WETHGateway

WETHGateway

The WETH Gateway contract is a helper to easily wrap and unwrap ETH (or native currency of chain eg. MATIC, AVAX etc) as necessary when interacting with the protocol.

Write Methods

depositETH

function depositETH(address pool, address onBehalfOf, uint16 referralCode)

Supplies themsg.value amount of ETH (or native chain token) into the Aave pool, minting the same amount of corresponding aWETH and transferring them to theonBehalfOf address.

Ensure that thedepositETH() transaction also includes the amount of ETH you are supplying in themsg.value . Call Params

Name Type Description pool address address of the targeted pool onBehalfOf address address who will receive the aWETH. Use msg.sender when the aTokens should be sent to the caller. referralCode uint16 unique code for 3rd party referral program integration. 0 for no referra

withdrawETH

function withdrawETH(address pool, uint256 amount, address to)

Withdrawsamount of the WETH (or wrapped native chain token), unwraps it and transfers ETH (or native chain token) to theto address.

TheWETHGateway contract must have an approved token allowance to spendaWETH on behalf of the user, example:

```

Copy IERC20(aWETHAddress).approve(wethGatewayAmount, amount)

```

Call Params

Name Type Description pool address address of the targeted pool amount uint256 amount to be withdrawn, expressed in wei units. Use type(uint).max to withdraw the entire balance. to address address that will receive the unwrapped ETH

repayETH

function repayETH(address pool, uint256 amount, uint256 rateMode, address onBehalfOf)

RepaysonBehalfOf 's debtamount of ETH () which has arateMode .

Ensure that therepayETH() transaction also includes the amount of ETH you are repaying in themsg.value . Parameter Name Type Description lendingPool address address of the targeted underlying lending pool amount uint256 amount to be borrowed, expressed in wei units. Use uint(-1) to repay the entire debt, ONLY when the repayment is not executed on behalf of a 3rd party. In case of repayments on behalf of another user, it's recommended to send an _amount slightly higher than the current borrowed amount. rateMode uint256 the type of borrow debt. Stable: 1, Variable: 2 onBehalfOf address address of user who will incur the debt. Use msg.sender when not calling on behalf of a different user.

borrowETH

function borrowETH(address pool, uint256 amount, uint256 interestRateMode, uint16 referralCode)

Borrowsamount of WETH withinterestRateMode , sending theamount of unwrapped WETH tomsg.sender .

TheWETHGateway contract must have anapproved credit delegation to borrowWETH on behalf of the the caller, example:

```

Copy IVariableDebtToken(wethAddress).approveDelegation(wethGatewayAddress, amount)

```

Call Params

Name Type Description pool address address of the targeted pool amount uint256 amount to be borrowed, expressed in wei units interestRateMode uint256 the type of borrow debt. Stable: 1, Variable: 2 referralCode uint16 unique code for 3rd party referral program integration. 0 for no referral code.

withdrawETHWithPermit

function withdrawETHWithPermit(address pool, uint256 amount, address to, uint256 deadline, uint8 permitV, bytes32 permitR, bytes32 permitS)

Withdrawsamount of the WETH (or wrapped native chain token) without a separate approval tx. The ETH (or native chain token) is sent to theto address.

Call Params

Name Type Description pool address address of the targeted pool amount uint256 amount of aWETH (or aToken corresponding to native token of chain) that will be burnt to withdraw ETH (or native chain token) to address address that will receive the ETH (or native chain token) deadline uint256 unix timestamp till which the signature is valid permitV uint8 Signature parameter v permitR bytes32 Signature parameter r permitS bytes32 Signature parameter s

emergencyTokenTransfer

function emergencyTokenTransfer(address Token, address to, uint256 amount)

Method for ERC20 recovery in case of stuck tokens due direct transfers to the contract address.

Can be called only by the owner of the contract i.e. Aave Governance

emergencyEtherTransfer

function emergencyEtherTransfer(address to, uint256 amount)

Method for ETH (or native chain token) recovery in case of stuck ETH due selfdestruct or transfer ether to pre-computated contract address before deployment.

Can be called only by the owner of the contract i.e. Aave Governance.

View

getWETHAddress

function getWETHAddress()

Returns the WETH address used by the WETHGateway.

ABI

WETHGateway ABI ```

Copy [{ "inputs": [{ "internalType": "address", "name": "weth", "type": "address" }, { "internalType": "address", "name": "owner", "type": "address" }], "stateMutability": "nonpayable", "type": "constructor" }, { "anonymous": false, "inputs": [{ "indexed": true, "internalType": "address", "name": "previousOwner", "type": "address" }, { "indexed": true, "internalType": "address", "name": "newOwner", "type": "address" }], "name": "OwnershipTransferred", "type": "event" }, { "stateMutability": "payable", "type": "fallback" }, { "inputs": [{ "internalType": "address", "name": "pool", "type": "address" }], "name": "authorizePool", "outputs": [], "stateMutability": "nonpayable", "type": "function" }, { "inputs": [{ "internalType": "address", "name": "pool", "type": "address" }, { "internalType": "uint256", "name": "amount", "type": "uint256" }, { "internalType": "uint256", "name": "interesRateMode", "type": "uint256" }, { "internalType": "uint16", "name": "referralCode", "type": "uint16" }], "name": "borrowETH", "outputs": [], "stateMutability": "nonpayable", "type": "function" }, { "inputs": [{ "internalType": "address", "name": "pool", "type": "address" }, { "internalType": "address", "name": "onBehalfOf", "type": "address" }, { "internalType": "uint16", "name": "referralCode", "type": "uint16" }], "name": "depositETH", "outputs": [], "stateMutability": "payable", "type": "function" }, { "inputs": [{ "internalType": "address", "name": "to", "type": "address" }, { "internalType": "uint256", "name": "amount", "type": "uint256" }], "name": "emergencyEtherTransfer", "outputs": [], "stateMutability": "nonpayable", "type": "function" }, { "inputs": [{ "internalType": "address", "name": "token", "type": "address" }, { "internalType": "address", "name": "to", "type": "address" }, { "internalType": "uint256", "name": "amount", "type": "uint256" }], "name": "emergencyTokenTransfer", "outputs": [], "stateMutability": "nonpayable", "type": "function" }, { "inputs": [], "name": "getWETHAddress", "outputs": [{ "internalType": "address", "name": "", "type": "address" }], "stateMutability": "view", "type": "function" }, { "inputs": [], "name": "owner", "outputs": [{ "internalType": "address", "name": "", "type": "address" }], "stateMutability": "view", "type": "function" }, { "inputs": [], "name": "renounceOwnership", "outputs": [], "stateMutability": "nonpayable", "type": "function" }, { "inputs": [{ "internalType": "address", "name": "pool", "type": "address" }, { "internalType": "uint256", "name": "amount", "type": "uint256" }, { "internalType": "uint256", "name": "rateMode", "type": "uint256" }, {

"internalType": "address", "name": "onBehalfOf", "type": "address" }], "name": "repayETH", "outputs": [], "stateMutability": "payable", "type": "function" }, { "inputs": [{ "internalType": "address", "name": "newOwner", "type": "address" }], "name": "transferOwnership", "outputs": [], "stateMutability": "nonpayable", "type": "function" }, { "inputs": [{ "internalType": "address", "name": "pool", "type": "address" }, { "internalType": "uint256", "name": "amount", "type": "uint256" }, { "internalType": "address", "name": "to", "type": "address" }], "name": "withdrawETH", "outputs": [], "stateMutability": "nonpayable", "type": "function" }, { "inputs": [{ "internalType": "address", "name": "pool", "type": "address" }, { "internalType": "uint256", "name": "amount", "type": "uint256" }, { "internalType": "address", "name": "to", "type": "address" }, { "internalType": "uint256", "name": "deadline", "type": "uint256" }, { "internalType": "uint8", "name": "permitV", "type": "uint8" }, { "internalType": "bytes32", "name": "permitR", "type": "bytes32" }, { "internalType": "bytes32", "name": "permitS", "type": "bytes32" }], "name": "withdrawETHWithPermit", "outputs": [], "stateMutability": "nonpayable", "type": "function" }, { "stateMutability": "payable", "type": "receive" }]

```

Previous WalletBalanceProvider Next AToken Last updated5 months ago On this page *WETHGateway * Write Methods * depositETH * withdrawETH * repayETH * borrowETH * withdrawETHWithPermit * emergencyEtherTransfer * View * getWETHAddress * ABI

Was this helpful?
```