SMCAUDIT

SMART CONTRACT
SECURITY AUDIT



AUDIT REPORT

BTC Dragon (BTCDragon) Security Review

PROJECT	BTC Dragon (BTCDragon)
CHAIN	BNB Chain (BSC, chain id 56)
TOKEN	0x1Ee8a2f28586e542af677eB15Fd00430f98d8fd8
DIVIDEND TRACKER	0xa67161238823209801d8629D64D293204C042cbd
VERSION	Deployed bytecode · Sourcify exact match (matchId 30939601) · solc 0.8.19 · not a proxy
REPORT ID	SMC-BTCDRAGON-202610-01
DATE	19 October 2026
PREPARED BY	SMC Audits
STATUS	Final

0 0

00 Confidentiality notice

This report is prepared by SMC Audits for the named project and contains confidential information, including identified security findings.

It should not be shared publicly until the project has remediated the issues described, and only with SMC Audits' agreement. Publication as part of public verification follows the process agreed in the engagement.

SMC AUDITS

01

01 Contents

02	Executive summary
03	Project overview
04	Scope
05	Methodology
06	Severity classification
07	Findings summary
08	Detailed findings
09	Remediation status
10	Conclusion
11	Disclaimer

SMC AUDITS

02

02 Executive summary

SMC Audits reviewed the BTC Dragon (BTCDragon) smart contracts on BNB Chain at address 0x1Ee8a2f28586e542af677eB15Fd00430f98d8fd8, using the source verified on Sourcify (an exact match of both creation and runtime bytecode, solc 0.8.19).

The review ran from 15 October 2026 to 19 October 2026 and covered the deployed BTC_Dragon token and the DividendTracker it deployed (0xa67161238823209801d8629D64D293204C042cbd) in full, including the linked IterableMapping library, the fee, swap-back, auto-liquidity, burn, limit and dividend logic, and the administrative history up to the renounce, cross-checked against live on-chain state and the LP lock. The code is a generator-style fee token (template header from the 20lab token generator) built on OpenZeppelin 4.9 with a BTCB dividend tracker; it is not a bespoke protocol. The manual review, in five passes (owner and privileged powers, attacker view, line-by-line, severity challenge, and a final adversarial pass), was supported by cast and Forge (Foundry 1.8.5) for live state and history; 38 Foundry fork tests at BSC block 127,820,893 covering every finding and every refuted candidate; an independent confirmation run with its own 38 fork tests; Slither 0.11.6 and Aderyn 0.6.8; and Sourcify for source and metadata verification. All 76 tests pass.

The review identified 15 findings: 0 critical, 0 high, 0 medium, 2 low, and 13 informational. No issue lets an outside attacker take user funds, and no wallet can halt sells. Ownership was renounced in January 2024 and our tests confirm that every owner and tracker admin function is dead. Unlike many renounced fee tokens, the marketing wallet is paid with a 2,300-gas send whose result is ignored, so an EIP-7702 delegation on that wallet cannot stop sells, even under BSC's per-transaction gas cap of 16,777,216 (BEP-652); a worst-case sell with the swap-back armed and every holder due a payout uses 1.16 million gas. The most significant observation is time-bound: the marketing wallet owns the PinkLock02 lock holding 71.37% of the LP, which unlocks on 13 January 2029; after that date it can remove the liquidity and sell fee-free for about 23.99 BNB, 91.7% of the pool's BNB today (SMC-L-01, Low). The other low item is the standard ERC-20 approve race (SMC-L-02). The informational items cover the frozen configuration, permanent fee exemptions, supply concentration, the sell path's dependency on the WBNB/BTCB pool, zero-slippage swaps, and fee and dividend edge behaviour.

0	0	0	2	13	15
CRITICAL	HIGH	MEDIUM	LOW	INFO	TOTAL

As of this report, 0 of 15 findings have been resolved and verified by SMC Audits. The contracts are immutable and ownership is renounced, so the recommendations are advisory: they apply to the project team, token holders, integrators and any future redeployment rather than to the live bytecode. Any residual risk is described in the conclusion.

03

03 Project overview

BTC Dragon (BTCDragon) is a taxed BEP-20 dividend token on BNB Chain. Buys and sells through its PancakeSwap V2 pair pay 9% (6% marketing, 1% auto-liquidity, 2% holder rewards); wallet-to-wallet transfers pay 1%, all to holder rewards. Collected fees are swapped back automatically: the marketing share is sold for BNB and sent to the marketing wallet, the liquidity share is added to the pair with the LP sent to address(0), and the rewards share is sold for BTCB and distributed to holders by the dividend tracker. The code is a generator template compiled with solc 0.8.19 on stock OpenZeppelin 4.9 (ERC20, ERC20Burnable, Ownable2Step). Neither contract is a proxy or upgradeable.

System at a glance

COMPONENT	ROLE
BTC_Dragon token contract	Balances, transfer / transferFrom, allowances, burn / burnFrom. 500,000,000,000 BTCD minted once to the supply recipient (18 decimals); 434,460,627,642.95 today after burns; no further mint.
Fee and swap-back logic (in-contract)	Non-exempt pair trades pay 9% to the contract, wallet transfers 1%. When the pending fee inventory reaches 0.5% of the pair's token balance, any transfer that is not a buy and not from the router runs the swap-back before moving the sender's tokens. The marketing BNB is paid with send (2,300 gas, result ignored).
DividendTracker (0xa671...2cbd)	Non-transferable shares mirroring BTCD balances of at least 1,000,000 BTCD; 3,695 holders; 0.36653 BTCB distributed to date. Every transfer processes holders for up to 300,000 gas inside a try / catch. Its owner is the token contract. Uses the linked library IterableMapping (0x31ee...bad5).
Admin / ownership	owner() held every setter (fees capped at 25% per side, exemptions, pairs, limits, swap threshold, processing gas, claim wait). Ownership was renounced to address(0) at block 35,373,852 (January 2024); pendingOwner() is also address(0). The marketing wallet 0x4B2A841135e5d1b819eB34d4BB52A1E7aB006bC1 is an EOA with no code.
External integrations	PancakeSwap V2 router 0x10ED...024E and pair BTCD/WBNB 0x5967...0101 (26.149 BNB and 84.207B BTCD at the fork block). BTCB, the WBNB/BTCB pool used for reward swaps, PinkLock02 (0x4079...1BBE) and market conditions are out of scope, but their live state was checked.

Key mechanisms to note

- Renouncing ownership removed every admin path. owner() and pendingOwner() return address(0), initialize has been consumed, and the tracker's admin functions can only be called by the token, which has no function that hands tracker ownership to anyone. What remains is permissionless: claim, process, addLiquidityFromLeftoverTokens, burn / burnFrom and the tracker's

distributeDividends. The marketing wallet is still a privileged recipient (fee-exempt, paid by every swap-back, owner of the LP lock), which is the basis of SMC-L-01 and SMC-I-02.

- The 9% tax applies to the registered pair only. Only the BTCD/WBNB pair is in AMMPairs, and no pair can be added, so trades through any other pool pay the 1% transfer fee (SMC-I-05). No other BTCD V2 pool exists today.
- Liquidity is locked until 2029. 71.37% of the LP (1,000,000.00 of 1,401,054.22 LP) is in PinkLock02 lock 1192506, owned by the marketing wallet, single unlock at 13 January 2029 05:29 UTC; the other 28.62% sits at address(0) (initial burn plus auto-liquidity). There are no BTCD token locks.
- Holder rewards depend on a second pool. The rewards share is swapped BTCD to WBNB to BTCB inside the swap-back, so the transfer path relies on the WBNB/BTCB pool having liquidity (SMC-I-04).

04

04 Scope

ITEM	DETAIL
Project	BTC Dragon (BTCDragon)
Chain	BNB Chain (BSC, chain id 56)
Language	Solidity
Contracts	0x1Ee8a2f28586e542af677eB15Fd00430f98d8fd8 (token) and 0xa67161238823209801d8629D64D293204C042cbd (dividend tracker, deployed by the token constructor), with the linked library IterableMapping at 0x31ee4a53Bd2C1c339662DfFB973017EF81A6bad5; verified source via Sourcify.
Commit / version	Deployed bytecode; Sourcify exact match of creation and runtime bytecode (matchId 30939601, verified 2026-06-04). Deployed at block 35,351,245 on 2024-01-18 by 0x6D18...56A9 (supply recipient and original owner). Not a proxy.
Compiler	solc 0.8.19+commit.7dd6d404; optimizer enabled (runs 200); EVM version paris (default); library IterableMapping linked.

Files reviewed

FILE	LINES	NOTES
Token.sol	516	BTC_Dragon: fees, swap-back, auto-liquidity, burn, limits, trading switch, dividend wiring. SHA-256 e9777fa5...5b3a2.

SMC AUDITS

TokenDividendTracker.sol	595	TruncatedERC20, DividendPayingToken, IterableMapping, DividendTracker, DividendTrackerFunctions. SHA-256 a36195d5...8b956.
ERC20.sol, ERC20Burnable.sol, Ownable.sol, Ownable2Step.sol, Context.sol, interfaces	833	OpenZeppelin 4.9 unmodified apart from import paths; Uniswap V2 interfaces.
Initializable.sol	46	Minimal one-shot initializer, consumed at deployment.

A local build of the verified source matches the on-chain runtime bytecode byte for byte apart from the metadata hash; there are no immutables.

Out of scope

Off-chain components, the project website and front end. The BTCB token, PancakeSwap contracts, the WBNB/BTCB pool, PinkLock02, and market or economic conditions; their live state was read, and BTCB's admin surface and the PinkLock02 lock and unlock paths were exercised on the fork, because they change the risk to holders. Deployment process and key management history. No evidence gap affects a Critical, High, or Medium conclusion.

05

05 Methodology

Every SMC Audits review follows the same process, so the result means the same thing on every engagement.

1. Threat modeling. Understand what the contract does, where value moves, and who holds privilege.
2. Manual review. A line-by-line reading of the code by the audit team. This is the core of the work and where most issues are found.
3. Tooling. Static analysis and security tooling support the manual review, they do not replace it.
4. Proof of concept. Every material finding is confirmed with a working proof of concept rather than asserted.
5. Reporting. Findings are written up with severity, reproduction, and a clear recommendation.
6. Re-audit. Fixes are verified by the team to confirm the issue is closed without introducing a new one.

Review checklist

Access control, signer and owner checks; privilege and centralization risk; arithmetic, overflow, and precision; reentrancy and external call safety; token and supply logic, mint and burn; fee, tax, and reflection mechanics; approvals and allowance handling; oracle and price handling; upgradeability and

SMC AUDITS

proxy risk; pause, blacklist, and emergency switches; front-running and MEV exposure; denial of service and gas griefing.

06

06 Severity classification

SEVERITY	MEANING
CRITICAL	Direct loss of funds or full compromise, realistic to exploit
HIGH	Serious impact, exploitable under conditions a project would actually meet
MEDIUM	Meaningful risk, limited by conditions or the size of impact
LOW	Minor risk, hard to exploit or low in impact
INFO	Best practice and code quality notes worth addressing over time

07

07 Findings summary

ID	TITLE	SEVERITY	STATUS
SMC-L-01	Marketing wallet controls 71.37% of LP after the January 2029 unlock	LOW	Open
SMC-L-02	ERC-20 approve race condition (SWC-114)	LOW	Open
SMC-I-01	Privileged functions rely entirely on renounced ownership	INFO	Acknowledged
SMC-I-02	Permanent fee exemptions, including a PinkLock02 transfer conduit	INFO	Open
SMC-I-03	Supply concentration in deployer-funded wallets	INFO	Open
SMC-I-04	Sell and transfer path depends on WBNB/BTCB pool liquidity	INFO	Open
SMC-I-05	Trades through unregistered pools pay 1% instead of 9%	INFO	Open
SMC-I-06	Swap-back and auto-liquidity use zero slippage limits	INFO	Open

SMC AUDITS

SMC-I-07	Zero-value transfers trigger the swap-back	INFO	Open
SMC-I-08	Public leftover-liquidity function nests the swap-back	INFO	Open
SMC-I-09	Marketing payment sweeps the contract's whole BNB balance	INFO	Open
SMC-I-10	Transfers to the router are fee-free and sweepable	INFO	Open
SMC-I-11	Untracked liquidity leftover and fee rounding dust	INFO	Open
SMC-I-12	Dividend tracker behaviour and code quality	INFO	Open
SMC-I-13	Permanent wallet and transaction limits	INFO	Open

0	0	0	2	13	15
CRITICAL	HIGH	MEDIUM	LOW	INFO	TOTAL

SMC AUDITS

08

08 Detailed findings

PoC note: fork tests run at BSC block 127,820,893 (15 October 2026) against the live bytecode, with an independent confirmation run. No live mainnet transaction was made. All 76 tests pass. Full tests are provided in the engagement materials.

SMC-L-01 LOW Open

Marketing wallet controls 71.37% of LP after the January 2029 unlock

CATEGORY

Privilege and centralization risk

LOCATION

Live state: PinkLock02 0x4079...1BBE lock 1192506; marketingAddressSetup L173-181 (fee and limit exemption of the marketing wallet), constructor L98

DESCRIPTION

PinkLock02 lock 1192506 holds 1,000,000.00 of the pair's 1,401,054.22 LP tokens (71.37%). Its owner is the marketing wallet 0x4B2A...6bC1, it has a single unlock at 13 January 2029 05:29 UTC, and nothing has been unlocked. The remaining 28.62% of LP sits at address(0) and can never be withdrawn. The marketing wallet is permanently fee- and limit-exempt, so after the unlock it can remove the liquidity and sell the BTCD it receives without the 9% tax.

IMPACT

After the unlock date the marketing wallet can take most of the pool's BNB in two transactions. At today's reserves it would receive 18.66 BNB from removing the LP plus 5.32 BNB from selling the 60.10B BTCD fee-free into what remains, 23.99 BNB in total, 91.7% of the pool's 26.15 BNB, with the price down 91.8% for holders. Rated Low rather than Medium: the lock is live for about 2.25 more years, the date and owner are public on-chain, and nothing can be done before then. It becomes a Medium-class risk (an expired large lock) once 13 January 2029 passes without a relock.

PROOF OF CONCEPT

P1_Privileged.t.sol::test_lpLock_after2029 at block 127,820,893: unlock(1192506) reverts before the date; after warping past it, the marketing wallet unlocks, calls removeLiquidityETHSupportingFeeOnTransferTokens and receives 18.663 BNB and 60,101,905,890.66 BTCD, then sells the BTCD fee-free for 5.325 BNB. Total 23.988 BNB, 91.73% of the pre-pull pool BNB; price impact -91.83%.

RECOMMENDATION

Live contract: no code change applies. The marketing wallet owner should extend the lock well before January 2029 or burn the LP, and publish the plan. Holders and listing sites should track lock 1192506's unlock date. Future versions: send launch LP to a burn address or a long-dated lock held by a multisig, and remove fee exemptions from wallets that own LP before renouncing.

SMC AUDITS

SMC-L-02 LOW Open

ERC-20 approve race condition (SWC-114)

CATEGORY

Approvals and allowance handling

LOCATION

OpenZeppelin ERC20.approve(address,uint256) L136, transferFrom(address,address,uint256) L158 (mitigations increaseAllowance L177, decreaseAllowance L197)

DESCRIPTION

approve overwrites the allowance. If an owner changes a non-zero allowance to another non-zero value, the spender can front-run the change, spend the old allowance and then spend the new one. Otherwise approve and transferFrom behave as stock OpenZeppelin 4.9: the fee is taken from the gross amount, an infinite allowance is not reduced, and a finite allowance is reduced by the gross amount.

IMPACT

A spender can take up to the sum of both allowances. It needs a specific victim action plus a transaction-ordering advantage, and the contract already offers increaseAllowance / decreaseAllowance, hence Low.

PROOF OF CONCEPT

P5_FinalRun.t.sol::test_F5_11_approveRace: Alice approves Bob for 100,000,000 BTCD, then tries to lower it to 50,000,000. Bob front-runs with transferFrom(alice, bob, 100,000,000) and then spends the new 50,000,000. Bob pulls 150,000,000 BTCD gross (148,500,000 after the 1% transfer fee) against Alice's final intent of 50,000,000.

RECOMMENDATION

Immutable, no code fix. Holders and integrators should use increaseAllowance / decreaseAllowance, or set the allowance to 0 before a new value. Future versions: keep these helpers, or support EIP-2612 permits with exact amounts.

SMC AUDITS

SMC-I-01 INFO Acknowledged

Privileged functions rely entirely on renounced ownership

CATEGORY

Privilege and centralization risk

DESCRIPTION

Every administrative function on the token is onlyOwner: swap threshold, marketing wallet, the four fee groups (capped at 25% per side in total), fee and limit exemptions, dividend exclusion, pair registration, max wallet and max transaction (with floors of 0.1% and 0.05% of supply), the one-way trading switch, processing gas (200,000 to 500,000) and claim wait (1 minute to 7 days). The tracker's admin functions (setBalance, excludeFromDividends, claim(address), process, setRewardToken, claimWaitSetup) are onlyOwner with the token as owner. owner() and pendingOwner() are address(0) and initialize is consumed, so none of these can run again, and the frozen configuration (9% / 9% / 1%, auto-burn 0) can also never be lowered.

CONDITIONAL SEVERITIES WITH A LIVE OWNER

FUNCTION	CAPABILITY	SEVERITY IF OWNER WERE ACTIVE
marketingFeesSetup and the other fee setters	Buy, sell and transfer tax up to 25% each	MEDIUM fee increase within a cap
setAMMPair, excludeFromLimits, updateMaxWalletAmount, updateMaxBuyAmount, updateMaxSellAmount	Tax any address as a pair; tighten limits to their floors	MEDIUM selective trading restrictions
excludeFromDividends, claimWaitSetup, gasForProcessingSetup	Withhold or slow holder rewards	LOW reward manipulation
excludeFromFees, marketingAddressSetup	Create tax-free insider wallets; redirect the marketing BNB	LOW unfair trading terms

IMPACT

None today (status Acknowledged: mitigated by the renounce). This is the main safety fact of the contracts and should be re-verified independently by any reader (cast call 0x1Ee8...8fd8 "owner()(address)"). It does not cover the marketing wallet's LP position in SMC-L-01.

PROOF OF CONCEPT

P1_Privileged.t.sol::test_ownerRenounced_allSettersDead: all 19 owner-gated token entry points revert for the former owner, the marketing wallet and a random caller, and all tracker admin functions revert for outside callers. P5_FinalRun.t.sol::test_F5_07_renouncedResidue: owner() and pendingOwner() are address(0), the tracker's owner is the token, and acceptOwnership, initialize, claimWaitSetup and every tracker admin function revert for the former owner.

RECOMMENDATION

No action on the live contracts. Future versions: use a timelocked multisig instead of renouncing if parameters still need to be managed, and emit events for every configuration change.

SMC-I-02 INFO Open

Permanent fee exemptions, including a PinkLock02 transfer conduit

CATEGORY

Privilege and centralization risk; fee, tax, and reflection mechanics

DESCRIPTION

Four addresses are fee-exempt forever: the supply recipient and former owner 0x6D18...56A9, the token contract, the marketing wallet and PinkLock02. An exempt sender or recipient skips the whole fee block in _transfer. Because PinkLock02 is exempt, a holder can lock BTCD for another owner with a short unlock date and the other owner can unlock it, moving tokens wallet to wallet without the 1% transfer fee.

IMPACT

The supply recipient and the marketing wallet hold 0 BTCD today, so the exemption is a capability, not a live position; it matters for SMC-L-01, where the marketing wallet would sell the LP's BTCD untaxed. The PinkLock02 route only avoids the 1% wallet-transfer fee, which goes entirely to holder rewards; it cannot avoid the 9% sell tax, which is charged at the pair.

PROOF OF CONCEPT

P5_FinalRun.t.sol::test_F5_02_pinkLockFeeFreeConduit: a direct 100,000,000 BTCD transfer delivers 99,000,000; the same amount locked in PinkLock02 for Bob for 60 seconds and unlocked by Bob delivers 100,000,000.

RECOMMENDATION

Live contract: no code change applies. Future versions: keep exemptions to the token and the router path, and do not exempt general-purpose lockers.

SMC AUDITS

SMC-I-03 INFO Open

Supply concentration in deployer-funded wallets

CATEGORY

Privilege and centralization risk

DESCRIPTION

The second-largest holder, 0xe8D88301c27100b2D5c8d6adcCFb36C5F3b45Be8 (10,000,000,000 BTCD, 2.30% of supply), is a small owner-controlled holding contract whose whole balance is two pre-launch allocations of exactly 5,000,000,000 BTCD from the supply recipient. It has never sold. More broadly, before the renounce the supply recipient sent 250B BTCD to the pair at launch, burned 60B and distributed about 200B (46% of today's supply) to about 30 wallets. None of these allocations is locked.

IMPACT

No code flaw. A single key can move holder #2's 2.30% at any time, and the early wallets together may control a large share of the float; a coordinated sale would move the price sharply, though each seller pays the 9% tax and the price impact. Holders should treat these balances as unlocked insider-origin supply.

RECOMMENDATION

The project should disclose which wallets received the pre-launch allocations and either lock or burn what they still hold. Future versions: vest team and partner allocations on-chain.

SMC AUDITS

SMC-I-04 INFO Open

Sell and transfer path depends on WBNB/BTCB pool liquidity

CATEGORY

Denial of service and gas griefing; integration safety

DESCRIPTION

The swap-back runs inside the user's transfer and, when rewards are pending and the tracker has holders, swaps BTCD to WBNB to BTCB through the router with no try. If the WBNB/BTCB pool were emptied of BTCB, the reward swap would revert and take the user's transfer with it. In that state every sell and every wallet transfer from a non-pair address reverts, including from fee-exempt wallets, while buys still work; addLiquidityFromLeftoverTokens cannot clear the pending rewards. Restoring the pool's liquidity restores the token immediately. BTCB itself cannot cause this: it is not a proxy and has no pause or blacklist.

IMPACT

Theoretical. The WBNB/BTCB pool held 2,202.29 WBNB and 19.58 BTCB at the fork block, so draining it costs more than 2,200 BNB to grief a token whose own pool holds 26.15 BNB, and arbitrage would refill it. No privileged wallet can trigger it and it fixes itself when liquidity returns, so it does not meet the bar for an unfixable sell halt.

PROOF OF CONCEPT

P4_RewardPath.t.sol::test_rewardPathDrain_sellsRevertBuysOk: with the swap-back armed and the pool's BTCB set to 1 wei, a 0.2 BNB buy succeeds while the buyer's sell, a wallet transfer and a transfer from the fee-exempt marketing wallet all revert; restoring the reserve makes the sell succeed. P5_FinalRun.t.sol::test_F5_09_rewardTokenNotPausable confirms BTCB has no pause or blacklist.

RECOMMENDATION

Nothing on the live contract. Future versions: wrap the reward swap and distribution in try / catch, or let holders pull rewards outside the transfer path.

SMC AUDITS

SMC-I-05 INFO Open

Trades through unregistered pools pay 1% instead of 9%

CATEGORY

Fee, tax, and reflection mechanics; integration safety

DESCRIPTION

Only the BTCD/WBNB pair is in AMMPairs, and no other pool can be added after the renounce. Transfers to and from any other pool are treated as wallet transfers and pay the 1% transfer fee instead of the 9% trade tax. Removing liquidity through the router is fee-free, because transfers to the router skip the fee. No other BTCD V2 pool exists today (USDT, BTCB and BUSD pairs checked).

IMPACT

No principal at risk. If volume moved to another pool, holders would lose the marketing and liquidity share of that volume and most of the rewards share.

RECOMMENDATION

Nothing on the live contract. Future versions: tax by default and whitelist wallets, or keep pair registration with a timelocked multisig.

SMC-I-06 INFO Open

Swap-back and auto-liquidity use zero slippage limits

CATEGORY

Front-running and MEV exposure

DESCRIPTION

The swap-back sells with amountOutMin = 0, adds liquidity with minimums of 0 and swaps into BTCB with amountOutMin = 0, all with block.timestamp as the deadline.

IMPACT

Not profitable to exploit at current sizes. The swap-back fires at 0.5% of the pair's token balance, and a sandwich pays the 9% tax on both legs. A measured sandwich around a 1.10B BTCD swap-back leaves the attacker down 17.40%.

RECOMMENDATION

Nothing on the live contracts. Future versions: use reserve- or TWAP-based minimum outputs.

SMC AUDITS

SMC-I-07 INFO Open

Zero-value transfers trigger the swap-back

CATEGORY

Front-running and MEV exposure; fee, tax, and reflection mechanics

DESCRIPTION

The fee block is skipped for amount == 0, but the swap-back check is not. Once the pending inventory reaches the threshold, any address, including one holding no BTCD, can run the full swap-back with transfer(anyone, 0).

IMPACT

No loss. Anyone can choose exactly when the swap-back runs and bundle it with their own trades, which makes MEV around it easier to plan; as SMC-I-06 shows, the 9% tax still makes a sandwich unprofitable. It also lets anyone settle pending fees for holders without trading.

RECOMMENDATION

Nothing on the live contract. Future versions: skip the swap-back for zero-amount transfers.

SMC-I-08 INFO Open

Public leftover-liquidity function nests the swap-back

CATEGORY

Reentrancy and external call safety

DESCRIPTION

addLiquidityFromLeftoverTokens is public and does not set _swapping. It turns the contract's untracked tokens into liquidity sent to address(0). If pending fees are at the threshold, the router's transferFrom inside it re-enters _transfer and runs a full swap-back in the middle. We checked that this is safe: OpenZeppelin spends the router's allowance before _transfer runs, so the nested _approve cannot be abused, and the state is consistent afterwards.

IMPACT

No loss; the effect is holder-positive (burned liquidity). A caller can move the price by about 11 bps by converting the leftover, at its own gas cost.

RECOMMENDATION

Nothing on the live contract. Future versions: guard the function with the same _swapping flag.

SMC AUDITS

SMC-I-09 INFO Open

Marketing payment sweeps the contract's whole BNB balance

CATEGORY

Fee, tax, and reflection mechanics

DESCRIPTION

The marketing leg computes coinsReceived = address(this).balance rather than the swap's output, and marketingPortion equals the whole balance. Any BNB in the contract (donations, addLiquidityETH refunds, an earlier failed payment) goes to the marketing wallet at the next swap-back. The payment is a 2,300-gas send whose result is ignored; if it fails, _marketingPending is still cleared and the BNB stays in the contract until a later send succeeds. There is no other way to withdraw BNB.

IMPACT

No holder loss: the BNB is either fee revenue meant for marketing or a voluntary donation. The send design is also what keeps the marketing wallet from being able to halt sells: a reverting or gas-burning receiver (for example an EIP-7702 delegation) only defers its own payment.

PROOF OF CONCEPT

P2_Adversarial.t.sol::test_grief_marketingRevertsOrBurnsGas: with gas-burning or reverting code on the marketing wallet, swap-back sells still settle at about 1.17M gas. P4_RewardPath.t.sol::test_I_strandedBnbSweptByNextGoodSend: once the receiver accepts again, the next swap-back pays out the stranded BNB too.

RECOMMENDATION

Nothing on the live contract. Future versions: pay out the swap's measured output and send fees through a pull pattern.

SMC AUDITS

SMC-I-10 INFO Open

Transfers to the router are fee-free and sweepable

CATEGORY

Fee, tax, and reflection mechanics

DESCRIPTION

Every transfer whose recipient is the router skips the fee block. The router normally holds no BTCD, but tokens sent to it by mistake stay there, and removeLiquidityETHSupportingFeeOnTransferTokens forwards the router's entire BTCD balance to whoever removes liquidity next.

IMPACT

User-error loss only: tokens sent to the router directly can be taken by anyone. The router held 0 BTCD at the fork block.

RECOMMENDATION

Users and integrators should never transfer BTCD to the router directly. Future versions: do not special-case the router in the fee gate.

SMC-I-11 INFO Open

Untracked liquidity leftover and fee rounding dust

CATEGORY

Arithmetic, overflow, and precision

DESCRIPTION

Each fee bucket is floored separately (fees * bucket / total), leaving at most 2 wei per taxed transfer unassigned, and the token half of each auto-liquidity add that the router does not use is not tracked. Both accumulate as balanceOf(this) minus pending; at the fork block that was 95,263,893.84 BTCD.

IMPACT

No loss. The amount is recoverable by anyone into burned liquidity through addLiquidityFromLeftoverTokens (SMC-I-08).

RECOMMENDATION

Nothing on the live contract. Future versions: assign the remainder to one bucket and track the liquidity leftover.

SMC AUDITS

SMC-I-12 INFO Open

Dividend tracker behaviour and code quality

CATEGORY

Best practice / code quality; token and supply logic

DESCRIPTION

The dividend accounting is the standard magnified-dividend-per-share scheme and holds up: corrections keep accrued rewards exact across transfers, burns and claims; the magnitude has about 4e20 headroom before overflow; and the processing loop is bounded by its 300,000-gas budget. Points to note: a holder whose balance drops below 1,000,000 BTCD leaves the auto-pay rotation but keeps what it earned, paid only on a manual claim(); 0x...dEaD is not excluded from dividends, so BTCD sent there earns BTCB that nobody can spend; swap-and-pop removal can make the rotation skip or repeat one holder per round, which is harmless because claims are idempotent; and the tracker's public distributeDividends accepts BTCB from anyone and credits the amount received. Template leftovers remain in comments and dead expressions.

IMPACT

No loss. Holders who sell to below the minimum without claiming leave rewards waiting; rewards sent to the dead address are lost to holders, a negligible share today.

PROOF OF CONCEPT

P5_FinalRun.t.sol::test_F5_04_worstCaseGasUnderBep652: with every holder due a payout and the swap-back armed, a sell uses 1,163,666 gas and process(300000) 346,970, both well under the BSC per-transaction cap. test_F5_05_burnAndDividends confirms burn and burnFrom keep withdrawable rewards exact.

RECOMMENDATION

Holders should call claim() on the token before selling below 1,000,000 BTCD and should burn with burn() rather than sending to 0x...dEaD. Future versions: exclude the dead address, pay out pending rewards when a holder leaves the rotation, and remove template leftovers.

SMC AUDITS

SMC-I-13 INFO Open

Permanent wallet and transaction limits

CATEGORY

Denial of service and gas griefing; integration safety

DESCRIPTION

The max wallet is 15,000,000,000 BTCD (3.45% of current supply) for every non-exempt recipient and can never be lifted. The max buy and max sell are 400,000,000,000 BTCD, close to the whole supply, so they have no practical effect. The trading switch is on and one-way.

IMPACT

No loss. A wallet near 15B BTCD cannot receive more by any route, which can surprise integrators that pool user balances in one address (custodians, vaults).

RECOMMENDATION

Integrators that pool balances should use several addresses. Future versions: make launch limits expire.

09

09 Remediation status

After the project addresses the findings, SMC Audits re-audits the changes and records the outcome here. The reviewed contracts are immutable (not proxies) and ownership is renounced, so on-chain code changes are not possible. SMC-L-01 and SMC-I-03 can be reduced by the project team through actions outside the contract (extending or burning the LP lock, locking or burning insider allocations); the rest is advisory, for holders, integrators and any future redeployment.

ID	SEVERITY	STATUS	VERIFIED FIX
SMC-L-01	LOW	Open	None. Lock owner can extend lock 1192506 or burn the LP before 13 January 2029.
SMC-L-02	LOW	Open	None (immutable). Mitigations in-contract (increaseAllowance / decreaseAllowance).
SMC-I-01	INFO	Acknowledged	Mitigated: powers neutralized by renounce to 0x0; verified live and in the admin history.
SMC-I-02	INFO	Open	None (immutable). Exemptions are permanent.
SMC-I-03	INFO	Open	None. Project can disclose and lock or burn insider allocations.

SMC AUDITS

SMC-I-04	INFO	Open	None. External dependency; restores itself with liquidity.
SMC-I-05	INFO	Open	None. Applies to future versions.
SMC-I-06	INFO	Open	None. Measured as unprofitable to sandwich.
SMC-I-07	INFO	Open	None. Applies to future versions.
SMC-I-08	INFO	Open	None. Holder-positive; applies to future versions.
SMC-I-09	INFO	Open	None (immutable).
SMC-I-10	INFO	Open	None. User guidance.
SMC-I-11	INFO	Open	None. Recoverable into burned liquidity.
SMC-I-12	INFO	Open	None. Holders can claim() at any time.
SMC-I-13	INFO	Open	None. Integration guidance.

Open items for completeness: the current holdings of the roughly 30 wallets that received pre-launch allocations were not fully traced because public archive RPCs rate-limited the log scan (SMC-I-03); this does not affect any severity. The EIP-7702 evidence is a fork simulation (code written to the marketing and holder addresses with vm.etch; no key was used), and Foundry does not apply BSC's 16,777,216 per-transaction gas cap, so the tests pass the cap explicitly as the call's gas. Balances, reserves and dividend figures are at block 127,820,893 (15 October 2026 16:45:53 UTC).

10

10 Conclusion

At the reviewed address, BTC Dragon (BTCDragon) is a generator-template fee token with a BTCB dividend tracker, a fixed 9% tax on trades through its PancakeSwap pair, a 1% transfer fee for holder rewards, an automatic swap-back and renounced ownership. The review found no critical, high or medium issue.

The renounce is complete: no owner, pending owner or tracker admin path remains, and the marketing wallet, the one privileged address left, is paid in a way that cannot block sells, even with EIP-7702 code and under BSC's per-transaction gas cap. The fee split, burn, rounding and dividend accounting work as written, and the tracker's processing loop stays far below the cap in the worst case we could build. The two low findings are the marketing wallet's control of 71.37% of the LP after 13 January 2029 (SMC-L-01) and the standard approve race (SMC-L-02); the thirteen informational items do not put user funds at risk in the deployed state.

The load-bearing condition is the LP lock. It holds until 13 January 2029, and after that the marketing wallet 0x4B2A841135e5d1b819eB34d4BB52A1E7aB006bC1 decides whether the pool's liquidity stays. Readers should check PinkLock02 lock 1192506 themselves as that date approaches, and should treat the pre-launch allocations described in SMC-I-03 as unlocked supply.

An audit reduces risk. It does not guarantee the absence of every vulnerability. Users and the project should treat this report as one part of a wider security practice.

11

11 Disclaimer

This report reflects the code reviewed at the specific commit or address stated in the scope, at the time of review. It does not cover changes made after the review, external dependencies outside the stated scope, off-chain components, or economic and market conditions. This report is not financial, investment, or legal advice. SMC Audits provides no warranty, express or implied, and accepts no liability for any loss arising from the use of the audited code. A clean report is not a guarantee of safety.

SMC AUDITS · CONFIDENTIAL · SMCAUDITS.COM