Security Analysis of Cryptocurrency Exchanges

Sazzadur Rahaman
Department of Computer Science
The University of Arizona
Tucson, USA
sazz@arizona.edu

Rupal Jain
Department of Computer Science
The University of Arizona
Tucson, USA
jainrupal@arizona.edu

Muaz Ali
Department of Computer Science
The University of Arizona
Tucson, USA
muaz@arizona.edu

Fahmida Alam
Department of Computer Science
The University of Arizona
Tucson, USA
fahmidaalam@arizona.edu

Priya Kaushik
Department of Computer Science
The University of Arizona
Tucson, USA
priyakaushik@arizona.edu

***Abstract*—The most popular implementation of the blockchain technology is cryptocurrency. Most people use cryptocurrency exchanges to perform transactions on these cryptocurrencies. There has been a lot of work that looks into the security of cryptocurrencies. However, there is little to no work that looks into the security of these exchanges. Since these exchanges handle the private keys of users on their behalf, there are many security risks associated with them that need proper inspection. Recently, there has been a lot of breaches in these exchanges that fuel the motivation to look into the underlying security mechanisms of these exchanges. In a first of its kind study, we perform a security analysis of top cryptocurrency exchanges that include Binance, Bitfinex, KuCoin, and Coinbase. We were able to identify a general trend of vulnerabilities that require attention of the developers. We also devise future research directions to look into the security of cryptocurrency exchanges.**

I. Introduction

There has been a surge of blockchain based applications in the recent past. The most popular implementation of blockchain is in *cryptocurrency*. Some of the popular cryptocurrencies include Bitcoin and Ethereum [10], [11]. The underlying mechanism of blockchain involves a distributed ledger in which a transaction takes place via a consensus mechanism. For instance, Bitcoin uses Proof-of-work consensus algorithm [11], whereas Ethereum uses Proof-of-stake consensus algorithm [11]. Many of the cryptocurrencies use *RSA public key infrastructure* to sign transactions [15].

Users on a cryptocurrency require a wallet to perform transactions. A wallet contains private and public key pair of a user. However, in order to use this wallet first-hand, the users need to interact with the official client of the cryptocurrency, which can prove to be a laborious task. For the ease of use, many people use cryptocurrency exchanges [12], which manage the wallets of users on their behalf. These exchanges provide an easy to use interface that allows users to perform transactions on the go. The relieves the users the hassle to sign the transactions, send them to *miners*, and wait for them to be effectively be a part of the ledger. These exchanges also allow the capability to exchange currency across different cryptocurrencies. These exchanges are available on mobile platforms such as playstore and apple app store.

The use of these exchanges also comes with certain security risks. Letting an exchange handle your wallet for you comes with a trust requirement. However, these exchanges frequently are involved in breaches in which millions of dollars worth of *coins* have been stolen. This fuels the motivation to inspect the underlying security mechanisms of these exchanges.

Existing literature only deals with enhancing the security of blockchain itself. However, to the best of our knowledge, there is very little work done to inspect the security of these exchanges. In this study, we design a methodology to perform a security analysis of top cryptocurrency exchanges. In this regard, we use a state-of-the-art static analysis tool, Cryptoguard [2], to identify vulnerabilities in exchanges [12]. We also perform a blackbox study of the breaches in these exchanges, and then try to summarize the insights that were learnt from them.

II. Background

As of 2021, there are 300 million cryptocurrency users worldwide. Over the past two years, bitcoin use has multiplied astronomically. Globally, the number of bitcoin users increased subsequently between 2018 and 2019. Also, with sophisticated techniques methods, the cryptocurrency realm seems to be in dark phase. The innovation of a blockchain is that it fosters confidence without the necessity for a reliable third party by ensuring the fidelity and security of a record of data. Blockchain is utilized in the context of Bitcoin in a decentralized manner, ensuring that no one user or organization has power but rather that all users collectively maintain control. This implies that transactions made using Bitcoin are publicly visible and permanently recorded. This paper covers an high-level overview of the exchanges which are responsible for transactions of cryptocurrency.

III. Methodology

To investigate several third party exchanges used in cryptocurrency and their security vulnerabilities, we aim to perform a study of the crypto exchanges from the lens of security. For our analysis, we considered the top exchanges that include Binance, CoinBase, Bifinex and KuCoin [12]. So, our aim is to study the front end of these exchanges by investigating their mobile applications in a way that we can answer our research questions. We aim to answer what data these exchanges collect from the user, what are the vulnerabilities these exchanges have in their android application, what are the some major breaches for these exchanges and how can we prevent similar security breaches in future.

We performed vulnerability analysis for these exchanges using state-of the-art-tools, Cryptoguard, which is a static code checking tool, designed for developers for detecting cryptographic vulnerabilities from large Java projects. It is built on specialized forward and backward program slicing techniques which are implemented by using flow-, context- and field-sensitive data-flow analysis. So, using this security tool we performed a static analysis and detected all the vulnerabilities for all the exchanges using their Android APKS.

For breach analysis we performed a blackbox study for major breaches of these exchanges, and evaluated how these breaches took place, what are the causes of these breaches and then summarized the outcome from our evaluation. Moreover, according to our analysis, we proposed some countermeasures which might be adopted as a prevention to mitigate similar types of security breaches in future.

IV. Evaluation

A. Binance

Binance [13] is the most popular cryptocurrency exchange on Android Playstore. We performed static analysis of its apk by running Cryptoguard on it. Table IV-A enumerates the different security vulnerabilities found in Binance. Some examples of the vulnerabilities found by cryptoguard are shown in figures [1, 2, 3].

While these vulnerabilities could be attacked in an adversarial fashion against user, it is unclear if these vulnerabilities are part of the external modules that Binance uses, such as ad or analytics manager, or whether they are crucial in performing the cryptographic operations that facilitate blockchain infrastructure of different cryptocurrencies. We expect this to be analyzed deeply in the future work surrounding the security of cryptocurrency exchanges.

In one of the recent breaches in Binance, about $570 million worth of BNC (binance coins) were stolen [9]. It turned out to be a vulnerability in the smart contract that was used as a cross-bridge to transfer coins to other cryptocurrencies. This poses another question regarding the security of these exchanges: how to review the security guarantees of smart contracts when they are used in conjunction with exchanges. We suggest a formal verification of these guarantees to be conducted.

```
public final class zzyu extends zzzz {

    /* renamed from: f */
    private final Random f23045f = new Random();

    /* renamed from: g */
    private long f23046g;

    /* renamed from: h */
```

Fig. 1. Cryptoguard found the vulnerability of using an untrusted PRNG in one of Binance's decompiled classes. *java.util.Random()* is not a secure method for PRNG.

```
L_0x0098:
    java.lang.String r2 = "1542658731108"
    java.io.File r4 = new java.io.File
```

Fig. 2. Cryptoguard found the vulnerability of using an using a constant key in one of Binance's decompiled classes.

```
private void m14650d() throws Exception {
    InputStream inputStream;
    InputStream inputStream2;
    BufferedOutputStream bufferedOutputStream;
    Throwable th;
    OutputStream outputStream;
    if (!this.f12916t) {
        String str = this.f12908l;
        if (!TextUtils.isEmpty(str) && Uri.parse(str).getScheme() == null) {
            str = "http://".concat(String.valueOf(str));
        }
        this.f12908l = str;
        HttpURLConnection httpURLConnection = (HttpURLConnection) new URL(this.f12908l).openConnection();
        this.f12914r = httpURLConnection;
        httpURLConnection.setConnectTimeout(this.f12910n);
        this.f12914r.setReadTimeout(this.f12911o);
        this.f12914r.setRequestMethod(this.f12909m.toString());
        this.f12914r.setInstanceFollowRedirects(this.f12912p);
        this.f12914r.setDoOutput(C6900c.kPost.equals(this.f12909m));
        this.f12914r.setDoInput(true);
        TrafficStats.setThreadStatsTag(1234);
```

Fig. 3. Cryptoguard found the vulnerability of using an using HTTP protocol for communication in one of Binance's decompiled classes. HTTP connection exposes the communication to an attacker evasdropping, as it does not support content encryption.

B. Coinbase

A safe online marketplace for purchasing, selling, transferring, and storing cryptocurrencies is provided by Coinbase With over 170 cryptocurrencies available for trade, Coinbase is the biggest cryptocurrency exchange in the United States.

a) The following types of sensitive data are gathered from users by Coinbase: personal identification information, official identification information, institutional information, financial information, transaction information, employment information, and online identifiers like geolocation/tracking information and browser fingerprints and Usage data such as Security questions and authentication information. Data subject access requests, often known as "DSARs," are permitted by laws like the General Data Protection Regulation ("GDPR") and the California Consumer Privacy Act ("CCPA"), which regulate how personal data is used by service providers like Coinbase.

b) Cryptoguard Analysis: We performed static analysis of its apk by running Cryptoguard on it. Table II enumerates

	Vulnerability	Description	instances
1	Found broken crypto schemes	Cryptographic operations are broken. They could potentiallyleak information or might be insecure for communication.	4
2	Used untrusted PRNG	An untrusted library is used to to get Pseudo Random Number Generator.	17
3	Used untrusted HostNameVerifier	Host Name Verifier that performs the verification of a given Host Name before a connection to it is from an untrusted source.	2
4	Used constant keys in code	Use of constant defined keys that are used in cryptographic operations.	6
5	Found constant salts in code	Use of constant defined salts that are used in hashing.	1
6	HTTP Protocol use	The use HTTP protocol to perform communication. As this protocol does not support encryption, any communication that takes place is not confidential.	17

TABLE I
THIS TABLE LISTS DOWN THE DIFFERENT VULNERABILITIES FOUND IN BINANCE WHEN ANALYZED USING CRYPTOGUARD.

	Vulnerability	Description	instances
1	Found broken hash functions	Cryptographic operations are broken. They could potentially leak information or might be insecure for communication.	35
2	Used constant keys in code	Use of constant defined keys that are used in cryptographic operations.	8
3	HTTP Protocol used	The use HTTP protocol to perform communication. As this protocol does not support encryption, any communication that takes place is not confidential.	11
4	Used less than 1000 iteration for PBE	Static salts make dictionary attacks easier ON PBE .	23
5	Used untrusted PRNG	An untrusted library is used to to get Pseudo Random Number Generator.	66

TABLE II
THIS TABLE LISTS DOWN THE DIFFERENT VULNERABILITIES FOUND IN COINBASE WHEN ANALYZED USING CRYPTOGUARD.

the different security vulnerabilities found in Coinbase. Following are the instances with possible vulnerabilities found in the source code of the cryptoguard. Fig. 4, 5, 6, 7, shows the bug which can be exploited as [1]. Predictable pseudorandom number generators (PRNGs) [3]–[5], Fig. 8, 9, 10 shows some broken hash functions which are easier to crack if attacked, Fig. 11, 12 used HTTP protocol, which doesn't uses encryption hence, can lead to the breach in the transfer protocol. Here, the PRNG protocol is violated maximum times as shown by the cryptoguard. The possibility of message conflicts caused by accidentally duplicating message hash codes is a big issue with MD5. The maximum length of an MD5 hash code string is 128 bits. Compared to other hash coding techniques that came after, they are therefore more vulnerable. Any user watching the session can view all requests and responses if a website utilizes HTTP rather than HTTPS. In essence, a malicious actor can simply examine the language of a request or response to determine exactly what information is being requested, provided, or received.

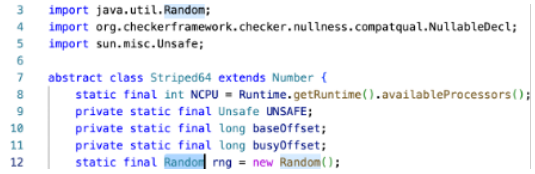

Fig. 4. Path: coinbase.apk/com/google/common/hash/Striped64.class

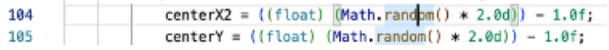

Fig. 5. Path: coinbase.apk/androidx/transition/Explode.class

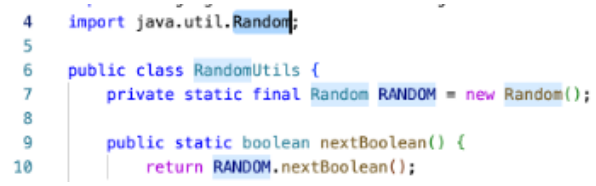

Fig. 6. Path: coinbase.apk/org/apache/commons/lang3/RandomUtils.class

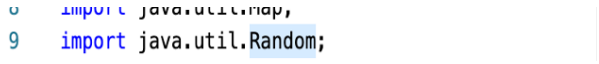

Fig. 7. Path: coinbase.apk/org/apache/commons/lang3/ArrayUtils.class

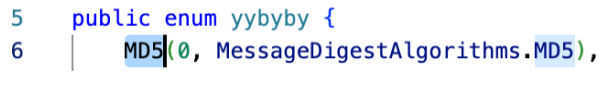

Fig. 8. Path: coinbase.apk/com/threatmetrix/TrustDefender/yybyby.class

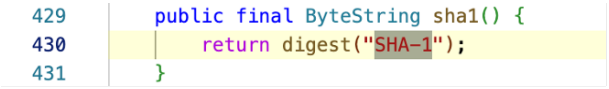

Fig. 9. Path: coinbase.apk/okio/Buffer.class

Fig. 10. Path: coinbase.apk/org/apache/commons/codec/digest/DigestUtils.class

Fig. 11. Path: coinbase.apk/retrofit2/Response.class

c) Breach Analysis for Coinbase Global Inc.: The accounts of at least 6,000 Coinbase users were compromised by hackers. Customers of affected Coinbase had money taken out of their accounts. Attackers most likely used phishing emails to obtain victims' email accounts before taking advantage of

Fig. 12. Path:coinbase.apk/com/facebook/react/devsupport/DevServerHelper.class

a security hole in Coinbase's two-factor SMS mechanism to access users' Coinbase accounts. Because of which the SMS two-factor authentication token required to access a guarded account was obtained by the hackers. Experts claim that SIM swapping is one cause of these instances. These several wallets are made to let you store your money without relying on Coinbase. They allow you the choice of using hardware, software, or an app to store your own cryptocurrency. Even if access is gained, the wallet, which is where your cryptocurrency is actually stored, won't be reachable. This method's disadvantage is that you must remember your own password, which is also known as a private key.

d) Prevention: There can be following ways instead of using SMS 2F authentication i.e. Time Based codes, Push Notifications Out-of-Band Authentication (OOBA), U2F Security Keys etc.

C. Bitfinex

Bitfinex, a cryptocurrency exchange, founded in Decembe 2012 [16] is peer-to-peer Bitcoin exchange digital asset trading that provided services to users across the world. There are 4 Fiat currencies as USD, EUR, JYP, GBP and 10 Cryptocu rencies as Bitcoin, Bitcoin Cash, Ethereum, Ethereum Classic Litecoin, Ripple, OmiseGO, Monero, NEO, EOS. Bitfinex uses two-factor authentication, advanced APIs for connecting third-party services, withdrawal protection features, and cold storage of customer assets [17]. Bitfinex stores 99.5% of user funds in cold storage with DDoS protection, database encryption, and regular backups [18].

a) Breach Analysis: Bitfinex suffered a major security breach on August 2, 2016 which become the second-biggest breach of a Bitcoin Exchange platform [20] A New York couple, Ilya Lichtenstein, and his wife, Heather Morgan, stole 119,756 units of Bitcoin, worth $72 million [21]. They initiated around 2000 approved transactions to transfer the funds to a single wallet from users' segregated wallets. The moved 10% of the Bitcoin by transferring the money through multiple dark-web applications. BitGo clarified that there were no server breaches at their end [19]. The source of the vulnerability appears to lie in the structure of the Bitfinex accounts. One cause is Bitfinex itself since it had the ownership of 2 out of 3 private keys needed for the funds lost from the multi-signature accounts [23]. Advocacy group Coin Center has claimed that a multi-signature security approach is prone to vulnerability or failure. While asymmetric encryption is well designed for sending and receiving sensitive data where the data is encrypted using the recipient's public key and can only be decrypted using the recipient's private key. This marks symmetric encryption is perfect for sharing access to a stationary file as the decryption password can be shared between the two parties [23].

b) Cryptoguard Analysis: Bitfinex has maintained its position to be in top 10 amongst the cryptocurrency exchanges, even after experiencing major breach. Let's conduct some vulnerability analysis for this exchange. The major count of instances is coming for rule number 2, 8 and 13. We have used the APK file for Android and then decompiled the file to get the source code of the exchange [27]. Using the source code, we have analyzed the vulnerabilities present in the application. Figure[13, 14] shows one such instance depicting the exploitation of rule 2 which checks for the insecure cryptographic hash functions. The application is getting MD5 value of the input parameter. The MD5 value is analyzed using the java library called org.apache.commons.codec.digest.MessageDigestAlgorithms. This vulnerability is found in path "bit.apk/com/facebook/common/util/SecureHashUtil.class". This function is using the value of MD5 directly which is making it vulnerable to get attacked. There are 27 such instances which are using easily predictable values of MD5 and SHA-1 [26].

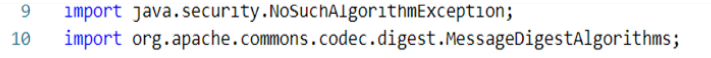

Fig. 13. MessageDigestAlgorithm used to get MD5 value is imported.

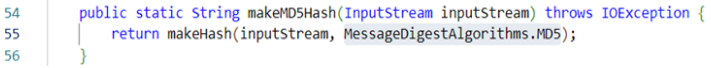

Fig. 14. Line 55 is using MD5 value.

Next is the exploitation of rule 7. This rule tells that there are instances which are using HTTP. HTTP is a less secure version of the protocol and is very prone to cyberattacks on SSL/TLS MitM. As seen in the below, the file "PackagerStatusCheck.class" at path "bit.apk/com/facebook/react/devsupport/" is using "http://s/status". We should also avoid using HTTP but instead opt for some more secure methods. Figure 15 shows how the bitfinex application is exploiting the application.

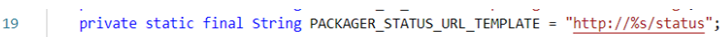

Fig. 15. Line 19 shows the use of HTTP

In further analysis, we have identified an Untrusted Pseudorandom number generator that is using java.util.Random. According to the rule, using these random generators becomes a source of vulnerability which can lead to major attacks [25]. Pseudorandom number generator seeds should not be predictable and should not be constant. In the file named "Striped64.class" situated at the path "bit.apk/com/google/common/hash/" is using static random number generator. Figure 16 depicts that same.

```
3   import java.util.Random;
4   import javax.annotation.CheckForNull;
5   import sun.misc.Unsafe;
6
7   @ElementTypesAreNonnullByDefault
8   abstract class Striped64 extends Number {
9       static final int NCPU = Runtime.getRuntime().availableProcessors();
10      private static final Unsafe UNSAFE;
11      private static final long baseOffset;
12      private static final long busyOffset;
13      static final Random rng = new Random();
```

Fig. 16. Line 13 is using Random function after importing java Random utility.

Password-based-encryption takes some additional input. One such input is iterations. The iteration count may be transmitted to the receiver along with the ciphertext. It is highly recommended to use 1000 or more iterations to achieve a good security level. If the number of iterations would be small enough then the chances of obtaining the secret key would increase. Brute-force on the large number of iterations would significantly increase the time complexity [24]. Figure 17 illustrates how there is only 20 iteration have been used.

```
123     protected SecureRandom random = CryptoServicesRegistrar.getSecureRandom();
124     private int saltLength = 20;
```

Fig. 17. Line 13 is using Random function after importing java Random utility.

	Vulnerability	Description	Instances
1	Found broken hash functions	Cryptographic operations are broken. They could potentially leak information or might be insecure for communication.	27
2	Used untrusted PRNG	An untrusted library is used to get Pseudo Random Number Generator.	49
3	Used $<$ 1000 iteration for PBE	Using fewer than 1000 iterations for PBE is not secure.	23
4	HTTP Protocol use	The use HTTP protocol to perform communication. As this protocol does not support encryption, any communication that takes place is not confidential.	6
5	Found constant IV in code	The use of predictable Initialization vector (IV) CBC can cause a Dictionary attack when they are encrypted with the same key.	5

TABLE III
THIS TABLE LISTS DOWN THE DIFFERENT VULNERABILITIES FOUND IN BITFINEX WHEN ANALYZED USING CRYPTOGUARD.

c) Prevention: Bitfinex does the following security measures to protect their system and reduce Bitfinex vulnerabilities: [28]

- Uses protection against distributed denial of service (DDoS) attacks
- Automatically backs up databases daily
- Automatically encrypts and duplicates backup data
- Updates software and Linux systems regularly
- we moved all of the bitcoins to our multi-sig cold wallet (secured hard-drive).

The most efficient way to hold a wallet is on a "custodial" basis because in this way the exchange operates and controls a single wallet or a set of wallets for its clients with its clients essentially being allocated bitcoin within that wallet [29].

D. KuCoin

Kucoin is a global cryptocurrency exchange for various digital assets and cryptocurrencies which was founded in 2013 by Michael Gan, Eric Don, Top Lan, Kent Li, John Lee, Jack Zhu, and Linda Lin, and was launched in September 15, 2017 [30] [31]. KuCoin has grown into one of the most popular crypto exchanges and already has over 8 million registered users from 200+ countries and region[1]. Currently, according to Alexa traffic ranking, globally KuCoin's monthly unique ranking is in the top 5 and known as the "People's Exchange".

a) The following types of sensitive data are gathered from users by Coinbase: The KuCoin users agree that the Platform may use cookies to track the user's actions in connection with their use of the platform and may collect and record all information left by users, including but not limited to their IP address, geographical location and other data. It also collects personal data which may include Personal Identification Information , Formal Identification Information, Virtual Identity, Institutional Information, Financial Information, Transaction Information [32].

b) Cryptography analysis for KuCoinn: KuCoin is the one of the top 5 most popular cryptocurrency exchange according to the Android Playstore. Due to its popularity we performed static analysis of using its apk by running Cryptoguard on it. We analysed its vulnerabilities and found how many instances are there for each vulnerability. Table IV shows the results of all the vulnerabilities with their description and found instances

From the table we can see major vulnerabilities are found for the rule number 2, 10 and 13. Moreover, There are a few instances found for rule 1, 3, 4, 7. In the following Figure 18, Figure 19, Figure 20, Figure 21, Figure 22, Figure 23, we can see all the vulnerabilities Cryptoguard detected for this exchange.

```
public final List<String> a(String str, String str2) {
    ArrayList arrayList = new ArrayList(512);
    MessageDigest messageDigest = MessageDigest.getInstance("MD5");
    MessageDigest messageDigest2 = MessageDigest.getInstance("MD5");
    MessageDigest messageDigest3 = MessageDigest.getInstance("MD5");
    DexFile dexFile = new DexFile(str2);
    try {
        Enumeration<String> entries = dexFile.entries();
```

Fig. 18. Cryptogurad found broken hash function in the code for MD5

c) Breach Analysis for KuCoin: On September 25, 2020, the cryptocurrency exchange KuCoin was hacked by a North Korean hacker crew called Lazarus Group and they managed to steal over 281m dollar worth of coins and tokens in this security breach. The KuCoin team explained that the hack was due to a leak of the KuCoin hot wallet private keys.

	Vulnerability	Description	Instances
1	Found broken crypto schemes	Cryptographic schemes are broken which may prompt insecure communication or may lead data vulnerabilities.	1
2	Found broken hash functions	Cryptographic operations are broken. They could potentially leak information or might be insecure for communication	67
3	Used constant keys in code	Use of constant defined keys that are used in cryptographic operations.	3
4	Uses untrusted TrustManager	Using untrusted TrustManager may prompt trusting untrusted certificate from malicious actor	3
5	Used HTTP Protocol	The use HTTP protocol to performcommunication. As this protocoldoes not support encryption,any communication that takes place is not confidential.	9
7	Found constant IV in code	The use of predictable Initialization vector (IV) CBC can cause a Dictionary attack when they are encrypted with the same key.	25
8	Used untrusted PRNG	An untrusted library is used to to get Pseudo Random Number Generator.	57

TABLE IV
THIS TABLE LISTS DOWN THE DIFFERENT VULNERABILITIES FOUND IN KUCOIN WHEN ANALYZED USING CRYPTOGUARD.

```
package com.xiaomi.push;

import com.google.common.base.Ascii;
import javax.crypto.Cipher;
import javax.crypto.spec.IvParameterSpec;
import javax.crypto.spec.SecretKeySpec;

/* loaded from: classes7.dex */
public class h {
    private static final byte[] a = {100, Ascii.ETB, 84, 114, 72, 0, 4, 97, 73, 97, 2, 52, 84, 102, Ascii.DC2, 32};

    private static Cipher a(byte[] bArr, int i2) {
        SecretKeySpec secretKeySpec = new SecretKeySpec(bArr, "AES");
        IvParameterSpec ivParameterSpec = new IvParameterSpec(a);
        Cipher cipher = Cipher.getInstance("AES/CBC/PKCS5Padding");
        cipher.init(i2, secretKeySpec, ivParameterSpec);
```

Fig. 19. Cryptoguard found constant IV in code

```
/* renamed from: a */
private Random f725a = new Random();
private int a = 0;
```

Fig. 20. Cryptoguard found Untrused PRNG (java.util.Random)

```
public class a {
    public static String a(String str) {
        try {
            Cipher cipher = Cipher.getInstance("AES/CBC/PKCS5Padding");
            SecretKeySpec secretKeySpec = new SecretKeySpec("5HZ2Maf9FDmDo2MA".getBytes(), "AES");
            AlgorithmParameters algorithmParameters = AlgorithmParameters.getInstance("AES");
            algorithmParameters.init(new IvParameterSpec("T3M7gLcZKXevBmSA".getBytes()));
            cipher.init(1, secretKeySpec, algorithmParameters);
```

Fig. 21. Cryptoguard found constant keys on code)

```
@Override // javax.net.ssl.X509TrustManager
public X509Certificate[] getAcceptedIssuers() {
    return new X509Certificate[0];
}
```

Fig. 22. Cryptoguard found untrusted TrustManager)

Hot wallets , also known as online wallets, store private keys on systems or devices that are connected to the internet. Private key represents a randomly generated number that

```
"Response.error()").protocol(Protocol.HTTP_1_1).request(new Request.Builder().url("http://localhost/").build()).build(
```

Fig. 23. Cryptoguard found using HTTP protocol)

signs transactions and protects user's assets from malicious attacks. If it gets compromised or lost, users won't be able to access their crypto wallet to spend, withdraw, or transfer their coins. Although KuCoin did not disclose the the exact way how they retrieved the private key, but according to our research we found some probable way that hacker could use to leak the private key, Such as Social Engineering attack, Malicious actions of responsible employees, Attack on web infrastructure. To launder stolen money hacker used DeFi platform, for instance Uniswap. So, to do that first they moved LINK from their initial wallet to an intermediary, and from there sent it to Uniswap. Then to complete the swap hacker sent it back to the same address and successfully launder the stolen money.

d) How these breaches can be prevented?: Following are some steps which can be followed to prevent such breaches [33],

- The hot wallet must be reinitialized periodically.
- Two-man rule can be adopted such as using secret sharing schemes. One of the most popular ways is to use Shamir's Secret Sharing scheme.
- One must not store more than 5 percent of all deposits in the hot wallets and 95 percent must be stored in a cold wallet.
- User should store crypto in several hot wallets for each cryptocurrency platform and each wallet must have its own private key.
- Performing regular penetration tests, phishing simulation, red team exercises are must.ey.
- Performing audits of the cryptocurrency storage system that is included in SOC2 and/or ISO27000 auditing is also compulsory.

DISCUSSION

No Cryptocurrency is perfectly secure in today's world. We discussed the vulnerabilities of the cryptocurrency exchanges and the usage of Cryptoguard. Analysis has been done on the Android application using Cryptoguard software which involves generating the vulnerability detection report in JSON format, and decompiling the APK file to get the source code. Next, we found the code where a vulnerability has been detected. We are planning to inspect the the threat level of these vulnerabilities by analyzing whether these vulnerabilities are part of the important blockchain infrastructure or if they belong to benign external components such as ad-manager. This future work will be crucial in in mitigating attacks that could be launched exploiting these vulnerabilities.

RELATED WORKS

There have been various speculations on the cryptocurrency exchanges and their breaches. It is claimed that cryptocur-

rency exchanges utilize wash trading to inflate their liquidity signals. The high trading volume feigned by wash trading exchanges raises their visibility on popular websites that monitor cryptocurrencies markets. Markets that are weak are more susceptible to manipulation. Such markets are known for their liquidity, which makes it one of the places where manipulation may result in significant profits at the expense of investors. The extent of wash trading, a practice used to boost an exchange's apparent liquidity, has been the subject of earlier research. These research have mostly examined transaction patterns, infrequently examined site traffic data, and never examined the balances of exchange tokens [6].

The first attempt to name and describe bitcoin exchange scams is made by [7]. By compiling previous reports and employing typosquatting creation techniques, more than 1500 scam domains and more than 300 fraudulent apps are first discovered. [7] then discovers 94 scam domain families and 30 phony app families by looking into the connections between the scam domains and false apps.

According to [8], the contemporaneous effect diminishes to a statistically negligible 1.100 percent return throughout the 2019–2021 decade. This shows that over the latter portion of the sample period, the market ceased pricing the news about cybersecurity breaches at cryptocurrency exchanges into Bitcoin. One possibility is that the price of bitcoin currently reflects the fact that exchanges have improved their cybersecurity procedures throughout the more recent subsample period. Due to their short history, diminutive sizes, and infrequent trade volumes, cryptocurrency exchanges typically lack ML detection capabilities. [14] creates a model to comprehend the format of reporting choices made by ML monitoring institutions. As demonstrated by [14], cryptocurrency exchanges with weak ML detection capabilities tend to overreport suspicious situations.

V. Conclusion

We performed a security analysis of different vulnerabilities found in cryptocurrency exchanges. Mainly using static analysis, we found a significant number of vulnerabilities in these exchanges. Based on the insights we gathered from this study, we listed potential research directions to further inspect the security of the conjunctions between exchanges and cryptocurrencies.

References

[1] S. Vaudenay (Ed.): Fast Software Encryption – FSE'98, LNCS 1372, pp. 168–188, 1998 Springer-Verlag Berlin Heidelberg 1998.

[2] Sazzadur Rahaman, Ya Xiao, Sharmin Afrose, Fahad Shaon, Ke Tian, Miles Frantz, Murat Kantarcioglu, Danfeng (Daphne). CryptoGuard: High Precision Detection of Cryptographic Vulnerabilities in Massive-sized Java Projects in Yao1JCCS '19 November 11–15, 2019 London, United Kingdom

[3] D.J.Bernstein,Y.Chang,C.Cheng,L.Chou,N.Heninger,T.Lange,andN.van Someren. Factoring RSA Keys from Certified Smart Cards: Coppersmith in the Wild. In ASIACRYPT'13, pages 341–360, 2013.

[4] I. Goldberg and D. Wagner. Randomness and the Netscape browser. Dr Dobb's Journal-Software Tools for the Professional Programmer, 21(1):66–71, 1996.

[5] N. Heninger, Z. Durumeric, E. Wustrow, and J. A. Halderman. Mining Your Ps and Qs: Detection of Widespread Weak Keys in Network Devices. In USENIX Security'12, pages 205–220, 2012.

[6] Le Pennec et al., 2021 G. Le Pennec, I. Fiedler, L. Ante Wash trading at cryptocurrency exchanges Finance Res. Lett., 43 (2021), Article 101982

[7] Xia P., Wang H., Zhang B., Ji R., Gao B., Wu L., Luo X., Xu G. Characterizing cryptocurrency exchange scams Comput. Secur., 98 (2020), Article 101993

[8] George Milunovich, Seung Ah Lee, Measuring the impact of digital exchange cyberattacks on Bitcoin Returns, Economics Letters, Volume 221, 2022, 110893, ISSN 0165-1765

[9] Binance hacked for 570 million dollars. https://arstechnica.com/information-technology/2022/10/binance-blockchain-suffers-570-million-hack/.

[10] Nakamoto, S. (2008) Bitcoin: A Peer-to-Peer Electronic Cash System. https://bitcoin.org/bitcoin.pdf

[11] Buterin, V. (2015). A NEXT GENERATION SMART CONTRACT & DECENTRALIZED APPLICATION PLATFORM.

[12] Top cryptocurrency exchange, https://coinmarketcap.com/rankings/exchanges/

[13] Binance at Playstore. "https://play.google.com/store/apps/details?id=com.binance.dev"

[14] Kim, D., Bilgin, M.H. Ryu, D. Are suspicious activity reporting requirements for cryptocurrency exchanges effective?. Financ Innov 7, 78 (2021). https://doi.org/10.1186/s40854-021-00294-6

[15] The RSA Public-Key Encryption Algorithm. In: Furht, B. (eds) Encyclopedia of Multimedia. Springer, Boston, MA. https://doi.org/10.1007/0-387-30038-4_206

[16] "Company Overview of iFinex Inc. (BVI)". Bloomberg. Archived from the original on July 1, 2018. Retrieved June 30, 2018.

[17] Bitfinex Security Policy, https://www.bitfinex.com/security-policy/

[18] Bitfinex Support, https://support.bitfinex.com/hc/en-us/articles/213892469-Security-Features-on-the-Bitfinex-Platform

[19] Shekhtman, Lonnie (August 3, 2016). "Bitcoin security breaches raise questions about digital currency's future". Christian Science Monitor. Archived from the original on May 28, 2017.

[20] "Bitfinex comes back from $69 million bitcoin heist". May 21, 2017. Archived from the original on May 22, 2017.

[21] Tsang, Amie (August 3, 2016). "Bitcoin Plunges After Hacking of Exchange in Hong Kong". The New York Times. Archived from the original on May 18, 2017.

[22] US Arrest Warrant, https://www.justice.gov/opa/press-release/file/1470211/download

[23] Cause of Hack, https://www.coindesk.com/markets/2016/08/03/the-bitfinex-bitcoin-hack-what-we-know-and-dont-know/

[24] PBE algorithm, http://www.crypto-it.net/eng/theory/pbe.htmlpart_iterations

[25] Untrusted PRNG, https://cwe.mitre.org/data/definitions/337.html

[26] Wang, X., Lai, X., Feng, D., Chen, H., Yu, X.: Cryptanalysis of the Hash Functions MD4 and RIPEMD. In: Cramer, R. (ed.) EUROCRYPT 2005. LNCS, vol. 3494, pp. 1–18. Springer, Heidelberg (2005)

[27] Decompiler, http://www.javadecompilers.com/apk

[28] Prevention startegies, https://cryptonews.com/reviews/bitfinex/

[29] Wallet Preferences, https://www.jdsupra.com/legalnews/the-us-4-5-billion-bitfinex-hack-five-7029856/

[30] KuCoin, https://en.bitcoinwiki.org/wiki/KuCoin

[31] KuCoin-Exchange, https://golden.com/wiki/KuCoin Exchange-NMGDK49

[32] KuCoin-hack, https://cryptonews.com/reviews/kucoin/

[33] https://blog.coinmarketcap.com/2020/10/16/kucoin-september-2020-hack-hacken-research/