HoneyDetails: A prototype for ensuring patient’s information privacy and thwarting electronic health record threats based on decoys

Abiodun Esther Omolara
Aman Jantan
Universiti Sains Malaysia, Malaysia

Oludare Isaac Abiodun
Bingham University, Nigeria

Humaira Arshad
Universiti Sains Malaysia, Malaysia

Kemi Victoria Dada
Ahmadu Bello University, Nigeria

Etuh Emmanuel
Arthur Jarvis University, Nigeria

Abstract
Advancements in electronic health record system allow patients to store and selectively share their medical records as needed with doctors. However, privacy concerns represent one of the major threats facing the electronic health record system. For instance, a cybercriminal may use a brute-force attack to authenticate into a patient’s account to steal the patient’s personal, medical or genetic details. This threat is amplified given that an individual’s genetic content is connected to their family, thus leading to security risks for their family members as well. Several cases of patient’s data theft have been reported where cybercriminals authenticated into the patient’s account, stole the patient’s medical data and assumed the

Corresponding author:
Abiodun Esther Omolara, School of Computer Sciences, Universiti Sains Malaysia, 11800 Penang, Malaysia.
Email: styleest2011@gmail.com

Creative Commons Non Commercial CC BY-NC: This article is distributed under the terms of the Creative Commons Attribution-NonCommercial 4.0 License (http://creativecommons.org/licenses/by-nc/4.0/) which permits non-commercial use, reproduction and distribution of the work without further permission provided the original work is attributed as specified on the SAGE and Open Access pages (https://us.sagepub.com/en-us/nam/open-access-at-sage).
identity of the patients. In some cases, the stolen data were used to access the patient's accounts on other platforms and in other cases, to make fraudulent health insurance claims. Several measures have been suggested to address the security issues in electronic health record systems. Nevertheless, we emphasize that current measures proffer security in the short-term. This work studies the feasibility of using a decoy-based system named HoneyDetails in the security of the electronic health record system. HoneyDetails will serve fictitious medical data to the adversary during his hacking attempt to steal the patient’s data. However, the adversary will remain oblivious to the deceit due to the realistic structure of the data. Our findings indicate that the proposed system may serve as a potential measure for safeguarding against patient’s information theft.

Keywords
adversary, attacker, deception, decoys, electronic health record, encryption, patient, privacy and security

Introduction
The electronic health record (EHR) system leverages on the machinery of information and communications technology (ICT) for the delivery of efficient, seamless and quality healthcare service. The EHR system otherwise referred to as electronic healthcare system (EHS), electronic medical records (EMRs), electronic personal record (EPR), telecare medicine information system (TMIS) and others provides a digital channel for integrating the computerized and the physical world of healthcare administration. It enables a patient to create, manage and control their health details in a seamless platform through the web, subsequently providing smooth transfer, storage, retrieval and sharing of patient’s personal and health records with medical personnel.1,2

The EHR system is also beneficial to the medical facility, as it allows the medical personnel to remotely access accurate and up-to-date patient’s records without the need to meet the patient on a regular face-to-face basis. Furthermore, the EHR system has proven very useful in providing coordinated and efficient care, for instance, in cases where a patient needs an emergency treatment and as the ambulance is conveying the patient to the clinic, the doctor is accessing their data to check drugs they may be allergic to and their medical history to avoid the adverse effect of administering the wrong treatment and reducing medical errors. The EHR system has given birth to the Internet of Medical Things (IoMT) which allows an Internet-enabled smart device (such as a smartwatch, smart lens) to integrate with implantable and wearable medical sensors to monitor patient’s health, predict an impending heart attack, check adherence to prescriptions, make diagnosis and so forth. This technological breakthrough has accelerated and improved correspondence between doctors and their patients, as well as enabling doctors to collaborate to share knowledge for the betterment of healthcare delivery and wellbeing of humanity.

A patient’s EHR record contains critical information such as personal identity details (full names, date of birth, social security number, address, phone number), medical diagnosis, prescriptions, genetic testing results, vital signs, sexual orientation, psychological details, accumulated hospital visitations records, radiology reports, allergies, family medical history and so on.3,4 The EHR record may also contain the patient’s genetic records which links him or her to their ancestral descent, thus, posing a threat to the family members in the event that the data get stolen. The value of medical data is comparable with a customer’s bank detail which contains the customer’s personal and financial information, and this makes it a remunerative target for cybercriminals. For instance, a malicious person may steal a patient’s record with the intention of impersonating the patient to make fraudulent health insurance claims or extract some details to access the patient’s account on other platforms. Other malicious acts involve using the stolen records to blackmail the
patient or sell the stolen data to agents in the black market or underground world. Consequently, the significance of a patient’s health information to healthcare providers, patients, insurance company and criminals is high as its misuse may present a detrimental effect on not only the patient but also the healthcare providers which may be held accountable. While Health Insurance Portability and Accountability Act (HIPAA) of 1996 regulations encourage patients to protect the privacy of their personal and health-related data, the EHR (under the management of the health facility and service provider) is held accountable for the confidentiality of patient’s data.

Furthermore, the threat to patient’s data can be terminal because unlike the theft of credit card or account data that could be short-term since the legitimate owner can cancel the credit card or change account details, healthcare data contain sensitive and personalized data that cannot be changed for life. For instance, people cannot change their age as it may affect medical judgment. Genetic details, family health reports, traits, allergies, psychological details and so on are patient’s details that remain unchanged for life. Thus, the successful theft of a patient personal record containing his or her genetic information creates an interdependent security risk for their family members as well. In this scenario, the criminal will not only gain access to the patient’s records but also his family historical and medical records and can use them to his advantage.

According to Protenus, in 2017, 477 healthcare breaches were reported to the US Division of Health and Human Services (HHS). The information available for 407 of those incidents showed that a total of 5.579 million patient records were affected. Furthermore, records from studies carried out by Ahmed and Ahmad showed that an average of 35 breaches occur every month and the cost implication of each breach is about $2.2 million. Verizon’s 2018 Protected Health Information Data Breach Report (PHIDBR) based on 1368 incidents across 27 countries reveals that the focus of healthcare breaches was mainly the patients, their identities, health histories and treatment plans. It was reported that some of the data and stolen identities were used to commit several frauds ranging from masquerading as the patient to fraudulently claim monetary benefits at government health-insurance facilities to selling the acquired patient’s data in the underground world, also known as the dark web. This health-related fraud costs the United States an estimate of $80 billion per year, and according to related studies, billions of dollars in other fraud cases have remained undetected.

Current and existing technologies enforce security in the EHR system using cryptographic-based approaches. Cryptography consists of a pair of encryption and decryption algorithms. Encryption is the process of transforming the patient’s data into an unrecognized form to prevent unauthorized access to the data using a cryptographic key and an encryption algorithm. Decryption is recovering the transformed message by supplying the cryptographic key(s) used in securing the message and a decryption algorithm. A couple of propositions have been made to control the security risk associated with patients’ personal records by enforcing several encryption schemes. However, these existing proposals provide only short-term security, thus, failing to provide an ideal solution to this huge security problem. In addition, the field of cryptography has seen rapid advancements: old encryption algorithms are circumvented, broken and replaced with new algorithms. However, new algorithms are also subsequently evaluated and broken. The release of new algorithms has inspired the design of complex cryptanalytic designs, for instance, the biclique, linear and differential attacks, boomerang attacks and so on.

This study proposes a long-term security measure for protecting patient’s information privacy by leveraging the intuition of honey/decoy systems in building realistic-looking but fake details as a counter-attack for thwarting threats in healthcare systems. Collectively, the contribution of this study is a decoy/deception system which generates fictitious patients’ information as a response to an attacker who attempts to brute force and steal the details of a patient.
Current cryptographic-based measures

The paper age of securing and storing patient’s data in an office storage room was transformed into a cryptographic-based approach with the advent of the EHR system. Patient’s information is transmitted over public and open channels protected using authentication and cryptographic protocols, and the patient is required to authenticate/log into the system using passwords or PINS.

A legitimate user/patient has a registered electronic account with the medical facility which contains his personal and medical profile/records. If he wants to access his account, he has to authenticate or log into the EHR system using his username and password. Existing EHR systems employ a single factor authentication system which requires the user to enter his username and password. The security of the password-based system depends on the strength of the user-selected password. However, various studies have shown that users select simple, weak, predictable and easy-to-guess passwords such as “iloveyou” and “123abc,” which are often lowercase/uppercase letters, digits, dictionary words, their children names, spousal names.15,16 The weak selection of user passwords makes the task of the attacker simpler as he does not need to guess too many combinations before he achieves access to the patient’s account. Furthermore, most attackers are aware of how users pick their passwords based on data released on the Internet on past data breach and leaks, thus increasing the attacker’s chances of successfully stealing the patient’s data.15,16 Several multi-factor authentication schemes have been proposed.17–19 However, they remain within the confines of academic research as most of them are not practicable in real-world deployment, their security is flawed; they limit the patient’s usability and impede the efficiency of the EHR system. Thus, their failure in being adopted in real-world medical systems.

A bio-cryptography framework was presented by Omotosho et al.20 where fingerprint and iris features are leveraged on to build secure cryptographic keys for an EHR system. Yu et al.21 investigated the likelihood that watermarks on leaked EHRs can be effectively detectable by a trusted organization in the cloud, without trading off the medical precision or functional proficiency of the EHR system. Lu et al.22 proposed an elliptic curve cryptosystem for the security and privacy of patient’s data. Also, Moon et al.23 proposed a security measure based on chaotic maps. Other efforts to secure the EHR data include increasing the encryption keys, the complexity of the encryption algorithms, introducing image encryption schemes on the EHR system.24–29 However, these existing proposals provide only short-term security, thus, failing to provide an ideal solution to this huge security problem. We highlighted the current security measures as short-term because a fool-proof encryption strategy that is strong today may fail tomorrow. An attacker may acquire the encryption key used in securing a patient’s data by brute-forcing/guessing the key under which the message is encrypted. More importantly, the advent of sophisticated high-computing tools such as graphical processing units (GPUs), field programmable gate arrays (FPGAs) and so on relinquishes a tremendous computational power to attackers to achieve the cryptographic keys in a short time and successfully stealing the patient’s detail. It is imperative to highlight that an encryption scheme considered reliable today might gradually debilitate to a brute-force attack in the long run. Consequently, it is not too farfetched to envision that an attacker in possession of a stolen encrypted patient record might have the capacity to decode it in a short time.30–33

Modern cryptography is constructed on mathematical theory and computer science practice; cryptographic algorithms are formulated around computational hardness hypothesis, making such algorithms hard to break in practice by any adversary. However, the emanation of practical quantum computers poses a consequential threat to cryptographic schemes currently in use. This is because quantum computers would allow solving the hard mathematical problem used in securing the encrypted data in a simple, fast and efficient way.34 Complex computations, such as integer
factorization and discrete logarithm problems, which are the heart of public-key cryptography, will be broken easily.\textsuperscript{35} Therefore, quantum computers will render every application protected by modern or conventional encryption to be ineffective. Thus, there is an urgent need for a robust EHR system to address the privacy issues related to safeguarding patient’s data.

**Deception and decoy-based systems and applications**

The use of decoys and deception is attracting widespread interest in the cyber domain. A decoy system is a technique of deceiving or misleading an attacker by providing him with plausible-looking but unknown to him, fake information during his attack. In the deception and decoy-based countermeasure of cybersecurity, a false reality is projected to the attacker as a reality which he is to accept as reality based on conviction from the target result of his attack.\textsuperscript{36} Deception and decoy-based measures have been applied extensively in the cyber world for curtailing several economic problems.\textsuperscript{37} For instance, *honeypots* are decoy systems intended to lure potential attackers away from critical systems and urging the attackers to remain attached to the system long enough for the system administrator to collect information about their malicious activities.\textsuperscript{38} Decoy data—such as decoy documents or valid-looking but fake information generated on demand—are used for detecting and luring away unauthorized access to data.\textsuperscript{39,40}

Huang et al.\textsuperscript{41} proposed the use of GenoGuard, a decoy-based genomic system to protect the privacy of patient’s DNA records against brute-force attacks. Park et al.\textsuperscript{42} proposed a novel approach of building disguised server against scanning attack by employing honey port. Irvene et al.\textsuperscript{43} proposed a novel honeypot that is set up as lures for attackers particularly for robotic systems.

This study explores the possibility of incorporating a decoy and deception system into the security of medical data to protect patient’s information privacy. Our justification for exploring and leveraging the mentioned approach stems from the intuition that decoy systems have been employed in real-life applications to tackle practical problems. The succeeding section gives a generic overview of an adversarial model applicable to patient’s medical data.

**Adversarial model**

We model a computationally unbounded adversary who has access to the databank containing the patient’s data through the EHR system. The adversary knows the encryption scheme used in encrypting the data, but he has no knowledge of the encryption/decryption key. We assume the encryption key to be derived from a user-chosen password. However, the adversary has the capability and resources of enumerating all the keys in the keyspace. The adversary leverages high-computing tools to brute-force the encrypted data by repeatedly guessing the password or encryption key used to encrypt the patient’s details. The attacker can determine if each decryption is successful as a wrongly guessed key will return non-uniform or unreadable characters as output. On the other hand, the attacker can infer that his guessed key is correct when uniform, plausible and readable data are returned as output. The current cryptographic measure is designed in such a way that only a key in the keyspace (which is the correct key) will yield the patient’s data. This opens a vulnerability as the adversary can run in polynomial time until he recovers any readable data which are the patient’s data. Figure 1 presents one of how the current cryptographic measures can be exploited to recover the patient’s data.

In Figure 1, the patient’s clear-text data are encoded in American Standard Code for Information Interchange (ASCII) format. The ASCII character set is the most widely used character set represented in the computer using standardized numeric codes. It has 128 characters, with values from
After each clear-text, medical data are represented in their ASCII form; they are now encrypted under an encryption key, $K_{ENC}$, using state-of-the-art encryption scheme and transmitted through the communication channel. The medical personnel of the health facility supplies the decryption key, $K_{DEC}$, to access the patient’s information. This means that $K_{ENC} = K_{DEC}$ yields the patient’s personal and medical data. However, a malicious person eavesdropping on the communication channel that tries to guess the decryption key, $K_{DEC}$, using random passwords gets non-uniform random characters/invalid input. In this case, $K_{ENC} \neq K_{DEC}$ yields gibberish or random non-uniform characters. This is a clear distinguisher to the attacker that his or her decryption keys are incorrect, and he or she continues trying (guessing/brute-forcing) other keys until he or she gets readable data that form meaning such as the patient’s name, address, medical reports and so on.

This study proposes a decoy-based approach where a realistic-looking but fictitious personal and medical record is served to an attacker in every ($K_{ENC} \neq K_{DEC}$) of his attempts to hack, guess or brute-force the patient’s password to steal the data. Other generic attack models often encountered with respect to EHR security and privacy are briefly discussed in Table 1.

Cybercriminals have employed some of the attacks listed in Table 1 as reported in past breaches released on the Internet. The rate of attack on medical data continues to be on the increase, thus demonstrating the need for alternative security measures that enhance current measures.
Table 1. Attack models.

| Attacks          | Categories                                      | How it is operated                                                                                                                                                                                                 | Recent attack/case                                                                                                                                                                                                 |
|------------------|-------------------------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Password         | Brute-force attacks                             | In password guessing and cracking, various techniques such as brute-force attack, dictionary attack, rainbow attack are leverage to recover the password of the client.                                                                 | Recently, it was reported that a malicious software was used to attack several Florida Hospital websites. Consequently, patient’s data (names, phone numbers, email addresses, height, weight, birthdates, insurance carriers and the last four digits of patients’ social security numbers) were compromised.45 |
|                  | Dictionary attacks, Rainbow table attacks        |                                                                                              |                                                                                                                                                                                                                                                                               |
|                  | Password recovery attacks                        |                                                                                              |                                                                                                                                                                                                                                                                               |
| Social           | Phishing                                        | Phishing, smishing, vishing, pharming, spear-phishing attack are social engineering attack where the user may be contacted by email, SMS, instant message, phone calls and so on by someone masquerading as a trustworthy establishment to lure the user into providing sensitive data such as social security number, passwords and so on. |                                                                                                                                                                                                                                                                               |
|                  | Smishing                                        |                                                                                              |                                                                                                                                                                                                                                                                               |
|                  | Vishing                                         |                                                                                              |                                                                                                                                                                                                                                                                               |
|                  | Pharming                                        |                                                                                              |                                                                                                                                                                                                                                                                               |
|                  | Spear phishing                                  |                                                                                              |                                                                                                                                                                                                                                                                               |
| Malware          | Worms                                           | Malware is a program that disrupts computer operation, gathers delicate information or accesses private frameworks without clients consent to intentionally cause harm or wreck the expected function of the system.                                         | The healthcare industry presents a lucrative target for cybercriminals/hackers who have in the past put patient’s sensitive data up for ransom, as selling the data can quickly translate into money. In 2016, the Hollywood Presbyterian Medical Center paid a huge US$17,000 ransom in bitcoin to cybercriminals who had hijacked and taken control of its computer network.46 |
|                  | Trojans                                         |                                                                                              |                                                                                                                                                                                                                                                                               |
|                  | Rootkits                                        |                                                                                              |                                                                                                                                                                                                                                                                               |
|                  | Adware                                          |                                                                                              |                                                                                                                                                                                                                                                                               |
|                  | Spyware                                         |                                                                                              |                                                                                                                                                                                                                                                                               |
|                  | Ransomware                                      |                                                                                              |                                                                                                                                                                                                                                                                               |
| Server           | N/A                                             | In this type of attack, the computer settings are compromised and the client’s request is redirected/diverted to a rogue server. Consequently, the rogue server will translate the request into a fake Internet protocol (IP) address that leads to a fake or malicious website.                            | Several other threats to EMR systems have been carried out using variants of the category of attacks described in this table.                                                                                                                                              |
| hijacking        | N/A                                             |                                                                                              | N/A                                                                                                                                                                                                                                                                          |
| Session          | N/A                                             | A session key or a legitimate computer session is exploited in order to gain unauthorized access to patient’s data or services in a computer system.                                                                                                                                    | N/A                                                                                                                                                                                                                                                                          |
| hijacking        | N/A                                             |                                                                                              |                                                                                                                                                                                                                                                                               |

EMR: electronic medical record.
Proposed approach

The proposed system is based on the intuition of honey/decoy system. It employs the standard honeyword generation system and proceeds further to generate fictitious personal/medical data which will be served to the adversary at the course of his attack. The definitions and notations used in this section are presented in Table 2 to allow understanding of the proposed system. Most of the definitions are taken from the previous studies.47–50

To give a general overview of the system model, a new user registers their user profile into the EHR system. During the registration process, a list of honeywords is created and attached to the patient’s profile and stored in the database. On the other hand, if the user has an existing user profile in the EHR system, then he or she can supply their user profile login details to authenticate into the system. If the user supplied a password that does not exist in the database, the user is denied entry into the system. However, if the password is found in the database, the honeychecker classifies the password to determine if it is a real password or a honeyword. The honeychecker authenticates the user into a genuine environment to access his or her data if the supplied password is a correct password. However, if the password is found to be a honeyword, then the user is authenticated into a fake environment where he or she can access fictitious data (HoneyDetails). An alarm with the audit trail of the user is activated during this phase.

The system administrator also collects and populates the patient’s data such as his name, age, address, blood samples and blood tests, accumulated visit to the hospital, diagnosis and prescriptions. The patient’s personal and medical details are populated into the database upon creating the patient’s account. A databank holds the genuine patient’s details, and an auxiliary or shadow database holds the list of honeywords associated with each user’s account. For instance, once a user submits his UserId $U_j$ to be enrolled into the system, a list of honeywords which are fake passwords will be created and stored in the shadow database. The user is prompted to select a SecretImage from a list of random images. User $U_j$ SecretImage is created as $SI_j$ and attached to a hashed password $H(P)$, $P_j$. Therefore, $U_j$, $SI_j$ and $P_j$ are created at enrollment. The idea of adding $SI_j$ was to address the typographical error limitation often encountered in honey systems. This is a situation whereby the legitimate patient made a mistake of submitting incorrect password that may fall in the range of a honeyword. The proposed system has been designed to produce fake personal and medical details if a user submits an incorrect password, that is, a honeyword. The patient will be able to tell that the record is not theirs when they are connected to data that do not bear a resemblance to their own. However, time will have been wasted and usability impeded during the process. We tackled the problem by introducing the $SI_j$ which will quickly point out the error to the legitimate user and save his or her time at the stage where he or she supplies his or her UserId. A flowchart showing an overview of the proposed model is depicted in Figure 2.

Figure 2 describes the system setup which is divided into two phases, the registration phase and the authentication/validation phase. The full details will be described in the following.

**Table 2. Definitions and notations.**

| Symbol          | Description                                                                 |
|-----------------|------------------------------------------------------------------------------|
| $H()$           | Cryptographic hash functions for computing hash value of the passwords       |
| $U_j$           | UserId for the $j$th user                                                   |
| $P_j$           | Password for the $j$th user                                                  |
| $SI_j$          | SecretImage for ($U_j$, $P_j$)                                              |
| Honeyword       | Fake/suspicious password                                                    |
| Real Password   | Correct password ($P_j$)                                                    |
| HoneyChecker    | A server that stores the index of the passwords and classifies the passwords |
| HoneyDetails    | Fake/fictitious personal and medical records                                |
In the registration phase, the patient’s UserId, password and a SecretImage are created. Assume $U_j$ represents the $j$th user for a computer system with $m$ users $U_1, U_2, \ldots, U_m$. $P_j$ denotes the password for the user with UserId $U_j$, and $SI_j$ represents the SecretImage for $(U_j, P_j)$. The patient chooses the data value for $U_j, P_j$ and $SI_j$. For simplicity, in this research design, we limited the length of $U_j$ and $P_j$ to be 15 characters at maximum and a minimum of 8 characters. The design choice of limiting the length of $U_j$ and $P_j$ was to allow the users enjoy the convenience of choosing low-entropy keys and also to make the implementation of the proposed system remain close to the current system so as not to impede usability. When the UserId has been selected and created by the patient, a list of honeyword is created automatically similar to previous studies. We omit the procedure for creating honeyword for simplicity and to avoid duplication. Any of the approaches of creating honeyword, such as Tweaking, Password models, Tough Nuts, Take-a-tail in previous studies, will serve the purpose of

---

**Figure 2.** A generic overview of the proposed method.
the honeyword creation. Honeywords are suspicious passwords that are close to the correct passwords. For instance, some attackers may use a dictionary containing passwords which users often use, such as “123456,” “iloveyou” and “abcdef”; usually, the attackers are able to get the dictionary based on past data breach details where the clear text passwords were published online. Also, there are cases where the attacker may try tweaking the passwords during his attack (tweaking, such as changing lowercase letters to uppercase letters or otherwise).

The second phase is the validation phase, where the patient (or an adversary) attempts to authenticate into the system using their registered (stolen or guessed) credentials. Algorithm 1 depicts the deciding factor if the user is to be redirected to a genuine environment with his or her details or the fake environment that produces the fictitious medical data.

**Algorithm 1. Authentication.**

**Input:** UserId, \( U_j \), password, \( P_j \), SecretImage, \( S_{lj} \)

**Output:** Genuine environment, fake environment

0. Start
1. User is prompted to submit \( U_j \) to the system
2. If \( U_j \) submitted is found in the genuine database
   a. A list of random images is displayed including the \( S_{lj} \) associated with \( U_j \)
   b. If User selects \( S_{lj} \) that corresponds to \( U_j \) in the system (Else go to Step 3a)
   c. Then User is prompted to submit \( P_j \)
   d. If \( P_j \) submitted by user corresponds with \((U_j, S_{lj})\), then the genuine environment is activated, and user is granted access into their account containing their medical data (Else go to Step 2c).
3. Else
   a. If \( U_j \) submitted in (Step 2) is found to be a honeyword in the shadow database
   b. A list of random images is displayed excluding \( S_{lj} \) associated with the genuine \( U_j \) from which the honeyword supplied by the user has been created. The concept here is to give a legitimate user that may have committed a typo error a chance to quickly retract the process when they cannot identify their correct \( S_{lj} \).
   c. If the user selects any random image at this point, then he will be prompted to supply his \( P_j \). Any password submitted will login successfully to the fake environment, and the user will be granted access into an account containing medical details of a patient. However, unknown to him that he is accessing fictitious data.
4. End

The proposed approach is different in that an incorrect password produces plausible data which are decoys but appear as realistic personal and medical details to mislead the attacker and to confuse him into thinking his hacking attempt was successful. Unlike current honey systems that generate fake passwords and load a fake session/environment which logs the activity of the criminals, we go further by supplying the adversary with fictitious medical data. The proposed system can also be useful for forensic purposes as the attacker can be caught in person when he tries to use the stolen data to make a fraudulent insurance claim.

**Building HoneyDetails using JavaScript API Faker.js**

The fictitious/fake data were created by leveraging the JavaScript API Faker.js to build on-the-fly realistic-looking but fictitious data that will be supplied to the adversary if he supplies an incorrect password (honeywords). Faker.js provides various categories to generate data. We collected the placeholder data/template used in the EHR system of National Hospital, Nigeria to build the fields required by the EHR system. Each field from the adopted template was created based on the structure of data expected as output. Details describing the process of building each field of HoneyDetails
can be found in the previous studies.51–54 We leverage on Faker.js to generate fake records for the attacker as it has a large database with fake user records. It supports about 37 languages. Therefore, the system can be customized to produce personal details for different countries/ethnicity based on their spoken language. Our adopted template contains generic but important fields used in the hospital to store each patient record. The Faker.js library generates specific data type, and as such, we manually set values to augment the fields that were not available in the library. Examples of such fields were patients’ genetic test results, prescriptions and so on. However, a full library to cater for robust medical data field for Faker.js is underway for our future work, but it is out of scope for this current work. A sample showing some data modeled in the template is shown in Table 3.

The pseudocode for building some structure of the Faker library is shown as follows. Details can be found in the previous studies 51–54 as mentioned earlier.

```javascript
var faker = require("faker")
var = {
    randomFirstName: faker.random.firstName(),
    randomMiddleName: faker.random.findName(),

```

| Database fields/details       | Elements                                                                 |
|-------------------------------|--------------------------------------------------------------------------|
| Patients personal information | First name, Middle name, Last name                                      |
|                               | Social security number                                                   |
|                               | Date of birth                                                            |
|                               | Phone number                                                             |
|                               | Email address                                                            |
|                               | Contact address                                                          |
|                               | Marital status                                                           |
|                               | Religion                                                                 |
|                               | Gender                                                                   |
|                               | Image                                                                    |
|                               | Psychological details                                                    |
|                               | Allergies                                                                |
|                               | Sexual orientation                                                       |
|                               | Vital signs                                                              |
|                               | Genetic testing results                                                   |
| Medical details               | Diagnostic history                                                       |
|                               | Disease name                                                             |
|                               | Prescriptions                                                            |
| Disease-related information   | Visit notes                                                              |
|                               | Admission/discharge details                                              |
| Hospital use information      | Name                                                                     |
|                               | Address                                                                  |
|                               | Mobile number                                                            |
| Next of kin details           |                                                                         |
|                               |                                                                         |
| Family medical history        |                                                                         |
| Notes and other reports       |                                                                         |

*Table 3. A medical database template showing patients’ medical details.*
randomLastName: faker.random.lastName();
randomSocialSecurityNumber: faker.random.number(),
randomDateOfBirth: faker.date.dob(),
randomPhone: faker.phone.phoneNumber(),
randomEmail: faker.internet.email(),
randomAddress: faker.address.streetAddress() + faker.address.city() +
  faker.address.country(),
randomImage: faker.random.image()
}

module.exports =

A random fake detail is created for each field. Other fields, such as medical histories and genetic results, were generated using faker.helpers to create robust fields. For instance,

randomMedicalHistory: faker.helpers.createMedicalHistory
randomVisitNotes: faker.helpers.createVisitNotes
randomPrescriptions: faker.helpers.createPrescriptions
randomReports: faker.helpers.createReports

Figures 5 and 6 in the supplementary file show a simulation of the proposed system. A generic scheme of the state-of-the-art system and the proposed scheme are illustrated in Figures 7 and 8, respectively, in the supplementary file.

**Performance/security of the proposed system**

The proposed system is easy to set up considering that the existing authentication system is maintained. Authentic users that supply their correct login details are conventionally logged into the system and connected to the genuine environment to access their data. Authentication for users who submit login detail which is not a honeyword is treated as standard authentication by been denied access into the system. An attacker/illegitimate user that supplies login details which is found to be honeyword is connected to a fake environment and supplied with fictitious patient’s details.

The storage cost of the proposed system is dependent on how the system is implemented. For the current study, the EHR system runs on a web server and the fictitious data are generated on the fly. Thus, there is no extra cost incurred in generating the fictitious data. Also, the generation of honeywords that may cause overhead as the index is saved can be optimized using several methods from Erguler47 to achieve flatness and cut costs.

**Performance evaluation of the security based on a Decoy Turing Test**

The proposed system was evaluated using a Decoy Turing Test (DTT). The DTT is a strategy used to show artificial intelligence through the inability of human judges to distinguish between machines and human simulators.55 A human judge engages a machine and a human in a discussion. The machine was said to have passed the test on the condition that the judge is unable to distinguish
it and the human. Following the DTT approach, the proposed system was evaluated using humans as the gold standard considering that attack on patient’s data is carried out by humans. The DTT test was carried out to verify if the fictitious data supplied to the adversary during his or her attack is able to fool him or her into accepting it as the genuine data.

We recruited 100 computer science students from the School of Computer Science, Universiti Sains Malaysia. Table 4 depicts the demographics of the participants recruited to test the proposed system.

We should state here that these students are more than the average users with a good knowledge of how we created the proposed system (this is because attackers are hardly novice but malicious people with a good knowledge of how most systems work). All of them were registered into the system and were asked to try to distinguish between original/genuine data and fictitious data. They were supplied with incorrect passwords and correct passwords. Each participant was instructed to make at least 40 attempts. They were supplied with a simple excel sheet to record a binary answer of yes or no for each authentication. Yes means the judges think a particular record is a fictitious data, and no means it is the genuine data. Figure 3 depicts the result for each one of the 100 participants (representing attackers). Collectively, a 100 percent failure rate was achieved. This implies that the fictitious data are indistinguishable from genuine data.

**Table 4.** Demographics of participants for the DTT test.

| Gender       | Number |
|--------------|--------|
| Male         | 57     |
| Female       | 43     |
| Total        | 100    |

| Age group | Number |
|-----------|--------|
| 20–24     | 21     |
| 25–35     | 41     |
| 36–46     | 38     |
| Total     | 100    |

| Educational qualification | Number |
|---------------------------|--------|
| Undergraduate (final year students) | 23     |
| Masters (in view)         | 37     |
| Doctorate (in view)       | 40     |
| Total                     | 100    |

Performance evaluation based on comparison with current cryptographic measures

The brute-force attack remains one of the most active attacks that plaques the current cryptographic schemes. Moore’s law states that computing power doubles every 18 months, while the costs remain constant. In cryptography, this implies that, if today we require a computer worth $1000 and a month to break a cryptographic scheme used in encrypting the data of patient’s Y, the cost for breaking the cipher Y in 18 months is $500, in 3 years is $250.56 Moreover, the brute-force attack is always possible when there are several pairs of plaintext–ciphertext. Trying the wrong keys yields gibberish that does not feature as the expected data of the patient. Thus, after exhausting every possible key, there remains only one genuine data that seems plausible which is the
Figure 3. Performance evaluation of the proposed system.
patient’s data. However, with HoneyDetails, every trial yields a plausible, genuine-looking (but fictitious) patient’s data.

In addition, current cryptographic schemes are designed based on mathematical theory (for instance, the Rivest–Shamir–Adleman (RSA) cipher is designed based on the difficulty of factoring out large prime integer). These schemes are used now as countermeasures for securing patient’s data, but the advent of quantum computers renders these ciphers insecure and unable to protect patient’s privacy.56,57 This is because the underlying mathematics used to secure current cryptographic schemes will be solved by the quantum computers in a jiffy. While there exists no known supercomputer or a large-scale quantum computer available now, the general conviction that it will be impossible to build quantum computers has been quashed, as the first quantum computers have seen the light of the day.58,59 This development implies that in a few years, months or weeks, large-scale quantum computers will be in existence, thus rending current applications insecure.

While we cannot directly compare our proposed approach with current cryptographic measures, we can show how current cryptographic measure fails to stop an adversary from stealing the encryption keys and acquiring the message. A simple mathematical description is presented as follows.

An adversary who intercepts a ciphertext will first determine the encryption scheme used and the language used for encryption. The encryption scheme is usually public knowledge; thus, determining the encryption scheme will not be a difficult task in the first place. For instance, an adversary that intercepted a ciphertext, \( C = 50 45 54 45 52 20 50 41 55 4c \), with the knowledge of the encryption scheme used to be Advanced Encryption Standard (AES), will further determine the language. If he knows that it is an English word, then he will try to brute-force the key using AES decryption to find out if the letters belong to the English alphabets

\[
C \leftarrow \text{enc} \left( P, K \right)
\]

Let’s assume that he has three six-digit keys to pick from

\[
M_{234561} = 2B 51 54 5B 4F 42
\]

\[
M_{953251} = 53 43 48 4F 4F 4C
\]

\[
M_{523434} = 26 45 4C 2F 46 51
\]

The effective way to distinguish between the keys is to check what each key corresponds to in the ASCII character dictionary. By the time he checks, he will find out that

\[
M_{234561} = 2B 51 54 5B 4F 42 = +Q\text{T}O\text{B}
\]

\[
M_{953251} = 53 43 48 4F 4F 4C = \text{SCHOOL}
\]

\[
M_{523434} = 26 45 4C 2F 46 51 = \&\text{EL} / \text{FQ}
\]
He will discard $M_{234561}$ and $M_{523434}$ as they do not show the randomness of natural language and, therefore, they do not spell out any meaningful word. Hence, he can easily pick out the correct key as $M_{953251}$ which spells out “school” and recover the message by applying the decryption function, $P \leftarrow \text{dec}(C, K)$, on the key and ciphertext

$$C = 50\; 45\; 54\; 45\; 52\; 20\; 50\; 41\; 55\; 4c = \text{PETER PAUL}$$

In summary, after exhausting all six-digit keys, the adversary finds only one message completely made of letters which forms a plausible meaning. Hence $k=953251$ and the plaintext message decrypts to “PETER PAUL,” which is the patient’s name. The adversary may use these tactics to recover all the private details of the patient. However, in our deception approach, for every key the adversary tries, he has plausible messages. For example, trying any incorrect key yields plausible messages such as EMMA JONES, NICK QUEEN and MARK MALT. In essence, an attacker trying to steal the patient’s data by trying random keys (password guessing or brute-forcing) in the current cryptographic approach gets gibberish which is an indicator that he has not yet gotten the data. However, in the proposed approach, trying random keys yields meaningful and plausible data to fool the adversary into thinking he has the data.

**Comparative experiment to check the usability of the proposed system compared to current system in use**

We also conducted a usability investigation of the proposed system using 30 users. Some of the users accessed the EHR system configured using their phones, and others used their laptops. We intended to evaluate how fast the system works and if it can be easily adopted in the real-world. We compared the current approach and the proposed approach. Each participant was instructed to authenticate into the system 10 times for uniformity. The average time used by each participant to

| Gender          | Number |
|-----------------|--------|
| Male            | 17     |
| Female          | 13     |
| Total           | 30     |

| Age group | Number |
|-----------|--------|
| 20–24     | 8      |
| 25–35     | 10     |
| 36–46     | 12     |
| Total     | 30     |

| Educational qualification | Number |
|---------------------------|--------|
| Undergraduate (final year students) | 9      |
| Masters (in view)          | 10     |
| Doctorate (in view)        | 11     |
| Total                      | 30     |
authenticate into the existing system and the proposed system is compared in order to determine if usability will be affected in terms of time. The demographics of the participants for evaluating the usability of the proposed system are provided in Table 5.

We show the time in seconds within which each participant took to authenticate to the existing and proposed method in Figure 4.

The arithmetic mean of the existing password-based system was 1.949 with a standard deviation of 0.232, while the arithmetic mean of the proposed system is 1.852 with a standard deviation of 0.224. While each participant authenticated with their devices and also using the Internet, it is apparent from the results that the minimal difference between both systems may have been due to several factors, for instance, how fast the Internet was at the time of authenticating and also the speed of the device.

**Discussion on limitation of the proposed approach**

The proposed system was tested in a setting where the fictitious data generated does not contain an image of the supposed patient. If the system is incorporated with the image of the patient during the decoy generation, then the proposed system may be weak to an individual attack where the adversary knows the face of his victim and can peruse the complete Oracle. For instance, in the setting where a fictitious image is generated, and an adversary is targeting a particular victim he knows, then HoneyDetails may fail to protect the patient’s data. This is because the adversary knows the person by face and can easily discard any record attached to an image that does not belong to his victim. The proposed system is more effective in a situation where the attacker has no target victim in mind, then it is challenging to differentiate fictitious from real data. However, one of the ways to bypass the limitation of the individual attack is to implement the system without a field that supplies the images of the patients.

This study is limited by the lack of a robust Fakers library for medical data. The medical domain as a whole contains extensive keywords and key terms relevant to the medical field alone. Such terms are yet to be incorporated in the Fakers library, and as such, certain database fields from the template were constructed manually. Unstructured data such as clinical notes and genetic details...
were constructed manually. The decoy generation limitation implies that more work is needed in terms of how to generate detailed decoys in the medical domain. This limitation can be addressed in future works by developing an extensive ontology-based keyword and glossary for the medical domain which will be integrated into the system to supply the patient’s decoy.

This study only considered words and texts generated in the English language. While the Fakers library contains about 37 languages, the researchers are native speakers of English and cannot translate the other languages or vouch if the other languages are language/culture accurate in this context.

HoneyDetails is an integration of the standard honey-based deception system, and so, it caters for other factors like logging in the activities and strategies of the attackers. We could not do a security and usability analysis of a large dataset. Thus, we are aware that our approach may have some other shortcomings even though it has been tested using a real-life dataset. To this end, we hope further research can be carried out on this novel decoy-based method to allow its full-adoption in the electronic health industry.

Conclusion

Cyber-criminals have significantly advanced in their ploys and grew their network to attack several financial, health and global institutions. They collaborate in groups using sophisticated tools and engineering methods to launch their attack to steal people’s sensitive data and sell in the underground world for real cash. Data stolen from the bank is rendered useless as soon as the breach is discovered and the account owner changes passwords, but data from the healthcare system which incorporate personal information, medical details and family/ancestral history can live a lifetime. The existing EHR security strategy mostly focuses on increasing the key, the complexity of the encryption algorithms and proffers a reactive response after the attacks have occurred. Such methods provide short-term solutions as an adversary is not deterred from brute-forcing the keys used in encrypting the patient’s details. Thus, enabling the adversary with sufficient time to test and learn our systems to launch attacks and achieve their goals within a short period and leaving little chance for defenders to defeat the attack actions. This research uses a proactive defense strategy that proffers long-term security using decoys and deception techniques to mislead the attacker. The proposed system generates fictitious patients’ information as a response to any hacking attempt with the specific goal of luring the attacker from the legitimate data. The result of the security evaluation shows significant improvement from state-of-the-art as the likelihood of a malicious person gaining successful access to patients’ medical information is impeded. Furthermore, the proposed system can be modified to suit any server hosting EHR systems.

To the best of our knowledge, the proposed work in this article is the first approach that uses a decoy-based defense system as a countermeasure for protecting patient’s information privacy in an EHR system. This research is worth further exploration considering the cases of recent and rather frequent incidents of massive theft of sensitive medical records.

Acknowledgements

The authors express their gratitude to the administrative staff of National Hospital Nigeria and the students of Security & Forensic Research Group (SFRG) Laboratory, School of Computer Sciences, Universiti Sains Malaysia, Penang for their material support during this study. Also, we thank reviewers of this work for their valuable comments and suggestions that improve the presentation of this research effort.

Author contributions

All authors listed have significantly contributed to the writing and development of this article.
Declaration of conflicting interests
The author(s) declared no potential conflicts of interest with respect to the research, authorship and/or publication of this article.

Funding
The author(s) disclosed receipt of the following financial support for the research, authorship and/or publication of this article: Center for Cyber Safety and Education, United States Internal Revenue segregated fund of (ISC) 2, Inc. Code. EIN: 45-2405127 through the (ISC)2 graduate cybersecurity scholarship award, 311 Park Place Blvd. Suite 610 Clearwater, FL 33759, United States. Likewise, partial funding from the Universiti Sains Malaysia RUI grant account number [1001/PKCOMP/8014017], Penang, 11800, Malaysia.

Informed consent
Informed consent was obtained from all individual participants included in this study.

ORCID iD
Abiodun Esther Omolara https://orcid.org/0000-0002-7801-2541

Supplemental material
Supplemental material for this article is available online.

References
1. Radwan AS, Abdel-Hamid AA and Hanafy Y. Cloud-based service for secure electronic medical record exchange. In: Proceedings of the 2012 22nd international conference on computer theory and applications (ICCTA), Alexandria, Egypt, 13–15 October 2012, pp. 94–103. New York: IEEE.
2. Cole AM, Stephens KA, West I, et al. Use of electronic health record data from diverse primary care practices to identify and characterize patients’ prescribed common medications. Health Informatics J 2020; 26(1): 172–180.
3. Tiwari B and Kumar A. Role-based access control through on-demand classification of electronic health record. Int J Electron Hlthc 2015; 8(1): 9–24.
4. Dinev T, Albano V, Xu H, et al. Individuals’ attitudes towards electronic health records: a privacy calculus perspective. In: Gupta A, Patel V and Greenes R (eds) Advances in healthcare informatics and analytics. Cham: Springer, 2016, pp. 19–50.
5. Brumen B, Hericko M, Sevenikar A, et al. Outsourcing medical data analyses: can technology overcome legal, privacy, and confidentiality issues. J Med Internet Res 2013; 15(12): e283.
6. Harman LB, Flite CA and Bond K. Electronic health records: privacy, confidentiality, and security. AMA J Ethics 2012; 14(9): 712–719.
7. Blightman K, Griffiths SE and Danbury C. Patient confidentiality: when can a breach be justified? Cont Educ Anaesth Crit Care Pain 2013; 14(2): 52–56.
8. Heather L. 2017 Breach report: 477 breaches, 5.6M patient records affected, 29 January 2018, https://www.healthcare-informatics.com/news-item/cybersecurity/2017-breach-report-477-breaches-56m-patient-records-affected (accessed 10 June 2018).
9. Ahmed M and Ahmad M. Combating abuse of health data in the age of eHealth exchange. In: Proceedings of the 2014 IEEE international conference on healthcare informatics (ICHI), Verona, 15–17 September 2014, pp. 109–118. New York: IEEE.
10. Enterprise.verizon.com. Protected health information data, 2019, https://enterprise.verizon.com/resources/reports/2018/protected_health_information_data_breach_report.pdf (accessed 7 July 2019).
11. Ponemon IR. Benchmark study on patient privacy and data security. J Healthc Prot Manage 2011; 27(1): 69–81.
12. Stallings W. *Cryptography and network security: principles and practice*. Upper Saddle River, NJ: Pearson, 2017, pp. 92–95.

13. Kiah MM, Nabi MS, Zaidan BB, et al. An enhanced security solution for electronic medical records based on AES hybrid technique with SOAP/XML and SHA-1. *J Med Syst* 2013; 37(5): 9971.

14. Li M, Yu S, Zheng Y, et al. Scalable and secure sharing of personal health records in cloud computing using attribute-based encryption. *IEEE T Parall Distr* 2012; 24(1): 131–143.

15. Bonneau J. The science of guessing: analyzing an anonymized corpus of 70 million passwords. In: *Proceedings of the 2012 IEEE symposium on security and privacy (SP)*, San Francisco, CA, 20–23 May 2012, pp. 538–552. New York: IEEE.

16. Florencio D and Herley C. A large-scale study of web password habits. In: *Proceedings of the 16th international conference on World Wide Web*, Banff, AB, Canada, 8–12 May 2007, pp. 657–666. New York: ACM.

17. Wu ZY, Lee YC, Lai F, et al. A secure authentication scheme for telecare medicine information systems. *J Med Syst* 2012; 36(3): 1529–1535.

18. Debiao H, Jianhua C and Rui Z. A more secure authentication scheme for telecare medicine information systems. *J Med Syst* 2012; 36(3): 1989–1995.

19. Wei J, Hu X and Liu W. An improved authentication scheme for telecare medicine information systems. *J Med Syst* 2012; 36(6): 3597–3604.

20. Omotesho A, Emuoyibo Farhe J and Meinel C. Ensuring patients’ privacy in cryptographic-based-electronic health records using bio-cryptography. *Int J Electron Hlthc* 2017; 9(4): 227–254.

21. Yu Z, Thomborson C, Wang C, et al. A cloud-based watermarking method for health data security. In: *Proceedings of the 2012 international conference on high-performance computing and simulation (HPCS)*, Madrid, 2–6 July 2012, pp. 642–647. New York: IEEE.

22. Lu Y, Li L, Peng H, et al. An enhanced biometric-based authentication scheme for telecare medicine information systems using elliptic curve cryptosystem. *J Med Syst* 2015; 39(3): 32.

23. Moon J, Choi Y, Kim J, et al. An improvement of robust and efficient biometrics based password authentication scheme for telecare medicine information systems using extended chaotic maps. *J Med Syst* 2016; 40(3): 70.

24. Al Hamid HA, Rahman SM, Hossain MS, et al. A security model for preserving the privacy of medical big data in a healthcare cloud using a fog computing facility with pairing-based cryptography. *IEEE Access* 2017; 5: 22313–22328.

25. Deshmukh P. Design of cloud security in the EHR for Indian healthcare services. *J King Saud Univ: Comput Inform Sci* 2017; 29(3): 281–287.

26. Zeadally S, Isaac JT and Baig Z. Security attacks and solutions in electronic health (E-health) systems. *J Med Syst* 2016; 40(12): 263.

27. Kwon J and Johnson ME. Protecting patient data-the economic perspective of healthcare security. *IEEE Secur Priv* 2015; 13(5): 90–95.

28. Boussif M, Aloui N and Cherif A. New watermarking/encryption method for medical imaging FULL protection in m-Health. *Int J Elect Comput Eng* 2017; 7(6): 3385–3394.

29. De Luca G, Brattstrom M and Morreale P. Designing a secure e-health network system. In: *Proceedings of the 2016 annual IEEE systems conference (SysCon)*, Orlando, FL, 18–21 April 2016, pp. 1–5. New York: IEEE.

30. Luo C, Fei Y, Luo P, et al. Side-channel power analysis of a GPU AES implementation. In: *Proceedings of the 2015 33rd IEEE international conference on computer design (ICCD)*, 18–21 October 2015, pp. 281–288. New York: IEEE.

31. Birukov A and Großschädl J. Cryptanalysis of the full AES using GPU-like special-purpose hardware. *Fund Inform* 2012; 114(3–4): 221–237.

32. Omolara AE, Jantan A and Abiodun OI. A comprehensive review of honey encryption scheme. *Indones J Elec Eng Comput Sci* 2019; 13(2): 649–656.

33. Jiang ZH, Fei Y and Kaeli D. A complete key recovery timing attack on a GPU. In: *Proceedings of the 2016 IEEE international symposium on high-performance computer architecture (HPCA)*, Barcelona, 12–16 March 2016, pp. 394–405. New York: IEEE.
34. Thangavel TS and Krishnan A. Performance of integrated quantum and classical cryptographic model for password authentication. In: Proceedings of the 2010 international conference on computing communication and networking technologies (ICCCNT), Karur, India, 29–31 July 2010, pp. 1–8. New York: IEEE.

35. Vandersypen LM, Steffen M, Breyta G, et al. Experimental realization of Shor’s quantum factoring algorithm using nuclear magnetic resonance. Nature 2001; 414(6866): 883–887.

36. Wang C and Lu Z. Cyber deception: overview and the road ahead. IEEE Secur Priv 2018; 16(2): 80–85.

37. Omolara A, Jantan A, Abiodun O, et al. A deception model robust to eavesdropping over communication for social network systems. IEEE Access 2019; 7: 100881–100898.

38. Krawetz N. Anti-honeypot technology. IEEE Secur Priv 2004; 2(1): 76–79.

39. Juels A and Ristenpart T. Honey encryption: encryption beyond the brute-force barrier. IEEE Secur Priv 2014; 12(4): 59–62.

40. Omolara AE, Jantan A, Abiodun OI, et al. A novel approach for the adaptation of honey encryption to support natural language message. In: Proceedings of the international multi conference of engineers and computer scientists (vol. 1), Hong Kong, 14–16 March 2018, http://www.iaeng.org/publication/IMECS2018/IMECS2018_pp134-139.pdf

41. Huang Z, Ayday E, Fellay J, et al. GenoGuard: protecting genomic data against brute-force attacks. In: Proceedings of the 2015 IEEE symposium on security and privacy (SP), San Jose, CA, 17–21 May 2015, pp. 447–462. New York: IEEE.

42. Park HS, Jeon YB and Yoon JW. A new approach to building a disguised server using the honey port against general scanning attacks. In: Barolli L, Xhafa F and Yim K (eds) International conference on broadband and wireless computing, communication and applications. Cham: Springer, 2016, pp. 453–465.

43. Ivanov C, Formby D, Litchfield S, et al. HoneyBot: a honeypot for robotic systems. P IEEE 2018; 106(1): 61–70.

44. Mukhopadhyay SK, Mitra S and Mitra M. ECG signal compression using ASCII character encoding and transmission via SMS. Biomed Signal Proces 2013; 8(4): 354–363.

45. Naseem S. Malware may have compromised some Florida Hospital Patient Information. Orlando Sentinel, 5 May 2018, https://www.orlandosentinel.com/health/os-florida-hospital-malware-websites-20180502-story.html (accessed 20 June 2018).

46. Winton R. Hollywood hospital pays $17,000 in bitcoin to hackers; FBI investigating, https://www.latimes.com/business/technology/la-me-ln-hollywood-hospital-bitcoin-20160217-story.html (accessed 6 February 2019).

47. Erguler I. Achieving flatness: selecting the honeywords from existing user passwords. IEEE T Depend Secure 2015; 13(2): 284–295.

48. Juels A and Rivest RL. Honeywords: making password-cracking detectable. In: Proceedings of the 2013 ACM SIGSAC conference on computer & communications security, Berlin, 4–8 November 2013, pp. 145–160. New York: ACM.

49. Juels A. A bodyguard of lies: the use of honey objects in information security. In: Proceedings of the 19th ACM symposium on access control models and technologies, London, ON, Canada, 25–27 June 2014, pp. 1–4. New York: ACM.

50. Catuogno L, Castiglione A and Palmieri F. A honeypot system with honeyword-driven fake interactive sessions. In: Proceedings of the 2015 international conference on high performance computing & simulation (HPCS), Amsterdam, 20–24 July 2015, pp. 187–194. New York: IEEE.

51. Generate massive amounts of realistic fake data in Node.js and the browser, https://github.com/marak/Faker.js/

52. A library for generating fake data such as names, addresses, and phone numbers, https://github.com/stumpy/faker/

53. Changes for version 0.07 - 2005-07-14, https://metacpan.org/release/JASONK/Data-Faker-0.07

54. The faker.js tutorial, http://zetcode.com/javascript/fakerjs/

55. Turing AM. Computing machinery and intelligence. Mind 1950; 59: 433–460, https://phil415.pbworks.com/f/TuringComputing.pdf
56. Paar C and Pelzl J. *Understanding cryptography: a textbook for students and practitioners*. Heidelberg: Springer, 2009.

57. Shor PW. Polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computer. *SIAM Rev* 1999; 41(2): 303–332.

58. Jones JA, Mosca M and Hansen RH. Implementation of a quantum search algorithm on a quantum computer. *Nature* 1998; 393(6683): 344–346.

59. Veldhorst M, Yang CH, Hwang JC, et al. A two-qubit logic gate in silicon. *Nature* 2015; 526(7573): 410–414.