SECURITY AUDIT REPORT

Ganjes Token (GNJS)

Contract File:	contracts/gnjToken.sol
Token Name:	Ganjes (GNJS)
Token Standard:	BEP20 (ERC20 Compatible)
Total Supply:	10,000,000 GNJS (Fixed)
Decimals:	18
Audit Date:	August 21, 2025
Audit Type:	OpenZeppelin Style Security Review
Overall Risk Level:	MEDIUM

EXECUTIVE SUMMARY

The Ganjes Token (GNJS) is a BEP20-compatible token implementing multi-signature governance, pause functionality, and burn capability. The contract demonstrates solid security practices with proper access controls and protection mechanisms. However, certain admin functions bypass the multi-signature governance system, creating centralization risks that should be addressed before deployment.

RISK ASSESSMENT SUMMARY

Risk Level	Count	Description
HIGH	0	Critical security vulnerabilities
MEDIUM	3	Centralization and governance bypass issues
LOW	2	Minor improvements recommended
INFORMATIONAL	0	Code quality suggestions

DETAILED SECURITY FINDINGS

MEDIUM RISK ISSUES

1. Direct Admin Function Calls Bypass Multi-Sig

Location: Lines 561-584

Description: Admin functions pause(), unpause(), and emergencyPause() can be called directly by owners, bypassing multi-sig governance and timelock protection.

Impact: Centralization risk that defeats the purpose of multi-sig governance.

Recommendation: Remove direct callable admin functions. All admin actions should go through submitTransaction().

2. Emergency Withdrawal Lacks Protection

Location: Lines 720-726

Description: emergencyWithdraw() function can be called directly by any owner without multi-sig approval.

Impact: Potential for unauthorized token extraction from contract.

Recommendation: Implement multi-sig approval and timelock for emergency withdrawals.

3. Individual Owner Token Burning

Location: Lines 594-597

Description: burn() function allows any owner to burn tokens from their balance without multi-sig approval.

Impact: Individual owners can make unilateral decisions affecting token supply.

Recommendation: Consider requiring multi-sig approval for large burn amounts.

LOW RISK ISSUES

1. Missing Input Validation

Location: Lines 654-670

Description: Transfer functions don't validate recipient addresses against known problematic contracts.

Recommendation: Consider implementing blacklist functionality for known problematic addresses.

2. Event Logging Gaps

Location: Various locations

Description: Some admin actions don't emit specific events for better transparency.

Recommendation: Add more detailed event logging for administrative actions.

SECURE IMPLEMENTATIONS ✓

• Multi-signature governance system with proper timelock implementation

• Reentrancy protection using custom ReentrancyGuard

• Pausable mechanism with emergency pause capability

- Fixed supply model preventing inflation attacks
- Standard BEP20 compliance with proper event emissions
- Safe mathematical operations with appropriate overflow protection
- Proper access control mechanisms with owner verification

COMPLIANCE ANALYSIS

Standard	Status	Notes
ERC20/BEP20	✓ COMPLIANT	Full BEP20 standard compliance
Ownable	✓ ENHANCED	Multi-sig implementation exceeds standard
Pausable	✓ COMPLIANT	Proper implementation
ReentrancyGuard	✓ COMPLIANT	Custom but secure implementation

PRIORITY RECOMMENDATIONS

HIGH PRIORITY

1. Enforce multi-signature governance for ALL admin functions

2. Implement proper safeguards for emergency withdrawal functionality

MEDIUM PRIORITY

1. Add burn amount limits for individual owners

2. Enhance event logging for better transparency

LOW PRIORITY

1. Implement address validation and blacklist functionality

2. Add comprehensive edge case testing

CONCLUSION

The Ganjes Token contract demonstrates solid security practices with multi-signature governance and proper access controls. The implementation follows industry standards and includes important security features like reentrancy protection and pause functionality. However, the direct callable admin functions create centralization risks that should be addressed before deployment. **Final Recommendation:** Address the medium-risk issues, particularly the direct admin function calls that bypass multi-sig governance, before proceeding with mainnet deployment. The contract is otherwise well-implemented and secure.

Audit Methodology: OpenZeppelin Security Standards