SMART CONTRACT SECURITY AUDIT REPORT

BDOGE_TOKEN.sol (BattleDogeToken)

Contract	BattleDogeToken.sol
Solidity Version	0.8.33 (locked)
OpenZeppelin Basis	v5.4.0 (embedded, no imports)
Audit Date	January 5, 2026
OVERALL VERDICT	**PASS - Ready for Production**

1. Executive Summary

This audit examines the BDOGE_TOKEN.sol smart contract, a single-file ERC-20 token implementation with embedded OpenZeppelin v5.4.0 code. The contract has been thoroughly analyzed for syntax correctness, security vulnerabilities, and compliance with OpenZeppelin's official implementation.

Key Finding: The contract is correctly implemented, compiles without errors or warnings, and accurately reproduces the OpenZeppelin ERC-20 implementation. No critical, high, medium, or low severity issues were identified.

2. Audit Scope

Files Reviewed: BDOGE_TOKEN.sol (single file, 221 lines)

2.1 Components Analyzed

Component	Status
Context (abstract)	VERIFIED - Matches OZ v5.0.1
IERC20 (interface)	VERIFIED - Matches OZ v5.4.0
IERC20Metadata (interface)	VERIFIED - Matches OZ v5.4.0
IERC20Errors (interface)	VERIFIED - Matches ERC-6093
ERC20 (abstract)	VERIFIED - Matches OZ v5.4.0
BattleDogeToken (concrete)	VERIFIED - Correctly implemented

3. Detailed Findings

3.1 OpenZeppelin Code Verification

Cross-referenced against OpenZeppelin Contracts GitHub repository (v5.4.0 and current master). All embedded code matches the official implementation exactly, including Context.sol, IERC20.sol, IERC20Metadata.sol, draft-IERC6093.sol, and ERC20.sol.

3.2 Compilation Results

Compiler	solc 0.8.33
Optimization	Enabled (200 runs)
Errors / Warnings	**0 / 0**
Bytecode Size	2,869 bytes (well under 24KB limit)

3.3 Security Analysis

Category	Risk	Details
Reentrancy	**None**	Follows checks-effects-interactions pattern
Overflow/Underflow	**None**	Solidity 0.8.x built-in checks; unchecked blocks correct
Access Control	**None**	No admin functions; _mint/_burn are internal only
ETH Handling	**Mitigated**	receive()/fallback() revert; selfdestruct acknowledged
Front-Running	**Standard**	ERC-20 approve() race condition (inherent to standard)

4. Token Specification Verification

Property	Specified	Verified
Name	Battle Doge	✓ **CORRECT**
Symbol	BDOGE	✓ **CORRECT**
Decimals	18	✓ **CORRECT**
Total Supply	100,000,000 × 10^{18}	✓ **CORRECT**
Initial Distribution	100% to deployer	✓ **CORRECT**
Mintable	No (one-time mint only)	✓ **CORRECT**
Burnable	No public burn function	✓ **CORRECT**
Permit (EIP-2612)	Not implemented	✓ **CORRECT**
Admin/Owner	None	✓ **CORRECT**

5. Recommendations

Pre-Deployment: (1) Deploy to testnet first and verify all functions. (2) Verify source code on Etherscan after mainnet deployment. (3) Document deployer wallet and distribution plan.

Post-Deployment: (1) Monitor for unusual transfer patterns in first 48 hours. (2) Ensure adequate DEX liquidity to prevent manipulation.

6. Conclusion

The BDOGE_TOKEN.sol contract is a **well-implemented, minimal ERC-20 token** that correctly embeds OpenZeppelin v5.4.0 code. The implementation compiles without errors, matches OpenZeppelin's official implementation exactly, contains no vulnerabilities, has minimal attack surface with no admin functions, and correctly implements ETH rejection.

✓ This contract is APPROVED for production deployment.

Disclaimer: This audit is based on the code provided at the time of review. It does not guarantee the absence of all vulnerabilities. Smart contract security is an evolving field and new attack vectors may emerge. Always maintain operational security practices post-deployment.