Security Research Labs

Xcavate baseline security assurance

Baseline security audit of Xcavate's blockchain

v1.1 – December 8th, 2025

Aarnav Bos	aarnav@srlabs.de
Cayo Fletcher-Smith	cayo@srlabs.de
Gabriel Arnautu	gabriel@srlabs.de
Marc Heuse	marc@srlabs.de

Abstract. This report is a summary of a thorough, independent security baseline assurance review of the *Xcavate blockchain.*

Xcavate is a blockchain-based platform that democratizes real estate investment by enabling fractional property ownership through tokenization. Users can list, buy and sell property on-chain through its marketplace implementation.

The focus of the audit was on discovering logical errors and runtime configuration issues, that could lead to market manipulation, stealing of assets, denial of service, and fee inconsistencies.

Twenty-seven security issues were identified during the audit: ten high severity, ten medium, six low, and one informational. The issues consisted of incorrect extrinsic weighting, front-running, sensitive state-transitions, circumventing access control, and missing consistency checks. Additionally, core design flaws, such as unimplemented mechanisms, were highlighted and acknowledged by Xcavate.

We highly recommend another baseline security review before launch, due to the large number of findings and low code maturity.

1 Overview

1.1 Motivation & Scope

Xcavate is a blockchain-based platform that democratizes real estate investment by enabling fractional property ownership through tokenization. The platform transforms traditional real estate transactions into a transparent, decentralized system where properties are converted into tradeable digital tokens, allowing investors to purchase fractional shares rather than requiring full property ownership. The platform handles property listing and tokenization, manages legal compliance through registered lawyers and Special Purpose Vehicles (SPVs), facilitates secure fund escrow and settlement, and enables ongoing property management through democratically selected letting agents. Token holders exercise governance rights proportional to their ownership stake, voting on critical decisions including property expenses, agent performance, and eventual property sales.

The audit was guided by a threat model which we developed in conjunction with the Xcavate team. The audit team utilized static analysis, dynamic analysis via fuzzing, and manual code review. Due to the complexity of the pallets and logic, the team decided to prioritize manual review over fuzzing, which in turn led to a large number of findings.

Our threat model's core themes were MEV exploitation, invalid handling of assets or funds in the marketplace pallet, bypassing access control mechanisms, and denial of service.

The scope was developed in collaboration with the Xcavate team. Key areas of scrutiny include pallet-marketplace, pallet-xcavate-whitelist, pallet-property-governance and the runtime configuration. The team performed the majority of the audit on an initial commit, before transitioning to a new version for the audit of pallet-bucket, this is reflected in Table 1.

Repository	Component(s)	Commit hash reviewed
xcavate-node-audit [1]	marketplace	82ceef10acf954bf12a658c9bde5f997a9b051d6
	pallet-bucket	77c79a703d2d23c65f75f9cd7394ebb6dab296a3
	property-governance	82ceef10acf954bf12a658c9bde5f997a9b051d6
	property-management	82ceef10acf954bf12a658c9bde5f997a9b051d6
	real-estate-asset	82ceef10acf954bf12a658c9bde5f997a9b051d6
	regions	82ceef10acf954bf12a658c9bde5f997a9b051d6
	xcavate-whitelist	82ceef10acf954bf12a658c9bde5f997a9b051d6
	runtime configuration	82ceef10acf954bf12a658c9bde5f997a9b051d6

Table 1. Scope overview

2 Findings

2.1 Findings summary

The audit found 27 issues: 10 high severity, 10 medium, 6 low, and 1 informational. The core themes of the issues concerned incorrect extrinsic weighting, front-running, sensitive state-transitions, circumventing access control, and missing consistency checks.

In the marketplace pallet, the core pallet of Xcavate, the team found three notable issues. One where an attacker could trick a user into accepting a fraudulent offer through front-running, another where a user could still conduct sensitive actions despite access revocation and one where an attacker could prevent property listing under certain circumstances.

The audit team additionally found three weight related issues, opening avenues for denial of service through spamming underweight transactions.

The audit also surfaced two core flaws in Xcavate's design. One, in slashing, which is a deterrence mechanism for misbehavior. Due to the lucrative aspect of real estate and large number of actors in the process, the incentives are high for misbehavior. The team found that deterrence was insufficient and did not scale as actors becomes increasingly involved, for example within a region. This was acknowledged by Xcavate as an intended mechanism to be implemented before release although remained insufficient at the time of this review. The second in the pallet-xcavate-whitelist, which implements access control mechanisms for users after they have completed KYC/KYB. The pallet can restrict actions a user can perform if they are suspected of fraudulent or suspicious activity. However, a large amount of activity, such as transferring certain funds or assets, remains possible despite access revocation. These flaws were acknowledged by the Xcavate team to be fixed in future versions.

As of December 8th, 2025: 21 issues have been remediated in collaboration with the Xcavate development team, and 6 issues remain unresolved. Unresolved issues have been acknowledged and are in the process of being remediated by Xcavate.

3 Evolution suggestions

Due to the large number of security issues and low code maturity, Security Research Labs recommends implementing three core suggestions.

Implement code review and threat modelling: Threat modeling and code review for all new features and before integration promotes better code security. Catching bugs early in the development cycle would enable Xcavate to shift left and focus on secure design instead of mitigation. This may be implemented as a continuous security review, pair-programming or PR review.

Use and maintain in-line documentation: Xcavate's codebase currently lacks sufficient documentation across extrinsics, storage and structures and supporting functions. Implementing extensive documentation simplifies code review, development, and developer onboarding.

Conduct another audit: In addition to implementing and auditing the fixes for all open issues, we strongly recommend performing an additional security audit before going live. Due to the large number of findings, complex business logic, and low code maturity, it is likely that there are security or business logic issues that were not found during the audit.

4 References

[1] [Online]. Available: https://github.com/XcavateBlockchain/xcavate-node-audit.