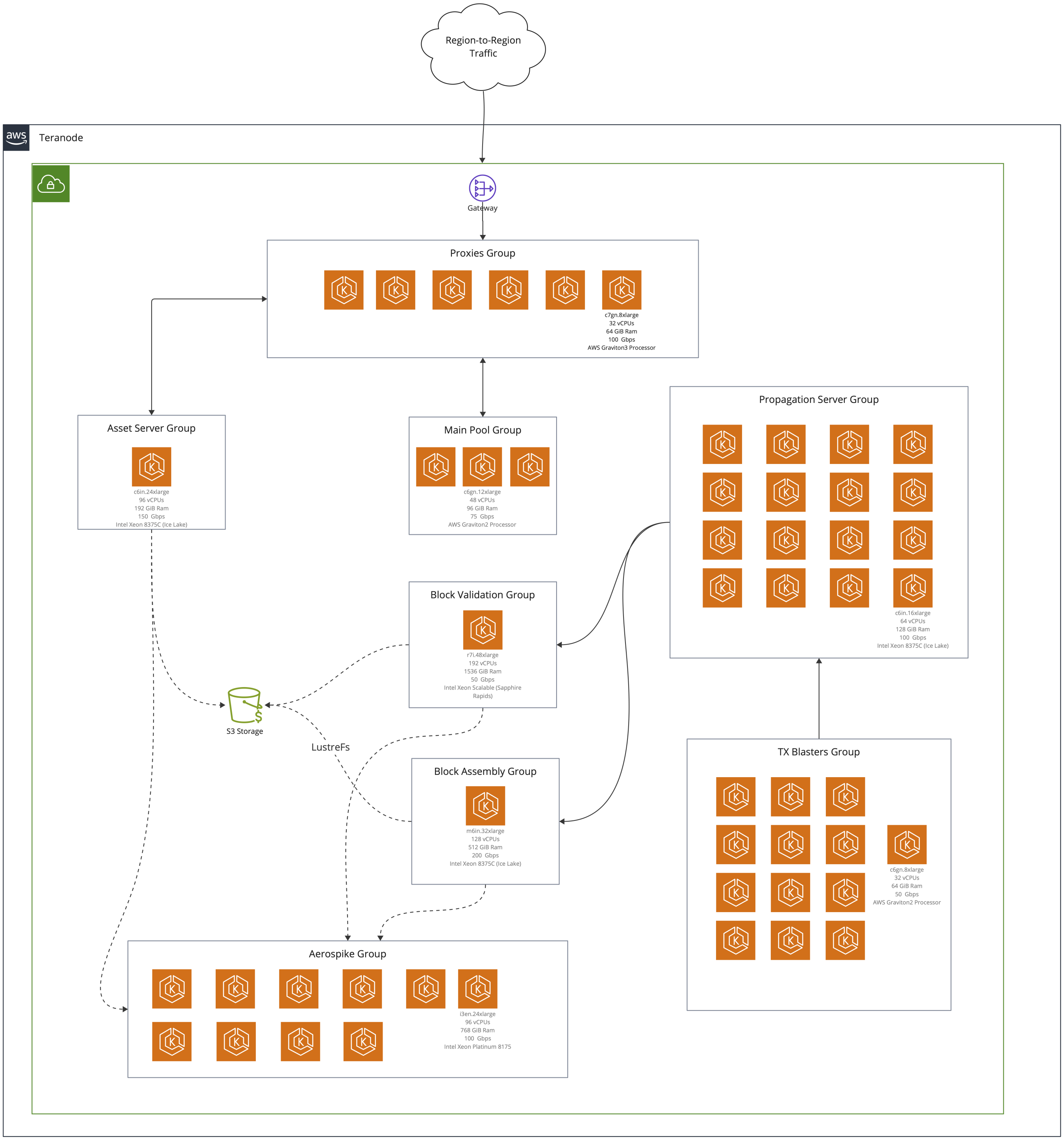
Teranode - POC Environment

1. Teranode AWS Setup



On February 2024, the BSV team started a multi-month test exercise (the proof-of-concepts or POC). The objective of the exercise is to prove that Teranode BSV can scale to process millions of transactions per second.

For the purpose of this test, Amazon AWS was chosen as the cloud server provider. Three (3) nodes were created in 3 separate AWS regions. Each node is composed of a number of server instances. The purpose of this document is to describe the setup of each node.

2. Instance Details

The table below provides a summary of the server groups, the number of instances per group, and their technical specification.

Instance Group	Instances	Instance Type	vCPUs	Memory (GiB)	Network (Gbps)	Processor
Asset-sm	1	c6in.24xlarge	96	192	150	Intel Xeon 8375C (Ice Lake)
Block Assembly	1	m6in.32xlarge	128	512	200	Intel Xeon 8375C (Ice Lake)
Block Validation	1	r7i.48xlarge	192	1536	50	Intel Xeon Scalable (Sapphire Rapids)
Main Pool	3	c6gn.12xlarge	48	96	75	AWS Graviton2 Processor
Propagation Servers	16	c6in.16xlarge	64	128	100	Intel Xeon 8375C (Ice Lake)
Proxies	6	c7gn.8xlarge	32	64	100	AWS Graviton3 Processor
TX Blasters	13	c6gn.8xlarge	32	64	50	AWS Graviton2 Processor
Aerospike	10	i3en.24xlarge	96	768	100	Intel Xeon Platinum 8175

AWS EKS (Kubernetes) is used to run services in all instances. Aerospike has been run in and out of Kubernetes as part of the test exercise.

Please note that exact instance count under the Main Pool, Propagation Servers, Proxies, TX Blaster and Aerospike groups can vary during the test.

3. Instance Group Definitions

Here's a structured table summarizing the instance groups, their descriptions, and purposes based on the provided details:

Instance Group	Description	Purpose
Proxies	Ingress Proxy Servers	Managing region-to-region ingress traffic with Traefik k8s services.
Asset-sm	Asset (Small Instance)	Hosting the Asset Server Teranode microservice.
Block Assembly	Block Assembly	Running the Block Assembly Teranode microservice.
Block Validation	Block Validation	Running the Block Validation Teranode microservice.
Main Pool	Overlay Services	Running the p2p, coinbase, blockchain, postgres, faucet, and other microservices.
Propagation Servers	Propagation Servers	A pool of Propagation Teranode microservices.
TX Blasters	TX Blasters	A pool of TX Blaster Teranode microservices.
Aerospike	Aerospike	A cluster of Aerospike NoSQL DB services.

4. Storage

- **AWS S3 -** Configured to work with Lustre filesystem, shared across services, and used for Subtrees and TX data storage.
- **Aerospike -** Used for UTXO and TX Meta data storage.
- **Postgresql -** Within the Main Pool instance group, an instance of Postgres is run. The Blockchain service stores the blockchain data in postgres.

5. Performance Metrics

In the context of our AWS-based infrastructure, the primary consideration for specifying instance types has been to meet the high network bandwidth requirements essential for the Teranode services. This approach was adopted following the identification of network bandwidth as a critical bottleneck in previous discovery phases.

However, the services are not fully utilizing the allocated resources in terms of CPU utilization, memory utilization, and disk I/O. There is excess capacity in these areas which may present future opportunities for cost optimization or reallocation of resources to better match actual usage patterns.