

<p>Default start <code>geth --http --dev</code></p> <p>Send transaction</p> <pre> "jsonrpc":"2.0", "method":"eth_sendTransaction", "params":[{" "from": "0x1f2939b7eb62cc5c2c317017eacaedcdd3cc81a1", "to": "0xd46e8dd67c5d32be8058bb8eb970870f07244567", "gas": "0x76c0", "gasPrice": "0x9184e72a000", "value": "0x9184e72a", "data": "0xd46e8dd67c5d32be8d46e8dd67c5d32be8058bb8eb970870f072445675058bb8eb970870f072445675" }], "id":1 } </pre> <p>Make request</p> <pre> { "jsonrpc":"2.0", "method":"eth_gasPrice", "params":[], "id":73 } { "jsonrpc": "2.0", "id": 73, "result": "0x746a528800" } </pre>	<p>Must return price per gas in wei as type quantity</p> <p>If there is not enough blocks</p> <p>Must return default gas price</p>
<p>No transactions on blockchain</p> <pre> { "jsonrpc":"2.0", "method":"eth_gasPrice", "params":[], "id":73 } { "jsonrpc": "2.0", "id": 73, "result": "0x1" } </pre> <p>Returns 1 no matter the <code>-gpo.maxprice</code></p>	<p>Must return 1 for gasPrice if the initialized client has no blocks</p>
<p><code>geth --http --dev --gpo.maxprice 5000000000000</code></p> <p>Send transaction first</p> <pre> "jsonrpc":"2.0", "method":"eth_sendTransaction", "params":[{" "from": "0x1f2939b7eb62cc5c2c317017eacaedcdd3cc81a1", "to": "0xd46e8dd67c5d32be8058bb8eb970870f07244567", "gas": "0x76c0", "gasPrice": "0x9184e72a000", "value": "0x9184e72a",b </pre>	<p>If transaction gasPrice is more than the max price Must return max price</p>

<pre> "data": "0xd46e8dd67c5d32be8d46e8dd67c5d32be8058bb8eb970870f07244567505 8bb8eb970870f072445675" }, "id":1 } </pre> <p>Then make request</p> <pre> { "jsonrpc": "2.0", "id": 73, "result": "0x746a528800" } </pre> <p>Returns the set value in hex</p>	<p>Uses max price for gas fee not the gasPrice</p>
<pre> geth --http --dev --gpo.maxprice 50000000000000000000 </pre> <p>Send transaction first</p> <pre> "jsonrpc": "2.0", "method": "eth_sendTransaction", "params": [{ "from": "0x1f2939b7eb62cc5c2c317017eacaedcdd3cc81a1", "to": "0xd46e8dd67c5d32be8058bb8eb970870f07244567", "gas": "0x76c0", "gasPrice": "0x9184e72a000", "value": "0x9184e72a", "data": "0xd46e8dd67c5d32be8d46e8dd67c5d32be8058bb8eb970870f072445675058bb8eb 970870f072445675" }], "id":1 } </pre> <p>Then make request</p> <pre> { "jsonrpc": "2.0", "id": 73, "result": "0x9184e72a000" } </pre> <p>Returns the same value as the send transactions, gasPrice</p>	<p>The gasPrice is</p>
<p>Max gas price range 9223372036854775807 wei or 7FFFFFFFFFFFFFFF hex</p>	<p>Max gas price must be less than or equal to 0x7FFFFFFFFFFFFFFF gasPrice must be less than or equal to 1 eth</p>
<p>Min?</p>	<p>GasPrice must be greater than 0</p>
<p>GasPrice returns the average gasPrice</p>	<p>no</p>
<pre> geth --http --dev --gpo.maxprice 0 </pre> <p>Send transaction</p> <pre> { </pre>	<p>Must not allow max gas price to be set to 0</p>

<pre> "jsonrpc": "2.0", "method": "eth_sendTransaction", "params": [{ "from": "0xb4369ac7c3647d4d46a9230d6ffa993855e8e6d", "to": "0xd46e8dd67c5d32be8058bb8eb970870f07244567", "gas": "0x76c0", "gasPrice": "0x9184e72a000", "value": "0x9184e72a", "data": "0xd46e8dd67c5d32be8d46e8dd67c5d32be8058bb8eb970870f072445675058bb8eb970870f072445675" }], "id": 1 } Call gasPrice { "jsonrpc": "2.0", "id": 73, "result": "0x746a528800" } Returns default value </pre>	
<pre> geth --http --dev --gpo.blocks 2 Send transaction with gasPrice of 1 Call GasPrice { "jsonrpc": "2.0", "id": 73, "result": "0x1" } Send transaction with gasPrice of 10 Call gasPrice { "jsonrpc": "2.0", "id": 73, "result": "0x1" } Send transaction with gas price of 10 Cal gasPrice Returns 0x10 </pre>	<p>Must use the gasPrice from the latest block - n blocks for the gasPrice</p>
<pre> geth --http --dev --gpo.blocks 2 Send transaction with 1 as gasPrice Send transaction with 100 as gasPrice X 3 Get gasPrice 0x100 Send transaction with 1 as gas price X 3 </pre>	<p>Gas price can be set to 1 initially as long as the gas price in each transactions stays 1 The GasPrice will stay 1</p> <p>Or in other words</p>

	The price per gas can be 1 if a client recievs transactions with only a gasPrice of 1