**1) Vulnerability**

This vulnerability typically arises when the approve function is misused or fails to provide safeguards against over-approving tokens, leading to unintended behaviors.

**Potential Vulnerability**

* Unintended Token Spending: If an attacker malicious contract or external interaction exploits this vulnerability, tokens could be drained from the owner's account without explicit approval for new transactions.
* Users may approve a high allowance (or even type(uint).max), which, if left unchecked, could result in token theft if the spender is compromised.

**2) Exploitation**

1. Infinite Approval: A user approves a large or unlimited allowance for a spender
2. Compromised Spender: If the spender's address is compromised or malicious, they could repeatedly call transferFrom to drain the user’s balance.
3. No Warning on Overwrites: The approve function does not warn or prevent allowance overwrites, allowing unintended approvals without user confirmation.

**3) Proposed Mitigation**

The GalacticToken contract could include safeguards to mitigate the risks associated with token approvals.

1. Use Safe Approvals: Implement a two-step approval process where the allowance must be explicitly reduced to zero before being updated.
2. Set Maximum Allowances Cautiously: Encourage users to approve only the required amount for a single transaction rather than unlimited allowances.
3. Monitor Allowance Usage: Implement event logs or transaction monitors to detect unusual activity related to transferFrom calls.