work-explained](https://cointelegraph.com/explained/proof-of-work-explained" \t "_blank)

In POS (Proof of stake), selecting each round of node that creates a new block depends on the held stake rather than the computational power

The different from PoW is that nodes do not need to adjust nonce for many times, instead, the key to solve this puzzle is the amount of stake (coins). Hence, PoS is an energy-saving consensus protocol. For more info [Proof-of-stake-explained](https://www.binance.vision/blockchain/proof-of-stake-explained)

We found one of the solutions to deploying blockchain was sentry node architecture.

Below Image display deploymet design



What is Sentry Node Architecture?

Sentry nodes are Full Nodes, so nodes that store the whole blockchain. Sentry Nodes are used to isolate your validator from the public. Your validator node only establishes private connections to your sentry nodes and they connect to the rest of the network.

By doing this, Sentry Nodes protect your validator from being attacked. One of the most common attack vectors is DDOS. Sentry Nodes can mitigate those attacks. This is especially important since a DDOS attack will prevent a validator node from communicating with the rest of the network. This leads to downtime and slashing. Therefore, it is a must-have for validators to secure their setup with sentry nodes. For more info [Sentry-node-architecture-overview](https://forum.cosmos.network/t/sentry-node-architecture-overview/454" \t "_blank)

Assuming by now your familiar with sentry node design lets move to our deployment process. We use kubernetes to create our blockchain containerized instance,We follow Container deployment era through which we control the incoming traffic .

Assuming by now you're familiar with sentry node design lets move to our deployment process. We use Kubernetes to create our blockchain-containerized instance; we follow the Container deployment era through which we control the incoming traffic.

Kubernetes is a portable, extensible, open-source platform for managing containerized workloads and services that facilitates both declarative configuration and automation. It has a large, rapidly growing ecosystem. Kubernetes services, support, and tools are widely available read more [Kubernetes](https://kubernetes.io/docs/concepts/overview/what-is-kubernetes/" \t "_blank)

We use pods. A Pod is a group of one or more containers (such as Docker containers), with shared storage/network, and a specification for how to run the containers. Inside pods, which create the container, where an instance of maxonrow blockchain will be running and its full node.

This pods which will interact with the second layer of the firewall, which contain public sentry node. The entire public node connected and running securely. We are using POS consensus engine. As you can see that public sentry node design is more secure in comparison to the traditional way of implementation. We as a maxonrow team believe public sentry node approach will bring the extra layer of security to our blockchain by which maxonrow blockchain is running safely.

If you like this article please hit the clap button on the right side to support us. If you have any query related to pulic sentry node design or on maxonrow blockchain please comment blow.

!!!! Stay tuned to maxonrow channel we have more to share...

To know more about maxonrow roadmap please visit the website

[maxonrow](https://www.maxonrow.com/)

To know more about maxonrow blockchain please find the whitepaper in below given link [WHITE\_PAPER\_MAXONROW](http://mxw.brboss-demo.com/wp-content/uploads/2019/06/WHITE_PAPER_MAXONROW_ENG_New.pdf)

To run our maxonrow blockchain please find the below github.com link [maxonrow-go](https://github.com/maxonrow/maxonrow-go/tree/develop) for setup please go through readme document.

To intracting with maxonrow blockchain we have built the tools for setup please go through readme document

<https://github.com/maxonrow/mxw-sdk-js>

If you are still stuck on any points, take a look through my [repository](https://github.com/nagarajmanjunath/maxonrow-go)

 to see if you can find a solution there, or get in contact with me [@linkedin](https://www.linkedin.com/in/nagaraj-m-25264b10a/)