

TokenX — Internal Security Report

Для внутреннего использования: подробности, PoC, тесты и патчи. Замените заполнители <.

1. Технические метаданные

Repo: <repo_url>
Commit before: <hash-before>
Commit after: <hash-after>
Compiler: Solidity 0.8.20, optimizer runs = 200
Инструменты: Slither vX, Mythril vY, Echidna vZ, Hardhat vX, Foundry vX
Test coverage: unit 96% (report: <link>), fuzz 5,000 inputs

2. Executive Summary

Объект аудита: ERC-20 TokenX. Цель — проверить безопасность функций, ограничение эмиссии.
Резюме: 0 Critical, 1 High (исправлено), 0 Medium, 2 Low/Informational (учтено). Контрак

3. Рейтинг находок (сводно)

ID	Ter	Описание	Статус	Priority	TKNX
HIGH-001	High	Unlimited Minting	Fixed	P1	TKNX-HIGH-001
LOW-002	Low	Redundant SafeMath	usage Recommended	P3	TKNX-LOW-002
INFO-001	Info	Missing admin events	Recommended	P4	TKNX-INFO-001

4. Подробные находки

TKNX-HIGH-001 — Unlimited Minting Risk (Fixed)

Описание: `mint()` была доступна владельцу без ограничения `MAX_SUPPLY`.

Impact: Потенциальная инфляция и полная потеря стоимости токена.

PoC: владелец вызывает `mint(attacker, very_large_amount)`.

Remediation (внесено):

Добавлен `MAX_SUPPLY`, замена управления на `AccessControl` с `MINTER_ROLE` и проверкой cap
solidity

```
uint256 public constant MAX_SUPPLY = 1_000_000 * 10**18;
```

```
function mint(address to, uint256 amount) external onlyRole(MINTER_ROLE) { require
```

Рекомендуется возможность `renounceRole(MINTER_ROLE)` или `finishMinting()` после `initia`
Проверочные тесты: `test_mint_within_cap`, `test_mint_exceeding_cap_reverts`, `fuzz asserti`

TKNX-LOW-002 — Redundant SafeMath usage (Recommended)

Описание: Используется `SafeMath` в Solidity 0.8.x; лишние вызовы.

Remediation: Удалить `SafeMath`, использовать встроенную проверку; добавить `gas benchmark`

INFO-001 — Missing admin events

Описание: Нет событий для ключевых административных действий.

Remediation: Добавить events: RoleGranted, RoleRevoked, FinishMinting, MinterRenounced

5. Patch / Diff (ключевые изменения)

Ветка/PR: <branch/patch-name>

Основные изменения: добавлен MAX_SUPPLY, введён AccessControl/MINTER_ROLE, обновлены т

Пример теста (Hardhat/Mocha):

js

```
it("reverts when mint exceeds cap", async function() { await expect(token.connect(minter).mint(user.address, MAX_SUPPLY.add(1))).to.be.revertedWith("TokenX: cap exceeded");});
```

6. CI / Tests / Gas

CI: GitHub Actions – unit tests + Slither + gas-report.

Gas report: transfer/mint – измерено до и после; PR <link> содержит результаты.

7. Доказательства и воспроизводимость

Commit before: <hash-before>

Commit after (post-fix): <hash-after>

Coverage report: <link>

Slither output: <link>

Artifacts: ABI, flattened.sol, deployment bytecode (в attachments)

8. Рекомендации по процессам

Governance: MINTER_ROLE + Timelock (48h) или multisig после distribution.

CI: Slither/MythX в pre-merge; nightly Echidna fuzz.

Bounty: разместить на Immunefi/HackerOne.

9. Action Items

Merge PR <link> и создать релиз v1.0.1 – Owner: Smart Contracts –

Deadline: <date>

Внедрить multisig и Timelock – Owner: DevOps – Deadline: `

10. Приложения

Flattened contracts (post-fix), тест-логи, Slither/Mythril outputs, gas report.