Relayer Calls

Relayer calls allow an end-user to sign a Ambient command using an off-chain EIP-712 signature. This signed command can then be executed inside an Ethereum transaction sent by a third-party relayer. The user can optionally "tip" the relayer to compensate for the gas cost.

```

Copy function userCmdRelayer ( uint16 callpathIdx, bytes cmd, bytes conds, bytes tip, bytes signature) returns (bytes memory)

```

callpathIdx andcmd are the same parameters that would be passed to the standarduserCmd and will behave identically.

Conds

conds is a bytestring encoded in a fixed format that governs the conditions of the relayer call. If any of the set conditions are not met at execution call, the transaction will fail. The bytestring is encoded as follows

```

Copy conds = abi.encode( deadline, // uint48 alive, // uint48 salt, // bytes32 nonce, // uint32 relayer) // address

```

block.timestamp must occur beforedeadline and afteralive

nonce defines a user-specific relayer nonce that increments by one on every relayer call. The user's current on-chain nonce must match the condition or the transaction will falt. This is to prevent replay attacks. Every unique value ofsalt defines a unique nonce track (that starts at 0), this allows for multidimensional nonces.

relayer restricts the address of the relayer that can execute the command. Eithermsg.sender ortx.origin must match this address or the contract call will revert. If this value is set toaddress(0) the condition is unenforced.

tip

The bytestring oftip can either be empty (in which case no tip is paid to the relayer) or can be set using the following fixed encoding scheme:

```

Copy tip = abi.encode( token, // address amount, // uint128 recv) // address

```

token is the token the tip is being paid in.amount is the total amount of the token being paid as a tip to relayer.

recv is the recipient of the relayer tip. It can either be set to a specific address (either EOA or smart contract), or can be generically paid based on the following magic values:

- address(256)
-
    - Tip is paid to themsg.sender
- of the contract caller
- address(512)
-
    - Tip is paid to thetx.origin
- of the Ethereum transaction.
-

Tips are always paid from the surplus collateral balance of the end-user. If the end-user has insufficient surplus collateral the contract call will revert.

signature

This is the EIP-712 signature of the command and conditions. It is constructed from the standard v, r, and s recovery signature as follows:

```
```

```
Copy signature = abi.encode( v, // uint8 r, // uint256 s) // uint256
```

The signature is constructed using the EIP-712 standard. The EIP-712 domain hash is constructed as:

```
Copy const domain = { name: "CrocSwap", chainId: [chain ID], verifyingContract: [CrocSwapDex contract address], version: "1.0" }
```

The typed content hash is constructed as

```
Copy CrocRelayerCall: [ { name: "callpath", type: "uint8"}, { name: "cmd", type: "bytes" }, { name: "conds", type: "bytes" }, { name: "tip", type: "bytes" } ]
```