

ZK Circuit Security Checklist

Use this checklist to assess the security posture of your ZK circuits. We recommend running through this before any audit engagement.

Project Name: _____

Date: _____

Reviewer: _____

PHASE 1: Planning & Design

- ☐ **Define a clear threat model** Identify potential attackers, attack vectors, and what assets need protection
- ☐ **Define circuit inputs/outputs and public values** document which signals are public vs. private to prevent soundness/privacy issues
- ☐ **Plan development**
 - ☐ Decide on proof system
 - ☐ Integration considerations with on/off chain components – how the circuit will be verified (on-chain/off-chain)?
 - ☐ Check existing, audited libraries that provide cryptographic guarantees required for the circuit (signatures, hash functions)

PHASE 2: Implementation & Testing

- ☐ **Write tests**
 - ☐ Unit tests on constraints: verify that valid witnesses satisfy the circuit (completeness tests)
 - ☐ Negative tests: invalid witnesses fail (soundness tests)
 - ☐ Edge case coverage (0 values, max field elements, empty sets, boundary conditions)
- ☐ **Review circuit constraints for over-constraining**
 - ☐ Negative tests: invalid witnesses fail (soundness tests)
 - ☐ Every required invariant is enforced
- ☐ **Review circuit constraints for under-constraining**
 - ☐ No missing constraints
 - ☐ No ambiguity in endianness, padding, or field reductions
 - ☐ No unsafe optimizations that remove necessary checks
- ☐ **Third party assessment**
 - ☐ Book security audit with ZK-specialized firm
 - ☐ Make valid fixes to the code based on findings
 - ☐ Re-audit if significant changes were made

Once all these have been checked off, it is possible to publish the circuit and can feel confident in its security.



Need help with your ZK circuit audit? Nethermind Security specializes in ZK circuit audits across Circom, Noir, Cairo, SP1, and RISC Zero.

 nethermind.io/zk-audits

 security@nethermind.io