BEISPIELBERICHT

Contract-Audit | PRO | ANALYSEBERICHT

ERGEBNIS: höchste Befundschwere HOCH, zu prüfen — Heuristischer Kandidat aus Quellcode: Ungeschützte Rechteänderung (jeder kann Eigentümer oder Rolle ändern) — Funktion DemonstrationOnly.setOwner.

HOCH

Was jetzt zu tun ist: Wenden Sie die vorgeschlagene Korrektur an (Diff im Abschnitt der wichtigsten Befunde), kompilieren Sie den Vertrag, ergänzen Sie einen Test, dass ein Konto ohne Berechtigung diese Funktion nicht aufrufen kann, und starten Sie die Analyse vor dem Deployment erneut.

Demonstrationsdaten: ein eigener Testvertrag, der nie in einem Netzwerk bereitgestellt wurde. Dies ist eine Kurzfassung eines Audit-Pro-Berichts — der vollständige Bericht enthält zusätzlich Ausführungsumfang, Messwerte, Datenquelle und die Dokumentkennung.

WICHTIGSTE BEFUNDE

HOCH / Zu prüfen: Heuristischer Kandidat aus Quellcode: Ungeschützte Rechteänderung (jeder kann Eigentümer oder Rolle ändern) — Funktion DemonstrationOnly.setOwner

Beschreibung: Statisches Quellsignal Ungeschützte Rechteänderung (jeder kann Eigentümer oder Rolle ändern) — Funktion DemonstrationOnly.setOwner (UNPROTECTED-PRIVILEGED-WRITE) in DemonstrationOnly.sol, Zeilen 7-7. Dies ist ein Prüfhinweis; er bestätigt weder einen Fehler noch Erreichbarkeit, die Zuordnung zum bereitgestellten Bytecode oder einen Exploit.

Empfohlene Korrektur: die Funktion mit einer Zugriffskontrolle absichern (z. B. Modifier onlyOwner aus Ownable, zweistufige Eigentumsübertragung mit Ownable2Step oder eine AccessControl-Rolle), die Nulladresse ablehnen (require(next != address(0))) und testen, dass ein nicht berechtigtes Konto Eigentümer oder Rolle nicht ändern kann.

Vorgeschlagene Korrektur — Diff zu Ihrer Datei (vor dem Deployment kompilieren und testen)

```
--- a/DemonstrationOnly.sol
+++ b/DemonstrationOnly.sol
@@ -5,6 +5,7 @@
  address public owner;
  constructor() { owner = msg.sender; }
  function setOwner(address next) external {
+ require(msg.sender == owner);
  owner = next;
  }
  }
```

Als erste Anweisung der Funktion wurde eine Aufruferprüfung mit der vorhandenen Zustandsvariable "owner" dieses Vertrags ergänzt. Prüfen Sie, ob dies die vorgesehene Rolle ist, kompilieren Sie den Code und testen Sie ein nicht berechtigtes Konto.

BEISPIELBERICHT

Contract-Audit | PRO | ANALYSEBERICHT

WAS ALS NÄCHSTES ZU PRÜFEN IST

Zu prüfen: Prüfen Sie den Kandidaten anhand des genauen Codes, Zustands und der referenzierten Bedingungen, bevor Sie ihn als bestätigtes Problem behandeln.
Zu prüfen: Externe Interaktionen: Prüfen Sie den erfassten Umfang und die Nachweise dieser Kontrolle. Ein ungeklärter Status beweist kein Problem und garantiert nicht, dass zusätzliche Daten das Ergebnis klären.

GRENZEN DIESES BERICHTS

1. Dies ist eine automatisierte Analyse des gelieferten Codes, kein manuelles Audit und kein Sicherheitszertifikat.
2. Wir haben den Code nicht mit einer im Netzwerk bereitgestellten Version verglichen; ist der Vertrag bereitgestellt, vergleichen Sie den Code vor einer Entscheidung.
3. Keine Befunde in einem Bereich beweisen nicht, dass der Code sicher ist — die Regeln erkennen bekannte Fehlermuster.
4. Es wurde kein Angriff auf einen bereitgestellten Vertrag ausgeführt und keine finanzielle Auswirkung berechnet.
5. Die vollständige Liste der erfassten Einschränkungen (26) steht im technischen Anhang der TXT-Datei und in der JSON-Datei.
Vollständige Befunde, Nachweise, Messwerte und Umfangszeilen stehen in der privaten JSON-Datei. TXT enthält eine ausführliche Darstellung.