Gas/Fee usage

Transaction priority and fee market

Secret network uses Tendermint as a consensus engine combined with the Cosmos SDK, as such it has a relatively high throughput and the possibility for easy horizontal scalability. Because of this, there isno fee market on Secret Network, which means there is no fee-based prioritization built into the standard tendermint binary required to run Secret. Using a higher transaction fee will only increase the number of block signers (validators) that are willing to process your transaction. This might speed up finality but does not prioritize your transaction in the cue.

Gas and the Gas fee

The fee a user pays for a transaction is based on two things: the computational resources/block space that is used and the fee that validators ask for these resources. These two elements are calledgas and thegas fee .

Thegas required for the computation is determined by the smart contract engine. The current gas estimates per type of contract interaction are listedhere (for SDK components) andhere (for enclave components). More complex transactions will consume moregas.

Thegas fee is determined by validators themselves as a parameter in their node service. Themin_gas_fee parameter allows validators to set a minimum fee denominated inuSCRT . If a transaction has a lower allocated fee, they will not add it to blocks they propose. Validators propose blocks based on random selection following their Voting power. Themin_gas_fee for each block is dependent on the proposing validator. If blocks become too full or an attacker spams the network, the validators can increase their fee.

Secret validators run three tiers of fee's; this choice is up to the validator. Overall, we can assume more than 40% of validators support the "low" fee option, meaning pushing a TX with the low fee often gets processed within a few blocks. Pushing a TX with the "High" fee will increase the chance that the next block can fit your transaction. These TXs are therefore considered "faster". The current recommended fee options are listed in theCosmos chain registry.

Fee calculation + execution time

Now let's assume we execute a swap for a single Dex pool. We can expect this to take roughly 150000Gas . Executing this at the "low" fee option, we will pay0.0125 uSCRT for eachGas unit used. The total fee for this transaction then becomes:150000 * 0.0125 = 1.875 uSCRT = 1.875/1000000 = 0.001875 SCRT . At 40% voting power supporting amin_gas_fee of0.0125 uSCRT we can expect this transaction to be finalized after 4 blocks (4*6s = 24s) with a 97.5% certainty [1- (40%*4) = 0,9744].

FeeGrant

Secret also allows for Gas abstraction for users by leveraging the CosmosSDKFeeGrant module . This module allows one to submit transactions where a different wallet is paying the gas fees as long as they granted you a budget to do that.

Documentation to create Feegrant functionality in your UI arehere .

This tool is widely used in different Secret UIs (for exSecret dashboard) and there is a community run FeeGrant faucet available for dApps to use. -Faucet -Code

Last updated3 months ago On this page *Transaction priority and fee market * Gas and the Gas fee * Fee calculation + execution time * FeeGrant

Was this helpful? Edit on GitHub Export as PDF