Fleta Mainnet Performance Test Report

Innovations for a Better Life

Fleta.io

Background

1 Maintaining consistent transaction speed and main chain scalability

It is very important to test the limits on how stable and consistent the chain can perform.

Simply boosting transaction speed does not guarantee its service capability under actual operating environment. To provide actual services, consistent processing performance of the chain must be tested.

There are many factors involved in process of generating, propagating and finalizing Tx. To verify the pure processing speed of the chain, FLETA came up with a method for the measurement.

2 Chain performance and security in practical operating environment

Many projects conduct internal laboratory test of the chain, but internal laboratory testing alone doesn't represent the actual chain performance.

Especially, since the chain is developed to be used for actual services, the test results in actual operating environment are important.

FLETA has tested and implemented the actual operating environment as much as possible to produce the most objective measurement.

Test Environment

FLETA is designed to well-operate even the nodes are spread all over the world. Therefore, FLETA tested after setting up cloud servers in 6 different regions.

1 Server Specifications

Vultr High Frequency 8 Core (3.8 GHz) / 32 GB RAM / 512 GB NVMe

2 Server Network Installation

All settings of the test server are configured in the same environment as the mainnet. All observer nodes use TCP to communicate within themselves. Formulators (Block Generating Node) also use TCP to communicate within themselves. When Formulators and Observer nodes communicate each other, they communicate through Websocket. Same as mainnet configuration, FLETA installed the cloud VM immediately after creating it, and did not proceed with tuning the network of the VM through Sysctl and used the default value provided in the cloud.

3 Server Distribution – Consistency Test

Region	Server Configuration
US, New Jersey	5 Observer Nodes / 3 Formulators
US, Miami	3 Formulators
US, Dallas	3 Formulators
DE, Frankfurt	3 Formulators
FR, Paris	3 Formulators
UK, London	3 Formulators

4 Server Distribution – Actual Operating Test

Region	Server Configuration
US, New Jersey	5 Observer Nodes / 2 Formulators / 1 Tx Generator
US, Miami	2 Formulators / 1 Tx Generator
US, Dallas	2 Formulators / 1 Tx Generator
DE, Frankfurt	2 Formulators / 1 Tx Generator
FR, Paris	2 Formulators / 1 Tx Generator
UK, London	2 Formulators / 1 Tx Generator

Measurement Scenario

1 General Scenario

The codes used for performance test is identical to the mainnet except for the chain defaults, including the Genesis configuration.

In the test,

1. Validating signature and transaction in the transaction propagation phase
2. Validating transaction and reflecting into the Context in block generation phase
3. Observer Nodes validating signature in block validation phase
4. Validating Generator's signature
5. Validating Generation order
6. Validating transaction signature
7. Validating transaction
8. Reflecting into Context
9. Validating by byte-level through comparing Context Hash values for changes
10. Storing chain data locally with fsync=on status and proceeding every process

These factors mean that there are no tampering or changes on this measurement, and the same cloud server, network configuration, and testing environment will be provided as same as the result of the report.

2 Consistency Measurement – Loading and Transaction Speed

Consistency measurement for loading and transaction speed is to test the consistent maximum transaction processing speed of the chain.

As a result, the signed transaction generates the block by including the number of transactions in the block directly during the block generation phase. This allows us to purely embed a transaction into a block, without the transaction being transferred to the block generator, the Formulator to measure the maximum performance for propagation and consensus.

3 Measurement of chain performance with real operating environment

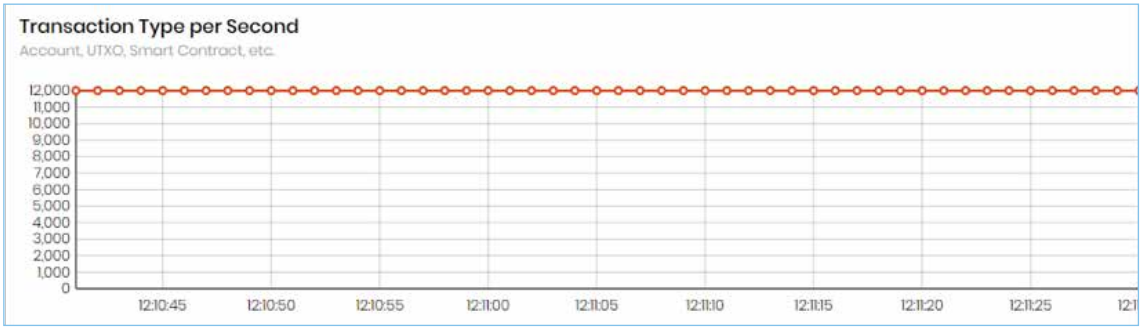
In the practical operating environment, we generate transactions and broadcast the transactions to all nodes. This is the same environment as the chain's actual operation environment.
Normal nodes, not the Formulators, propagates the actual transactions, and the received node validates it, saves it in the transaction pool, and propagates it again.

Test Result

12,000TPS

1 Consistency Measurement – Loading and Transaction Speed

When testing the performance of the chain, we ran Block Explorer, which operates as a normal node, to obtain transaction information and achieved chain performance graph. We checked the information of 600 generated blocks at once, and averaged generation interval. As a result, we were able to measure the average time of generating blocks and the average number of processed transactions.



This transaction graph combines two blocks into one point and records one point per second. You can see that 12,000 TPS is consistent in the graph.

Section (Height)	Number of Blocks	Number of Transactions	Start Time	End Time	Time Consumed	TPS (Average)
1~600	600	3,600,000	12:07:21	12:12:36	315sec	11,428
601~1200	600	3,600,000	12:12:36	12:12:36	300sec	12,000
1201~1800	600	3,600,000	12:17:36	12:22:36	300sec	12,000
1801~2400	600	3,600,000	12:22:36	12:22:36	300sec	12,000
2401~3000	600	3,600,000	12:27:36	12:27:36	300sec	12,000

This table shows transaction process by sections. There was a slight delay because the start speed of the Formulator was slightly different in the first section, but from the second section, the blocks were synchronized, and blocks have been generated stably. As a result, no block delay has occurred since the second section, and transactions have been processed stably.

Test Result

12,000TPS

Measurement	Detailed Information	Result (Average)
Block Packet Decompression Time	Decompress gzip compressed packets	20ms
Block Deserialization Time	Convert Block Packets to Block Messages	20ms
Transaction Signature Validation Time	Validate all transaction signatures in blocks	250ms
Context Generation Time	Validate and process transactions	70ms
Block Disk Save Time	Save blocks to Disk (fsync=on)	60ms
Block Validation Consensus Time	Observer Nodes validate and consent blocks	30ms

This table is a result of measurement that has great effect on the performance of the chain. The measured results show that the average time needed to process a block is 420ms from block packet decompression to block disk storage time, and 450ms including the validation process by Observer Nodes. Therefore, 2 blocks can be processed per second since it takes less than 500ms per block. FLETA has developed this structure to minimize the impact on network latency when processing blocks. Therefore, even if there is a network latency of 100ms per second, it can be handled seamlessly since only 900ms is required to process 2 blocks.

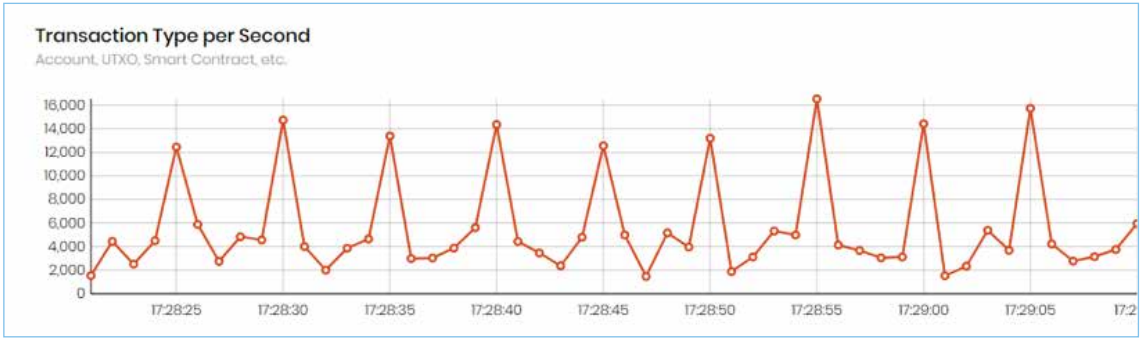
Test Result

Maximum:
12,000 TPS

Average:
7,000 TPS

2 Measurement of Chain Performance with real operating environment

When testing the performance of the chain, we ran Block Explorer, which operates as a normal node, to obtain transaction information and achieved chain performance graph. We recorded the block delay and information of the 600 generated blocks at once, and average TPS (Total transactions / generated blocks * delay).



This transaction graph combines two blocks into one point and records one point per second. In this graph, you can see that the number of transactions differs from block to block. This is because the transactions are propagated through the nodes and included in the Formulator, so each block generation contains different transactions.

Section (Height)	Number of Transactions	Start Time	End Time	Time Consumed	Maximum TPS	Average TPS
1~600	2,131,248	16:29:42	16:34:51	309sec	12,428	6,897
601~1200	2,161,217	16:34:51	16:39:52	301sec	13,356	7,180
1201~1800	2,177,712	16:39:52	16:44:52	300sec	14,002	7,259
1801~2400	2,121,227	16:44:52	16:49:52	300sec	12,367	7,070
2401~3000	2,144,324	16:49:52	16:54:53	301sec	12,951	7,124

There was a slight delay in the first section because the start process of Formulators differs, but from the second section, the blocks were synchronized, and blocks have been generated stably. There was a slight delay since CPU usage was distributed when the Formulators receive transactions, validate them, and store them into the transaction pool. (which is different from putting stable transactions)

Conclusion

Improving transaction speed is very important part of the development of blockchain ecosystem. FLETA aims to provide blockchain technology that can be utilized and used for real-world services.

FLETA has confirmed its maximum speed of 14,000TPS on the chain, and has developed core technology that presents more than 12,000TPS allowing continuous Tx process.

In addition, these tests were not conducted in FLETA's internal development lab, they were conducted in a large number of external local networks after setting up Formulator and environment. This means that the test results are very objective and will not differ much from the chain performance in a real operating environment.

FLETA will continue to develop its own blockchain technology and will put most efforts to put ourselves as the biggest competitor for the best technology.

Thank You.

SEND SQUARE

Fleta

Innovations for a Better Life Fleta.io