**Code Audit Review**   
  
***Suggestion 1:***

Use **ReentrancyGuard** so avoid Async issues while minting.

function safeMint(address \_to, uint256 quantity) external **nonReentrant** onlyRole(MINTER\_ROLE)  
{  
require(totalSupply() + quantity <= MAXIMUM\_SUPPLY, "Max supply reached");  
\_safeMint(\_to, quantity);  
}  
  
Note🡪 It can also be used in buying tickets to avoid async calling and overselling in case of large load.

***Suggestion 2:***

**Please add a require statement to make sure that length of both the arrays are equal.**   
  
function uploadWhitelist(address[] calldata addresses, uint256[] calldata counts) public onlyOwner {  
for (uint256 i = 0; i < addresses.length; i++) {  
 addressToMaxMintWhitelist[addresses[i]] = counts[i];  
}  
}

***Suggestion 3:***

// Simply Make it a toggle function

function makePhaseOneActive() external onlyOwner {  
 phaseOneActive = !phaseOneActive ;  
}

// We can also make other phases turned off if that aligns with requirement

function makePhaseOneActive() external onlyOwner {  
 phaseOneActive = !phaseOneActive ;  
 phaseTwoActive = false;  
 phaseThreeActive = false;  
}