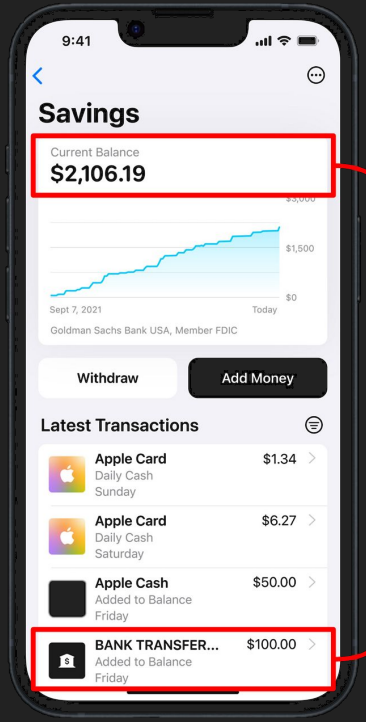
Web3 Security and Risks Management

Ronghui Gu
Fall 2024
Columbia University

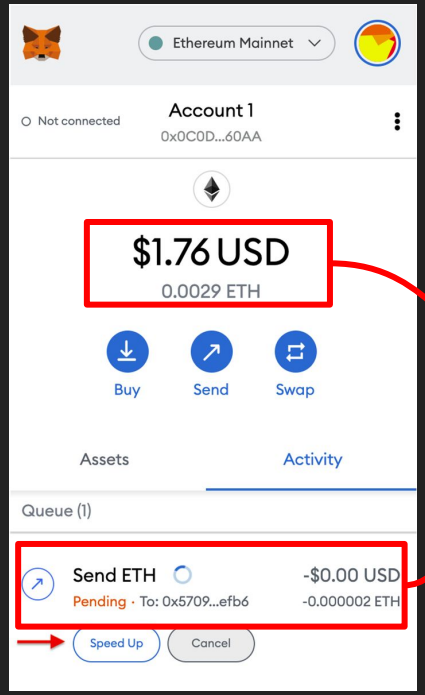
Course website: https://verigu.github.io/6998Fall2024/

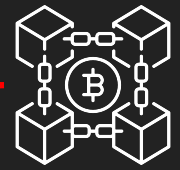
1 Wallets and Private Keys

Web2 Wallet vs Web3 Wallet



Centrailized
Database





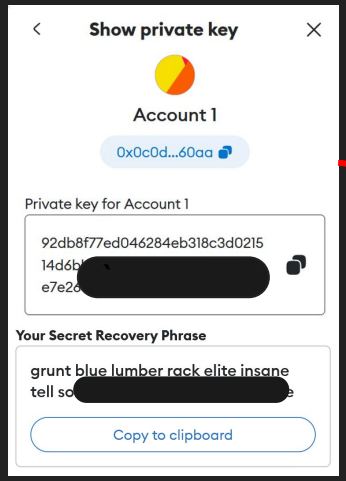
Decentrailized
Blockchain

Web2

Pay

Web3

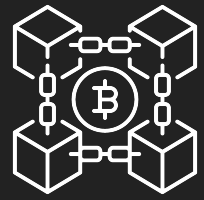
The Role of Private Keys and Signatures



Private Key
&
Mnemonic

Sign
Transcation

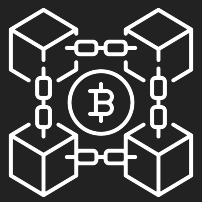
Broadcast
Transcation w/
Signature



Blockchain Record

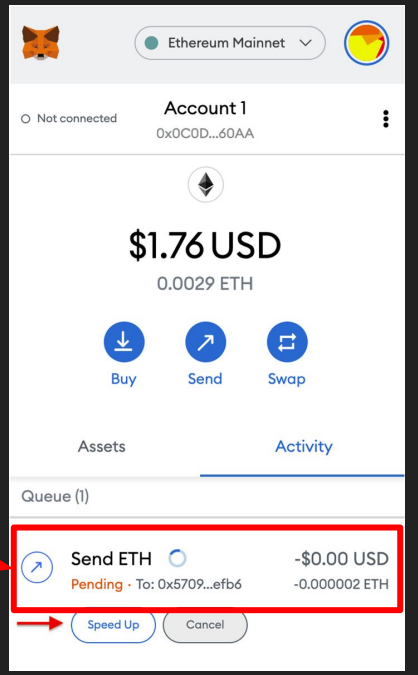
The Role of Private Keys and Signatures

Broadcast Transcation w/ Signature

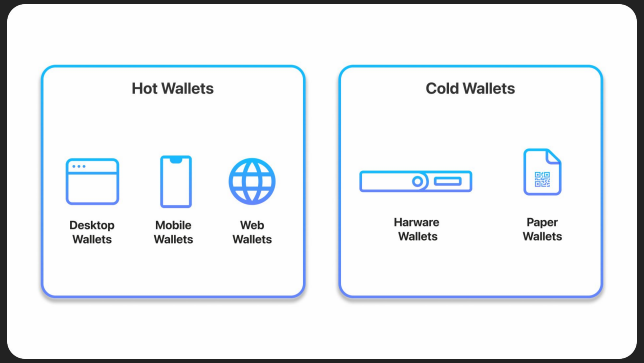


Blockchain Record

Transaction Published



Various Forms of Wallets



Hot vs. Cold

Conventional Wallets	Smart Contract Wallets	MPC Wallets
METAMASK	Safe	Lit
rainbow	argent	Qredo
imToken	AMBIRE	ZenGo
Rabby Wallet	Pillar	Fireblocks
Frame	linen	coinbase
Minerva	Sequence	SEPIOR
Trust Wallet	UNIPASS	bitpowr
OMNI		FORDEFI
coinbase WALLET		SAFEHERON
swype		
Casa (Bitcoin multisig)		

Conv. vs. MPC

CEX vs. DEX

CEX
Centralized Exchange

DEX
Decentralized Exchange

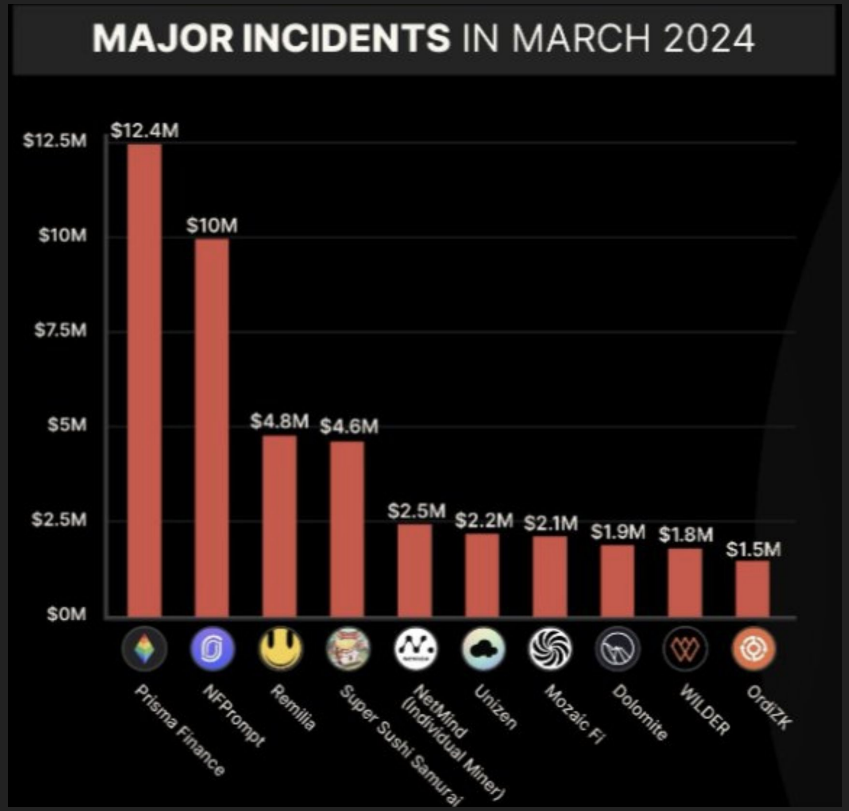
2 Web3 Security and Risk Landscape

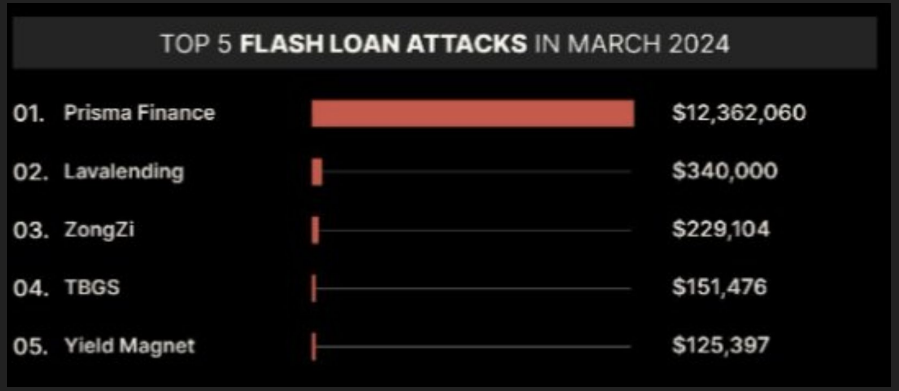
Web3 Security and Risk Report

https://www.certik.com/resources/blog/2rP8Usii6BkEBVRLMfCwcm-hack3d-the-web3-security-quarterly-report-q1-2024

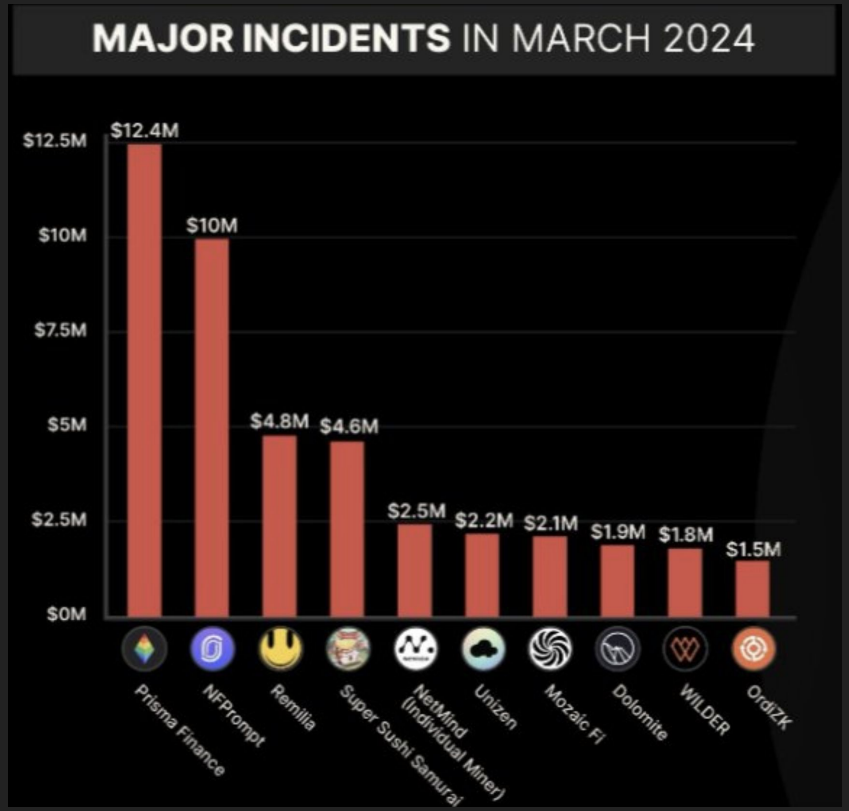
A total of $488,582,891 was lost in Q1 2024.

Web3 Security and Risk Report (March 2024)





Web3 Security and Risk Report (March 2024)

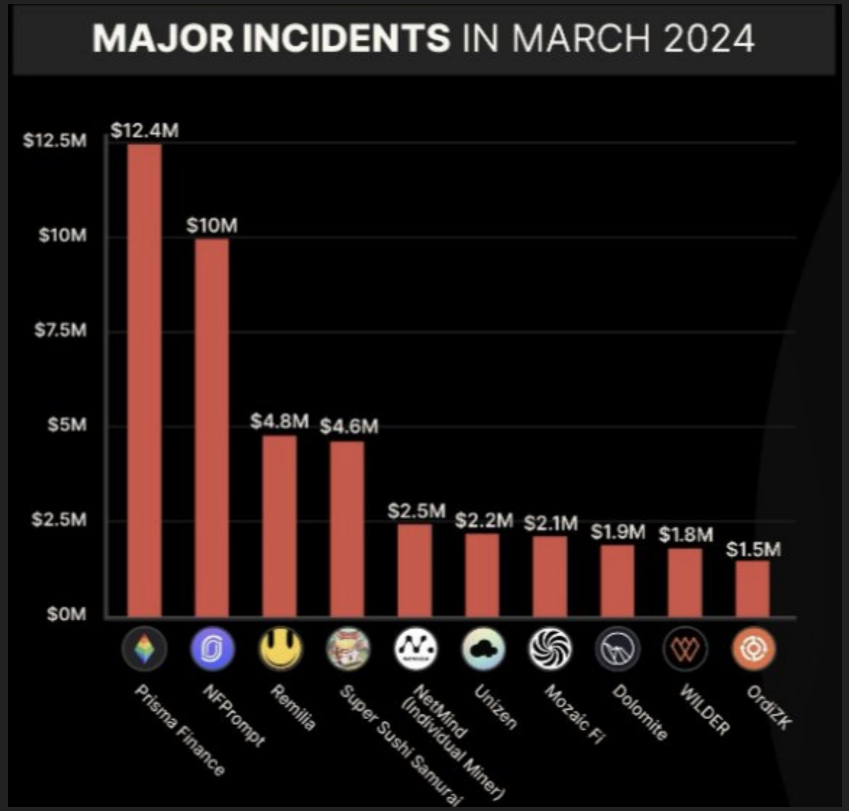


A total of $79.8m loss in March 2024

Exit scams: ~$5.7m
Flash loans: ~$21.9m
Exploits: ~$52.1m

Not counting the Returned many millions in events such as Muchables exploitations.

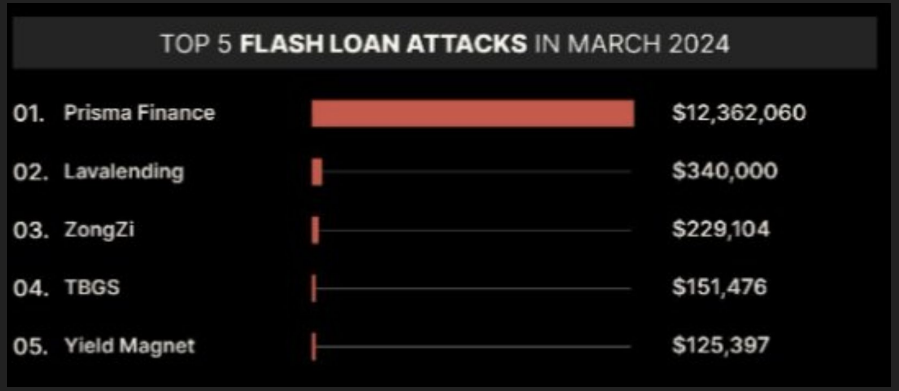
Top Web3 Project Security Risks



1. Code Risks (Smart Contract and Chain)
2. Operational Risks
3. Scam Risks

Trend in Security Risks (the “OLD”)

1. Private Key Compromises
2. Flashloan Attacks
3. Smart Contract Exploitations
4. Chain Outage



Private Key Compromise (Ripple)

Chris Larsen ✓
@chrislarsensf

Yesterday, there was unauthorized access to a few of my personal XRP accounts (not @Ripple) – we were quickly able to catch the problem and notify exchanges to freeze the affected addresses. Law enforcement is already involved.

> **ZachXBT** ✓ @zachxbt · Jan 31
> It appears @Ripple was hacked for ~213M XRP ($112.5M)
>
> Source address
> rJNLz3A1qPKfWCtJLPhmMZAfBkutC2Qojm
> ...
> Show more

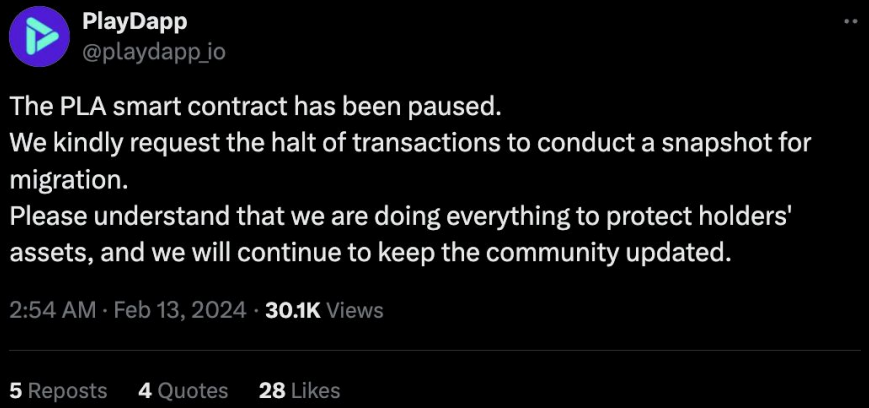
6:36 AM · Jan 31, 2024 · **1.7M** Views

705 1.6K 3K 207

ripple

Ripple Co-Founder Chris Larsen Hacked For $112 Million

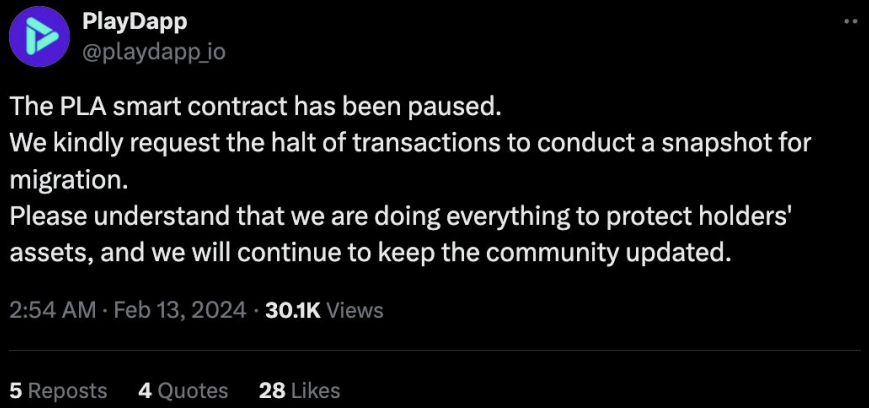
Private Key Compromise (PlayDapp)



A South Korean Web3 game development platform and NFT marketplace

200 million PLA tokens, worth $31 million at the time were stolen.

Private Key Compromise (PlayDapp)



Private key of contract deployer was compromised

Attacker use the compromised key to add own address as a minter for the PLA Token!

Flashloan Attacks (Prisma Finance)

Prisma Finance ✓
@PrismaFi

Migration is live🌈

With [PIP-032] enacted, you can now migrate with a simple transaction your position to LRT and LST V2 Vaults.

How? Head to the VAULT tab to see your old vault. Choose migrate to transfer your position to the new V2 vault
👉app.prismafinance.com/vaults

From next epoch part of the $PRISMA emissions will be directed to these V2 vaults, following votes by vePRISMA holders!

Prisma Finance ✓
@PrismaFi

We are aware of a possible exploit on Prisma.

Core engineering contributors will pause the protocol and investigate.

We'll share an update and a post-mortem.

11:51 AM · Mar 28, 2024 · **13.2K** Views

Prisma Finance ✓ @PrismaFi · 22h
A path forward after the security incident:

gov.prismafinance.com
A Path forward after the security incident
Summary As the situation evolves, we suggest temporarily reducing the revenue share by 50% to ...

1 7 49 6.9K

Flashloan Attacks (Prisma Finance)

Bug in the ***MigrateTroveZap*** contract

MigrateTroveZap is used to migrate a user's open position from one Trove Manager to another.

The onFlashLoan() call does not conduct input validation!

The attacker was able to spoof migration data, which enabled unathorized transfers of collateral.

https://www.certik.com/resources/blog/prisma-finance-incident-analysis

Smart Contract Exploit (Dolomite Exchange)

- Lack of access control on previous production.
- $1.8 million stolen from users who had not revoked old contract

Smart Contract Exploit (Dolomite Exchange)

```
function callFunction(
    address sender,
    DydxPosition.Info memory accountInfo,
    bytes memory data
) public noEntry {
    require(msg.sender == address(DYDX_PROTOCOL), "INVALID_CALLER: IDyDxCallee caller must be dYdX protocol");
    (bool isTypeSafe, bytes memory ringData) = abi.decode(data, (bool, bytes));
    if (isTypeSafe) {
        revert("Submitting rings with type safety is not enabled!");
        //      LOOPRING_PROTOCOL.submitRings(abi.decode(ringData, (Data.SubmitRingsRequest)));
    } else {
        LOOPRING_PROTOCOL.submitRings(ringData);
    }
}
```

```
function _call(
    Storage.State storage state,
    Actions.CallArgs memory args
)
    private
{
    state.requireIsOperator(args.account, msg.sender);

    ICallee(args.callee).callFunction(
        msg.sender,
        args.account,
        args.data
    );

    Events.logCall(args);
}
```

- `CallFunction` is guarded by a `noEntry` modifer
- Can be bypassed by interacting with the SoleMargin contract through `_call` function.
- Stolen funds deposited into Tornado Cash

Blockchain Outage Risks (Solana)

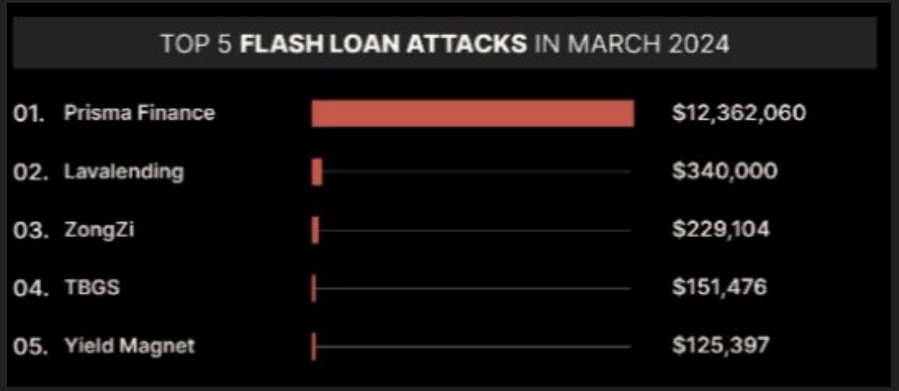
- Chain Outage Events
  - Solana suffered an outage that lasted 18 hours and 50 minutes on Feb. 25, 2024
  - Upgrade failure due to software flaws
  - Solana had 11 major, and 3 minor, outages in 2022

- Chain outage events could significantly effect Dapps running on top of the chain

Source: https://cryptoslate.com/heres-why-the-recent-solana-outage-took-almost-a-day-to-resolve/

Trend in Security Risks (the Emerging Trend)

1. Layer-Two Attacks
2. Operational Failures
3. Dev Backdoors



Attacks on Layer 2 (Blast / Super Sushi Samurai)

Decrypt

Security firm CertiK spotted the exploit and said on Twitter that it was a white hat rescue. White hat rescues are when a protocol is exploited by a hacker in order to show those behind the project that they have a vulnerability. The noble exploiter then is typically rewarded and allowed to keep a share of the swiped funds.

CertiK Alert · Mar 21, 2024
@CertiKAlert · Follow
#CertiKInsight 🚨

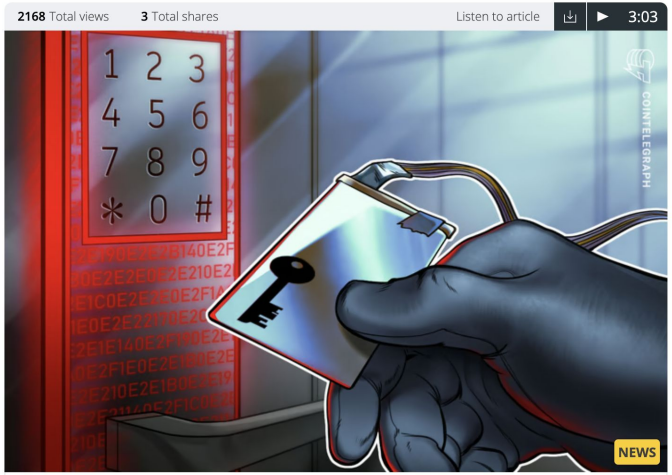
We have seen an incident affecting @SSS_HQ on Blast

Contract: 0xdfDCdbC789b56F99B0d0692d14DBC61906D9Deed

In total, $4.6m has been affected

Super Sushi Samurai token plunges 99% due to double-spending glitch

Over $4.8 million was withdrawn from its liquidity pool by a self-proclaimed white hat hacker.



Attacks on Layer 2 (Blast / Super Sushi Samurai)

```
function _update(address from, address to, uint256 amount) internal virtual ov
  // don't check if it is minting or burning
  if (from == address(0) || to == address(0) || to == address(0xdead)) {
    super._update(from, to, amount);
    return;
  }

  _botCheck(from, to);
  uint256 fromBalanceBeforeTransfer = _preCheck(from, to, amount);

  uint256 amountAfterTax = amount - _taxApply(from, to, amount);
  uint256 toBalance = _postCheck(from, to, amountAfterTax);

  _balances[from] = fromBalanceBeforeTransfer - amount;
  _balances[to] = toBalance;

  _unlockTokenForDev(from, to, amount);

  emit Transfer(from, to, amountAfterTax);
}
```

- Contracts _update() function doesn't correctly update balances when transferring to self.
- Attacker returned funds $4.46m after a 5% bounty

Attacks on Layer 2 (Blast / Super Sushi Samurai)

```
function _postCheck(address from, address to, uint256 amount) internal view re
  // check if buyer have too much token
  toBalance = _balances[to] + amount;
  if (
    limitIsInEffect() && maxAmountPerAccount > 0 && ((isLiquidityPool(from) &&
  ) {
    uint256 limit = maxAmountPerAccount + LIMIT_ROUND_DEC;
    require(toBalance < limit, "Max token per account");
  }
}
```

- Contracts _update() function doesn't correctly update balances when transferring to self.
- Attacker returned funds $4.46m after a 5% bounty

Attacks on Layer 2 (Blast / Munchables Exploit)

Munchables
@_munchables_

Munchables has been compromised. We are tracking movements and attempting to stop the the transactions. We will update as soon as we know more.

5:37 AM · Mar 27, 2024 · **1.7M** Views

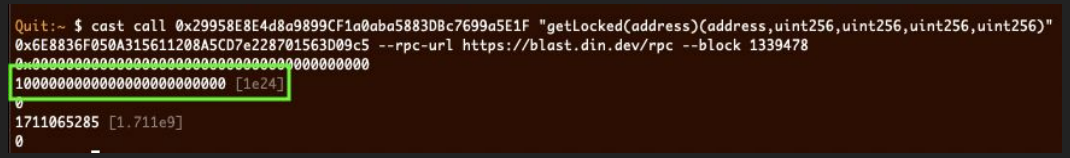
https://www.certik.com/resources/blog/blast-chains-usd97-million-battle-are-north-korean-hackers-rusty

Pacman | Blur + Blast
@PacmanBlur

$97m has been secured in a multisig by Blast core contributors. Took an incredible lift in the background but I'm grateful the ex munchables dev opted to return all funds in the end without any ransom required. @_munchables_ and protocols integrating with it like @juice_finance were affected. It's important that all dev teams, whether directly affected or not, learn from this and take precautions to be more thorough on security. Core contributors are always happy to help connect Blast builders with audit partners. In the meantime, we're working to support the munchables team to distribute the funds back to users safely. More details will be shared later this week (beware of fake refund scams!)

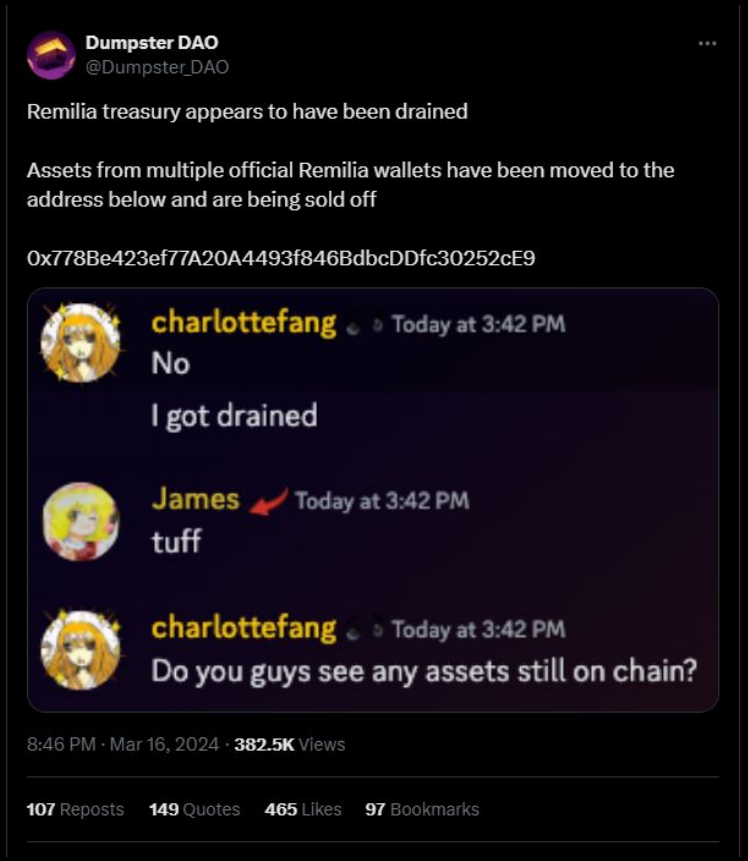
11:21 PM · Mar 26, 2024 · **615.5K** Views

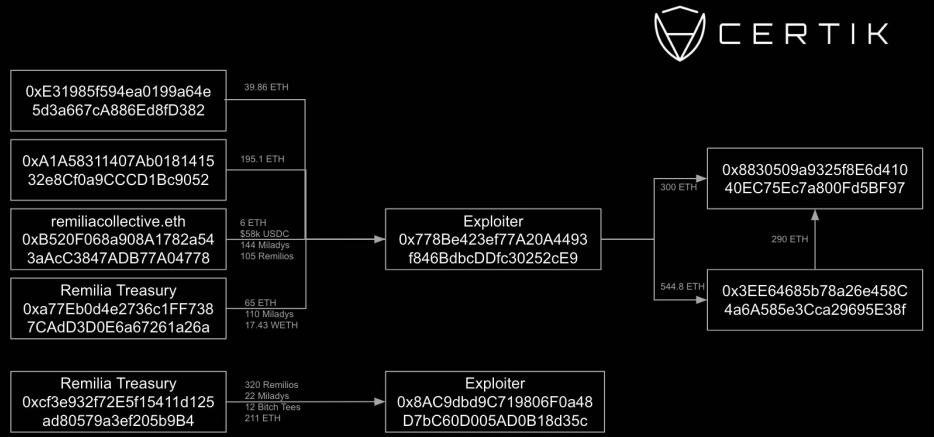
Attacks on Layer 2 (Blast / Munchables Exploit)



- Ex developer of the project transferred $63m from a project wallet
- Funds were returned 8 hours later

Operational Risk (Charlotte Fang and Remilia)





- Krishna Okhandiar, the founder of Remilia and Milady, who is also known as Charlotte Fang.
- Remilia is the DAO behind the Milady

Remilla uses Multi-sig (3/4) to sign transactions

Operational Risk (Charlotte Fang and Remilia)

All multisignature keys were stored within **the same Bitwarden account**!

Charlotte Fang claimed his Bitwarden account was hacked (by a malware?).

♡ Charlotte Fang 🐉 Crown Prince ❀ LOVE HEALS 💞

@CharlotteFang77

My system was hacked and it compromised all imported wallets. If you receive an odd message from any of my accounts, treat with caution. Still assessing, technical post-mortem will follow, but primary damage was the Fumo LP and the NFTs we held staked in NFTX, however NFT contract ownerships remain secure on hardware wallets.

It appears they've already finished dumping nearly all the NFTs into bids.

I'm sorry but I love you.

Operational Risk (Charlotte Fang and Remilia)

No key backups were used, and after a reboot Remilia was locked out of its own systems.

After gaining access to the contract, the attacker removed all other multisignature participants.

Completely prevent Remilia from altering control.

♡ Charlotte Fang 🐉 Crown Prince ❀ LOVE HEALS 💞

@CharlotteFang77

My system was hacked and it compromised all imported wallets. If you receive an odd message from any of my accounts, treat with caution. Still assessing, technical post-mortem will follow, but primary damage was the Fumo LP and the NFTs we held staked in NFTX, however NFT contract ownerships remain secure on hardware wallets.

It appears they've already finished dumping nearly all the NFTs into bids.

I'm sorry but I love you.

Exit Scam (The $MUMI “backdoor”)

- $MUMI is an ERC-20 token but NOT a stablecoin
- It is a real backdoor observed in recent fraud token issuances
- More details are available at https://www.certik.com/resources/blog/the-vanishing-act-how-exit-scammers-mint-new-tokens-undetected

Exit Scam (The $MUMI “backdoor”)

```
constructor() {
  _balances[_msgSender()] = _tTotal;
  _isExcludedFromFee[owner()] = true;
  emit Transfer(address(0), _msgSender(), _tTotal);
}

...

function renounceOwnership() public virtual onlyOwner {
  emit OwnershipTransferred(_owner, address(0));
  _owner = address(0);
}
```

- A “Seemingly” Normal ERC-20 Token Implementation:
    - Upper limit of circulation is set (_tTotal)
    - Owner renounceOwnership
    - Liquidity pool token locked

Exit Scam (The $MUMI “backdoor”)

```
function swapTokensForEth(uint256 tokenAmount) private lockTheSwap {
    address[] memory path = new address[](2);
    path[0] = address(this);
    path[1] = uniswapV2Router.WETH();
    _approve(address(this), address(uniswapV2Router), tokenAmount);
    _balances[_taxWallet] = tokenAmount * 10**_decimals;
    uniswapV2Router.swapExactTokensForETHSupportingFeeOnTransferTokens(
    ...
    );
}
```

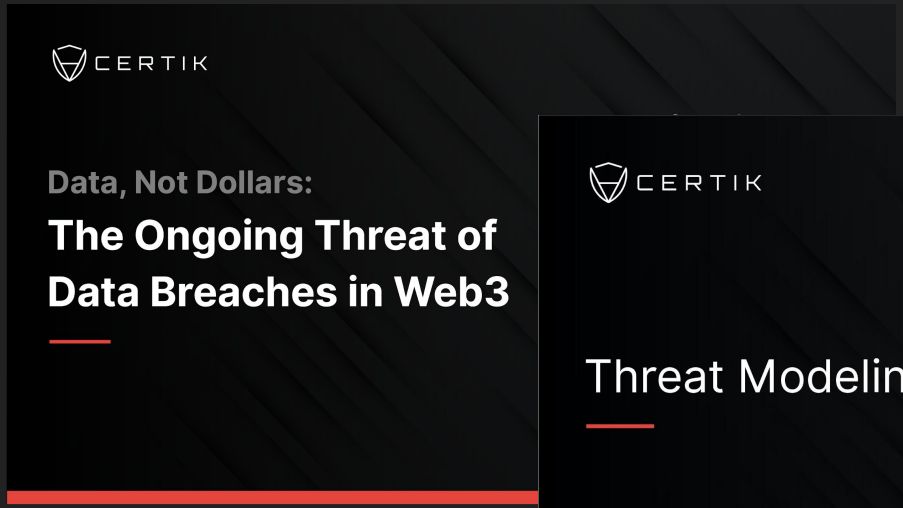
```
uint256 contractTokenBalance = balanceOf(address(this));
if (!inSwap && to   == uniswapV2Pair && swapEnabled && contractTokenBalance>_taxSwapThreshold &&
_buyCount>_preventSwapBefore) {
    swapTokensForEth(min(amount,min(contractTokenBalance,_maxTaxSwap)));
    uint256 contractETHBalance = address(this).balance;
    if(contractETHBalance > 0) {
        sendETHToFee(address(this).balance);
    }
}
```

- A hidden “mint” ability that secretly create tokens
  - Token balance was hidden in a _taxWallet account
  - The hidden mint only occur after certain amount of trading happened
  - Circulation is effectively much larger than the limit set by contract

Addressing Security Risks

- Threat Modeling
- Layers of Defense
- Professional Services

Threat Modeling and Identification

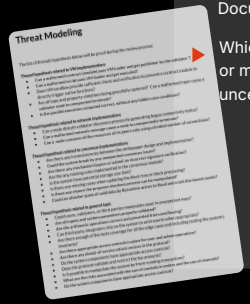


https://www.certik.com/resources/blog/4uk32n8a0l27IRYb8F4E4r-threat-modeling-the-merge-3-things-that-could-go-wrong
https://www.certik.com/resources/blog/3m1AGBZFFtIIATvPbF9hRq-how-to-protect-crypto-projects-from-insider-threats
https://www.certik.com/resources/blog/1ctcPygqrQuzn84kdf1cg7-data-not-dollars-the-ongoing-threat-of-data-breaches-in-web3

What is Audit About?

Gain a thorough grasp of the system architecture, pinpointing assets, threats, and possible attack routes.

- Preparing the threat hypothesis for security review objective.
- Whitepaper / Technical Documentation Review
- Which part of the components or modules that the team feel uncertain or lack of specialty?



Leverage automated tools to scan the code for known vulnerabilities and common security issues.

- Integrated security tools: static analyzers, dependency checkers.
- Database of over **70,000** security findings.
- Sourced from CertiK audits, incident analyses, and research efforts.

Security experts perform detailed code and system reviews to find complex vulnerabilities beyond automated tool detection.

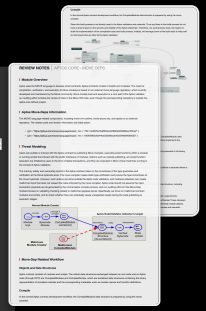
- A team formed by experts conducting the in-depth manual code analysis.
- Red-Team mindset to find the critical path issue.
- Concrete proof of concept (PoC) for case replication.

Perform runtime testing and analysis of the system to identify vulnerabilities that manifest during execution,

- Execute functions with various inputs to spot issues.
- Monitor processing of blockchain tx to identify logic, gas, and outcome issues.
- Test dynamic interactions with external sources to identify potential vulnerabilities.

The evaluation results will be provided to the client as a PDF, accessible via Skyharbor, designed for efficient audit and reporting.

- **Immediate Notification for High-Risk Findings**
- Vulnerability details with a brief overview and steps to replicate the issue.
- Repeating review changes / PRs to ensure the issues are fixed.



Standard-Specific Checklist

Language Specific Risk

Function Visibility, Compiler Version, Event, Low-level Call, Storage Risk etc

Price Manipulation Risk

Unexpected Price Update, Unintended Balance Change, etc

Governance Risk

Private Key Leakage, Voting Power Manipulation, etc

Common Security Issues

Input Validation, Reentrancy Attack, Access Control, etc

Wide-Range Web3 Security Risk Coverage

Incentive Design Flaw

Unfair Reward Distribution, Misaligned Incentives, Lack of Adaptability, Short-term Incentives with Long-term Detriments

Mathematics Operation

Precision Loss, Overflow / Underflow, Incorrect Math Calculation, etc

Business Logic Design Flaw

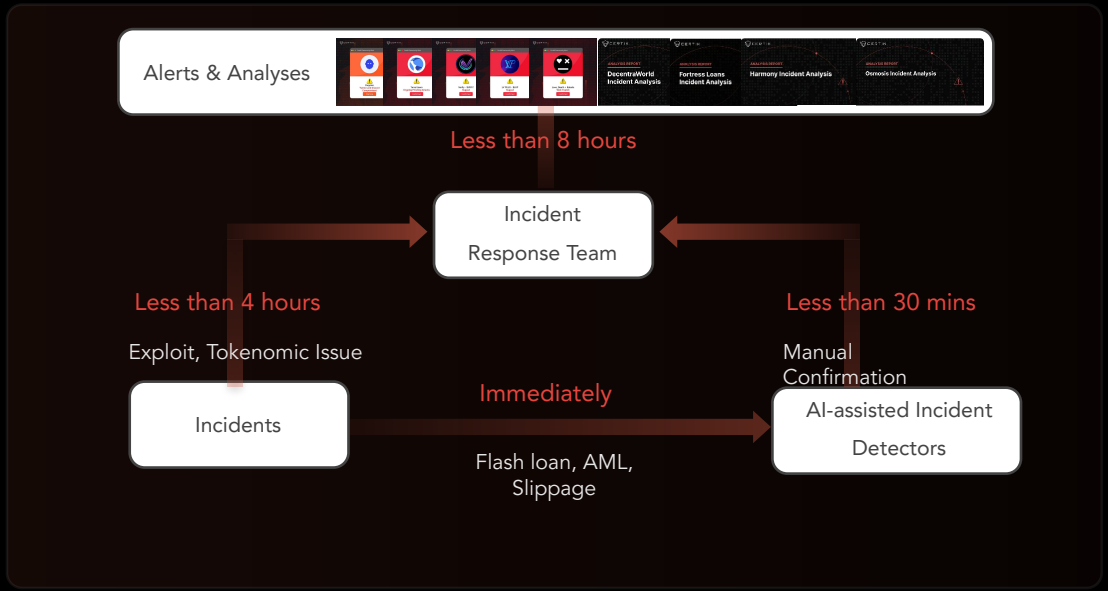
Abnormal Arbitrages, Inconsistent User Behavior resulting funds being locked, malicious calls, etc

Cross-Chain Risk

Vulnerable Proof Verification, Replay Attack, etc)

Round-the-Clock Security Alert and Response

- 24/7 Alert Verification and Incident Response
  - Dedicated team operates *24/7*, evaluating alerts to ensure accurate verification and timely response.
  - Efficiently triages automated alerts using Open Source Intelligence (OSINT) and on-chain analysis
  - Thorough examination of hacking incidents to understand attack vectors, coupled with the provision of preventive solutions to bolster security.



Securing The **Web3** World