# **Contract Asset.sol**

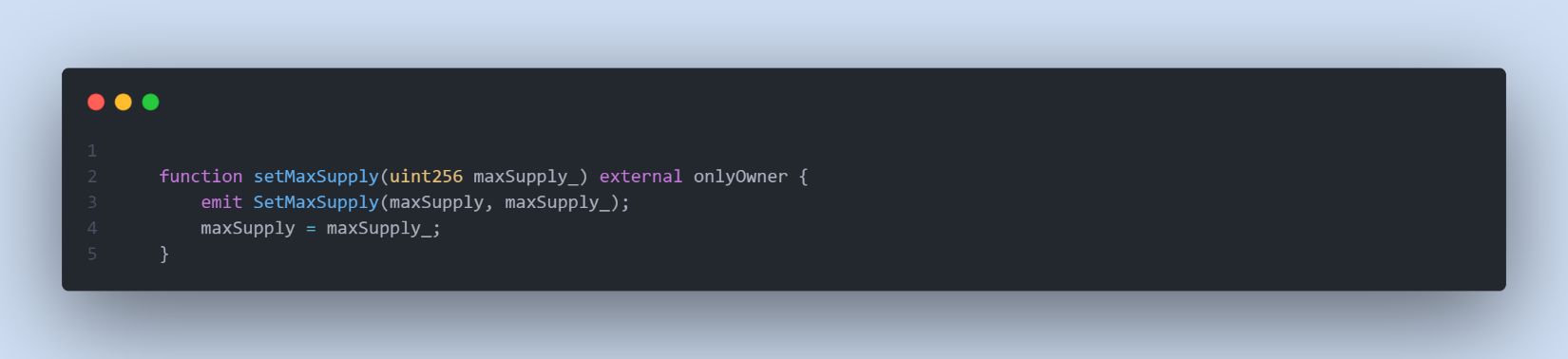
# **Medium severity issues**

* **Centralization Risks**

**In the contract, the owner has authority over the functions shown below.**

* function setPool(address pool\_);
* function setMaxSupply(uint256 maxSupply\_)





**Any compromise to the onlyOwner may allow the hacker to take advantage of this authority and drastically change the stored addresses and state.**

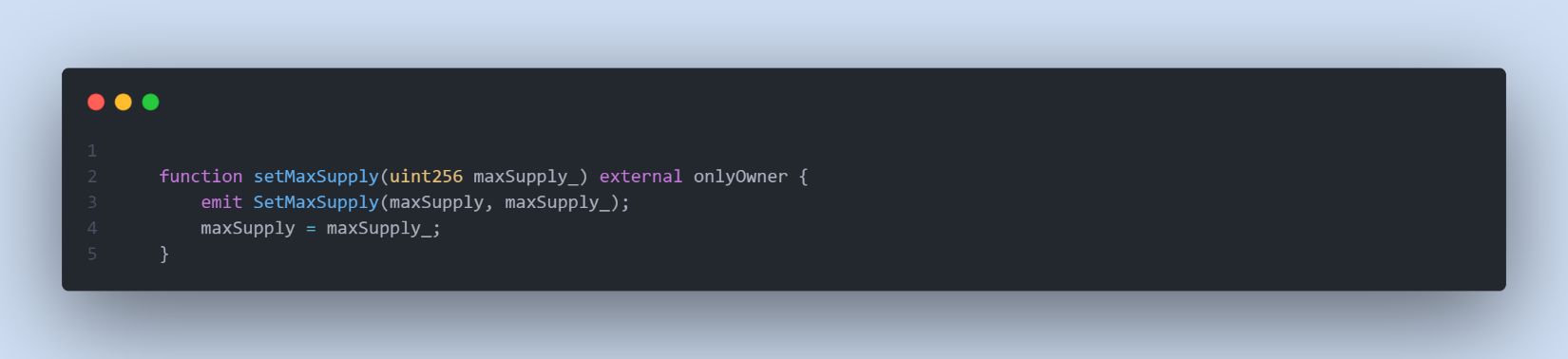
Remediation

The risk describes the current project design and potentially makes iterations to improve in the security operation and level of decentralization, which in most cases cannot be resolved entirely at the present stage. We advise the client to carefully manage the privileged account's private key to avoid any potential risks of being hacked. In general, we strongly recommend centralized privileges or roles in the protocol be improved via a decentralized mechanism or smart-contract-based accounts with enhanced security practices, e.g., multisignature wallets.

# **Low severity issues**

* **Value Validation check**

**The input maxSupply\_ is not checked to see if it is greater than 0 in function setMaxSupply.**

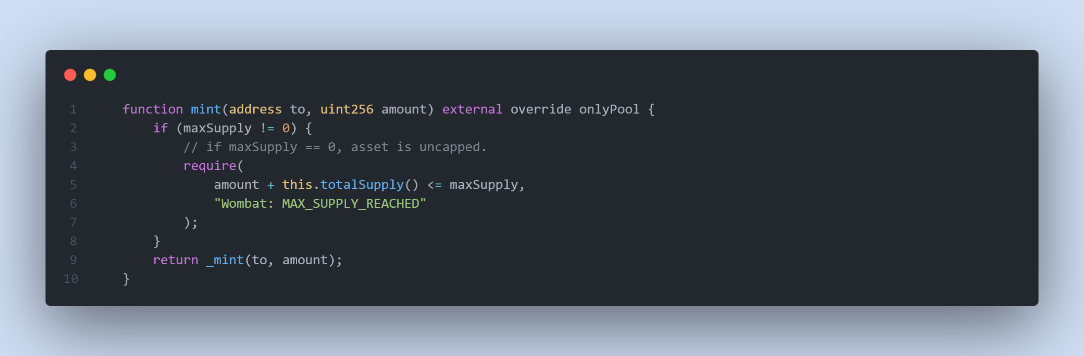


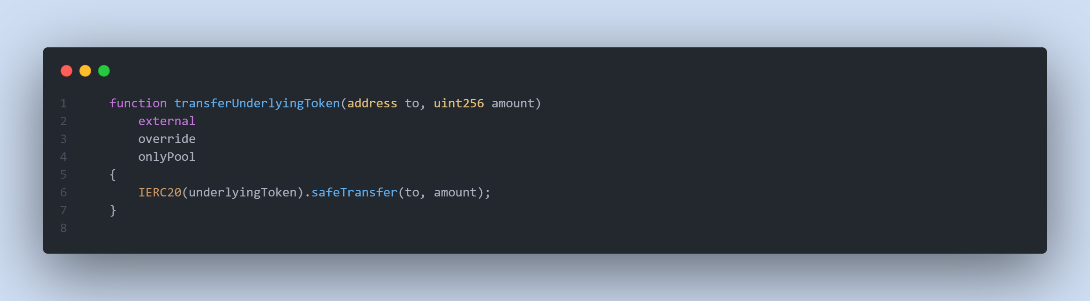
Remediation

**We advise adding a require statement for setMaxSupply function.**

* **Zero Address Validation check**

**The input address to is not checked in function** transferUnderlyingToken, mint,





Remediation

**We advise adding a require statement for** above **functions about to input, to not be zero address.**

* **Missing Emit Events**

**There should always be events emitted in the sensitive functions .**

* **function setMaxSupply**

Remediation

**It is recommended emitting events for the sensitive functions that are controlled by centralization roles.**

# **Contract Pool.sol**

# **Medium severity issues**

* **Centralization Risks**

**In the contract, the owner has authority over the functions shown below.**

* **function pause();**
* **function unpause();**
* **function pauseAsset();**
* **function unpauseAsset();**
* **Function setDev();**
* **Function setAmpFactor();**
* **Function setHaircutRate();**
* **Function addAsset();**
* **Function removeAsset()**

**Any compromise to the onlyOwner and \_onlyDev account may allow the hacker to take advantage of this authority and drastically change the stored addresses and state.**

Remediation

The risk describes the current project design and potentially makes iterations to improve in the security operation and level of decentralization, which in most cases cannot be resolved entirely at the present stage. We advise the client to carefully manage the privileged account's private key to avoid any potential risks of being hacked. In general, we strongly recommend centralized privileges or roles in the protocol be improved via a decentralized mechanism or smart-contract-based accounts with enhanced security practices, e.g., multisignature wallets.