A Differential Privacy Mechanism for Deceiving Cyber Attacks in IoT Networks

A Differential Privacy Mechanism for Deceiving Cyber Attacks in IoT Networks

Guizhen Yang[1], Mengmeng Ge[2], Shang Gao[1], Xuequan Lu[1], Leo Yu Zhang[1], and Robin Doss[1]

[1] Deakin University, Geelong Waurn Pounds, Australia
{guizhen.yang, shang.gao, xuequan.lu, leo.zhang, robin.doss}@deakin.edu.au
[2] University of Canterbury, Christchurch, New Zealand
{mge43}@uclive.ac.nz

Abstract. Protecting Internet of Things (IoT) network from private data breach is a grand challenge. Data breach may occur when networks' statistical information is disclosed due to network scanning or data stored on the IoT devices is accessed by attackers because of lack of protection on IoT devices. To protect IoT networks, effective proactive cyber defence technologies (e.g., Moving Target Defence (MTD) and deception) have been proposed. They defend against attacks by dynamically changing attack surface or hiding true network information. However, little work considered the protection of statistical information of IoT network, such as the number of VLANs or the number of devices across VLANs. This type of information may leak the network's operational information to attackers (e.g., functional information of VLANs). To address this problem, we propose a differential privacy (DP)-based defence method to mitigate its leakage. In this paper, we strategically obfuscate VLANs' statistical information by integrating DP with MTD and deception technologies. Software-defined networking technology is leveraged to manage data flows among devices and support shuffling-based MTD. Two strategies (random and intelligent) are considered for defence deployment. A greedy algorithm is designed to explore the trade-off between defence cost and privacy protection level. We theoretically prove that the proposed method meets the definition of DP, thus offering solid privacy protection to the operational information of an IoT network. Extensive experimental results further demonstrate that, for a given defence budget, there exists a trade-off between protection level and cost. Moreover, the intelligent deployment strategy is more cost-effective than the random one under the same settings.

Keywords: Differential privacy · Internet of Things · Deception · Moving target defence · Software defined networking.

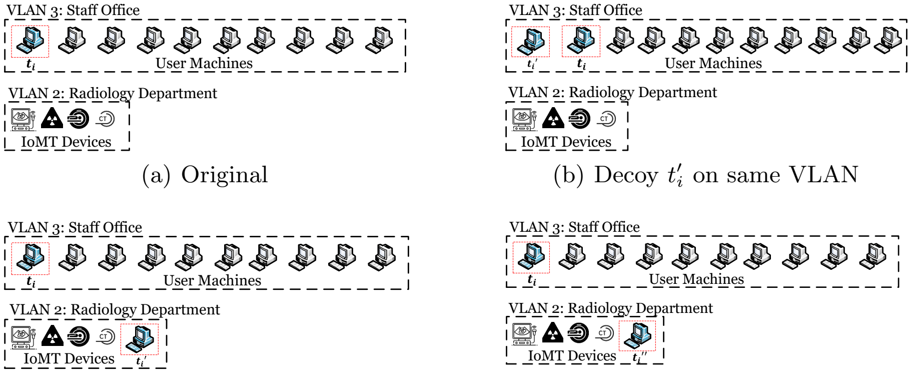
1 Introduction

Internet of Things (IoT) is a network of physical objects (e.g., devices, instruments, vehicles, buildings and other items) embedded with electronics, circuits, sensors and network connections to collect and exchange data [1]. It allows these objects to communicate by wired or wireless communications (e.g., Bluetooth, ZigBee and 5G) and share data across existing network infrastructure [2]. Nowadays, IoT networks are growing rapidly and contain around 28 billion objects [3] that communicate with each other. They bring a lot of benefits to human beings but also provide opportunities for attackers to collect valuable information and launch attacks which may severely impact operations and normal functionality of IoT devices [4] [5]. Thus, effective protection and defence mechanisms are needed to protect the IoT networks from potential attacks.

In an IoT network, devices are usually grouped into different VLANs according to their functions and/or locations [6]. Statistical information, such as the total or average number of devices per VLAN, or the number of VLANs, may leak the network's operational information, or expose potential attack targets. For example, a Radiology department often has a limited number of medical imaging devices due to the budget limitation. The medical imaging devices collect data from patients and upload it to the server where doctors access patients' data for diagnosis purposes. Attackers may be able to deduce the VLAN where these medical imaging devices are located based on the network statistical information (e.g., the number of user machines in a VLAN is potentially much larger than the number of critical devices, such as medical imaging devices or servers, in a separate VLAN). Once attackers identify a vulnerable Internet-of-Medical Things (IoMT) device or a vulnerable user machine, they can pivot toward the server. The consequences are expensive if patients' private information is disclosed. Therefore, it is necessary to minimise the leakage of statistical information, increasing the difficulty of deducing potential targets by the attackers, preventing them from breaking into the network and launching further attacks.

However, there is little prior work considering the security issues caused by the leakage of statistical information. The motivation of our work lies within the privacy protection of statistical information of IoT networks. We aim at developing a defence method to obfuscate the statistical information of an IoT network and mislead attackers. Fig. 1 shows two examples of VLAN obfuscation. In a healthcare IoT network, we assume there is a vulnerable device t_i (in red square, belonging to a radiologist) on VLAN 3 (i.e., the staff office). It has direct access to a file server on VLAN 5 which stores medical imaging data from multiple medical devices on the network. If t_i is compromised, the attacker may pivot and compromise the server.

Intuitively, to prevent the attacker from locating the server through devices in other VLANs, we can increase the attack cost of identifying a potential target by hiding true network information. It can be done by increasing the device diversity, such as deploying decoy t_i' (1(b) and 1(c)), or by changing the attack surface, such as shuffling t_i's IP address to make it appear on VLAN 2 but not on VLAN 3 (1(d)). When the attackers scan the network, they may not accurately deduce the operational information of VLANs because of the obfuscated virtual



(a) Original

(b) Decoy t_i' on same VLAN

(c) Decoy t_i' on different VLAN

(d) t_i mapped to t_i'' by IP shuffling

Fig. 1. Examples of VLAN obfuscation.

location of t_i. Hence, the attack cost is increased and the operational information of the network is protected to some extent.

However, this kind of simple deception methods do not protect the operational information well against the attacker with stronger background (i.e., side channel information). For the scenario considered above, this background knowledge can be derived from the simple fact that the probability of t_i on VLAN 3 is much higher than that of the decoy t_i' on VLAN 2 (Fig. 1(c)). That said, by simply counting the number of devices on both VLANs, the attacker can distinguish real t_i and fake t_i' or t_i'', and easily workaround the aforementioned deception methods. From this view, it is necessary to also obfuscate the VLAN's statistical information (e.g., the number of devices) to offer protection against informed attackers.

In this paper, we first use differential privacy (DP) to obfuscate VLAN's statistical information. The number of devices per VLAN is strategically changed under a given privacy budget ϵ. Two defence mechanisms are then applied to achieve the obfuscation. Finally, we adopt a greedy algorithm to optimise the deployment of defence choices and find the trade-off between defence cost and privacy budget. The defence mechanisms used in this work include: (1) deception technology to deploy decoys into network [7], and (2) IP-shuffling based Moving Target Defence (MTD) technology to obfuscate the attack surface [8].

The main contributions of this paper are summarised as follows:

– We are the first to integrate DP with software-defined networking (SDN)-based MTD and deception technologies to solve the security issues caused by leakage of statistical information of IoT network. Using VLAN information as an example, we add the Laplace noise to the number of devices per VLAN and obtain the obfuscated set of VLANs, based on which, deception technologies can be further applied by the defender.

- We design a greedy algorithm to find optimal trade-offs between privacy budget and defence cost under given defence budgets.
- Extensive experimental results validate the effectiveness and scalability of the proposed method on different scaled IoT networks.

The rest of the paper is organised as follows. Section 2 introduces related work. Section 3 presents the proposed method. System model, attack model and defence model are described in section 4. Experimental results and their analysis are given in section 5. Section 6 discusses the limitations and future directions.

2 Related Work

MTD technologies: Moving target defence [8–10] is one of the common proactive defence mechanisms that has emerged to deceive potential attacks [11,12]. It aims at hurdling attacks by constantly changing the attack surface. With MTD, the complexity, diversity and randomness of systems or networks are increased to disrupt attackers' actions during the reconnaissance phase of cyber kill chain. There are three common MTD techniques: shuffling, redundancy and diversity. In our case, we use IP shuffling which is a common network shuffling technique for an increased complexity of the IP address space.

Ge et al. [13] re-configured the IoT network topology to deal with non-patchable vulnerabilities. By maximising the number of patchable nodes along the route to the base station, the attack effort is increased while maintaining the average shortest path length. The work [14] considered hybrid approaches by combining different defence mechanisms. Decoys are strategically deployed into the network and a patch management solution is applied to solve unpatchable vulnerabilities under a constraint budget. Further study in [15] proposed an integrated defence technique for intrusion prevention. It explains "when to move" and "how to move". The former performs network topology shuffling with four strategies (i.e., fixed/random/adaptive/hybrid), and the latter shuffles a decoy IoT network with three strategies (i.e., genetic algorithm/decoy attack path-based optimisation/random).

The work [16] randomly shuffled communication protocols in an IoT network. It solves problems such as "what to move" by utilising moving parameters to determine shuffled protocol, "how to move" by using a discrete & uniform probability distribution to determine the next moving parameter, and "when to move" by adopting fixed or random time interval. The paper also analyses multi-criteria to find a trade-off among system performance, business impact and the success probability of a given attack.

Deception techniques: Deception techniques aims to mislead and deceive attackers [7,17]. Honeypot [18,19] is one of the commonly exploited technologies in cyberdeception. It is created as a fake asset and shows attackers misleading information, luring them into the honeypot environments to divert them. However, how to create honeypots, how many to deploy, when and where to deploy are still unsolved problems [12].

La et al. [19] formulated a Bayesian Game model to reflect the defender's imperfect knowledge of the incoming user's type (e.g., malicious or not). It addresses the key issues of how the defender reacts different observations on the attacker and which deception strategies are optimal to both the attacker and defender. However, It does not take the false positive and false negative into account to measure the effective results. Tsemogne et al. [20] introduced a zero-sum one-sided partially observable stochastic game model to investigate cyberdeception techniques against botnets propagation in IoT networks.

Ye et al. [21] presented a differentially private game-theoretic approach for cyber deception. They use DP to dynamically change the number of systems and obfuscate their configurations in three situations: reallocating system configurations, deploying decoys and taking systems offline. However, limitations, like how to guarantee the number of systems to be taken offline fewer than that of the current live hosts, are not addressed. Additionally, taking systems offline is irrational in reality due to due to interrupting normal services from systems.

This paper focuses on solving the security issues caused by the leakage of statistical information of IoT networks. Our work is also built on top of DP, but it obfuscates IoT network's statistical information (e.g., the number of devices) to resist informed attackers through the integration of MTD and deception technologies.

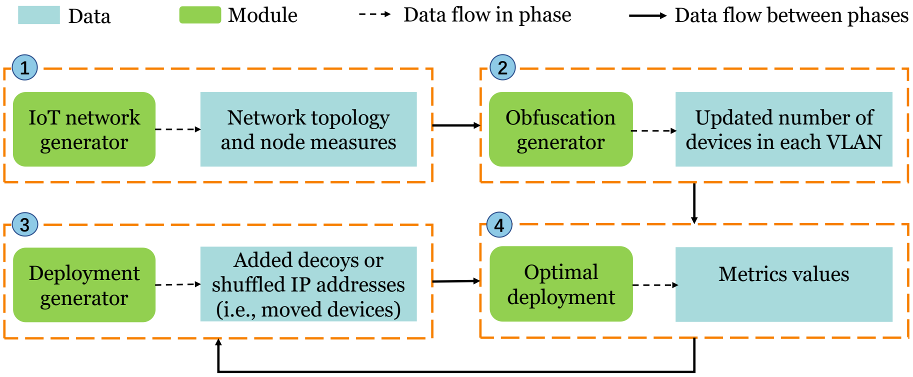


Fig. 2. Workflow of the proposed approach.

3 Proposed Approach

To effectively manage nodes (devices) and their data flows, we utilise SDN technology [22]. An SDN controller communicates with SDN switches in the IoT network. Servers, user machines and IoT devices are connected to the SDN switches. The switches transform data flows to the SDN controller for further processing. We also leverage the SDN technology to deploy shuffling-based MTD, as well as managing the communication between the SDN controller and switches via virtual-to-real or real-to-virtual IP addresses mapping.

The overall workflow of our proposed approach is shown in Fig. 2. It consists of 4 phases: IoT network model generation, information obfuscation by DP mechanism, deployment strategy generation using IP shuffling and deception, and deployment optimisation by finding the trade-off between defence cost and privacy budget.

In **Phase 1**, we generate the system model along with node measures based on the network information. In specific, the IoT network generator takes network topology as input with node connectivity information. The output is statistical information of the network (e.g., set of devices in each VLAN and set of VLANs on a given IoT network) and node measure (e.g., the betweenness centrality (BC) of each node that captures how much a given node is in-between other nodes [23]).

In **Phase 2**, we adopt DP to obfuscate the statistical information of VLANs. The obfuscation generator takes the statistical information of the network (e.g., the number of devices in each VLAN) and the node measures from **Phase 1** as input, and adds Laplace noise to the set of devices per VLAN under the DP framework. The output is the updated number of devices in each VLAN. We denote the original IoT network as N and the new obfuscated IoT network as N'. For each VLAN k, the number of devices is changed from $|N_k|$ to $|N'_k|$ after obfuscation. There is a possibility that $|N'_k| < 0$ since the Laplace noise is a random variant. It means that the devices moved out from VLAN k is large than the devices on VLAN k, which is violated in the real world. Hence, we use $\Delta N^*_k = |N'_k| - |N_k|$ as an optimal set of devices that should be moved out from or added into VLAN k to solve this problem, aiming at adapting to real-world scenarios, including guaranteeing the number of devices moved out is less than that of the current live devices. These to-be-moved devices are also candidates for IP-shuffling based MTD in **Phase 3**. Therefore, we call ΔN^*_k is an optimal set of devices after the improved obfuscation.

In **Phase 3**, we use the deployment generator to update the network information as specified by the updated set of devices for each VLAN from the output of **Phase 2** and the original set of devices per VLAN. The deployment generator deploys the defence strategies produced by the randomisation module or the optimisation module in **Phase 4**. The output is the updated IoT network after the deployment of defences. We use the MTD technology - IP address shuffling to change the attack surface and the deception technology - decoy to mislead attackers. As mentioned earlier, we leverage SDN to implement shuffling-based MTD. In our proposed approach, the SDN controller is also used to manage SDN switches. The SDN switches are used to forward packets to the SDN controller for handling the data flow. control packet forwarding. We assume each device has a real IP address (rIP). The rIP is mapped to one virtual IP address (vIP) which is selected from a group of randomly generated virtual IP addresses (vIPs). Only the SDN controller and the device know its rIP, while other devices in the IoT network use the mapped vIP to communicate with the device. The mapping between rIP to vIPs is managed by the SDN controller [24, 25].

In **Phase 4**, we consider two strategies for defence deployment: random and intelligent strategies. The randomisation module starts by randomly selecting a VLAN k. Depending on the obfuscation outcome of **Phase 2**, it either randomly selects a device for IP shuffling or deploys a decoy on VLAN k. The intelligence module starts by selecting a VLAN k based on their criticality. Depending on the obfuscation outcome of **Phase 2**, it then selects a device for IP shuffling or deploys a decoy on VLAN k based on the devices' or decoys' criticality. In particular, if the number of devices on VLAN k increases (i.e., $\Delta N_k^* > 0$), the defender deploys ΔN_k^* number of decoys into VLAN k; If the number decreases (i.e., $\Delta N_k^* < 0$), the defender moves $|\Delta N_k^*|$ number of devices from VLAN k to another VLAN by shuffling the IP addresses of these devices to that VLAN. As discussed in **Phase 3**, the SDN controller can shuffle vIP addresses[3]. We develop a greedy algorithm (GA) to compute the optimal deployment by exploring the trade-off between defence cost and privacy protection level. GA aims to minimise the defence cost under different privacy budgets at each stage. The selection of VLAN, decoy or device can be determined by their criticality. The higher criticality the object has, the higher priority it takes.

We consider the security metric as the defender's cost. It is the total cost of deploying decoys to the IoT network and moving devices from one VLAN to another (e.g., shuffling IP addresses). The privacy protection level is determined by the privacy budget used in **Phase 2** (i.e., how much Laplace noise to add for information obfuscation).

4 System, Attack and Defence Models

This section describes the system model, attack model and defence model of our proposed approach.

4.1 System Model

We consider the internal network of a smart healthcare network as shown in Fig. 3. The internal network consists of servers, a SDN controller and SDN switches, user machines (e.g., computers) and IoT devices. Assume traditional defence techniques are in place, including intrusion detection systems (IDS) and anti-virus software. The servers, user machines and IoT devices are connected to the SDN switches.

4.2 Attack Model

In this smart healthcare network, servers are used to store patients' medical records (e.g., medical diagnosis reports and radiological images). If they are

[3] IP shuffling does not change the VLAN that a device resides nor affect the communications between the device and other devices; but it gives attackers a different network view in their reconnaissance phase.

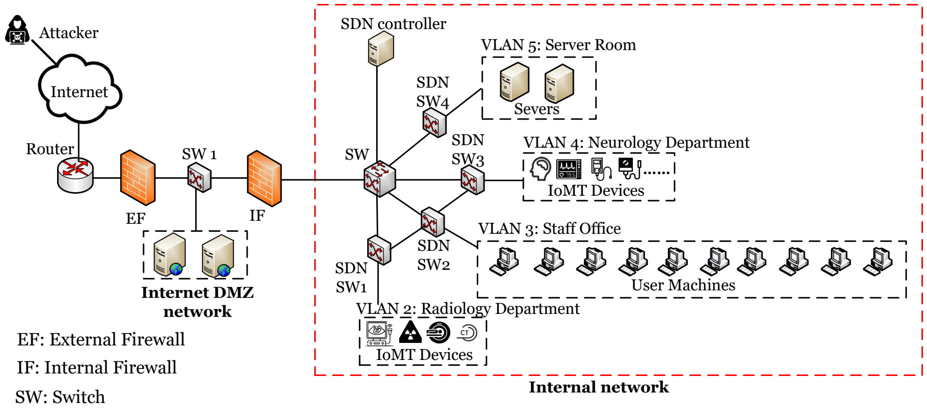


Fig. 3. Example of Software-Defined healthcare IoT network.

compromised, attackers may steal the data for economic gain [26]. The disclosure of patients' data can pose a serious threat to the health and safety of individuals. Therefore, we assume these servers could be potential attack targets.

Based on the observation of real-world scenarios, we assume the attackers have the following capabilities.

- Attackers can leverage various scanning tools to collect information about the target network (e.g., number of devices, network topology and operating system of a host) and identify weaknesses for exploitation (e.g., known and zero-day vulnerability). Attackers are able to utilise some firewall/IDS evasion techniques to avoid blocking and detection.
- Attackers may be able to identify attack targets with less time or cost by analysing the collected information. For example, the real location of one device can be deduced by comparing gathered information (e.g., counting the number of changed devices per VLAN) before an attack is launched on the device.
- It is highly unlikely for attackers to directly compromise the servers as they are assumed to be well-protected due to the traditional defence techniques placed on the network.
- Attackers lack knowledge of existence of decoy system. Once the attackers realise the device they interact with is a decoy, they terminate the interaction immediately and attempt to find a new target.
- Attackers can not compromise the SDN controller and SDN switches which are assumed to be secure.

4.3 Defence Model

The proposed defence model covers two stages: the obfuscation of VLANs through DP, and the deployment of two defence technologies for obfuscation. Table 1 lists the notations used in this work.

Table 1. Notations and definitions.

Notations	Definitions
N	set of devices in a network
N'	set of devices in a network after obfuscation
K	set of VLANs
N_k	set of devices in VLAN k
N'_k	set of devices in VLAN k after obfuscation
$\varDelta N_k^*$	an optimal set of devices moved out from or added into VLAN k after improved obfuscation
$m1(k,j) \rightarrow 0,1$	function to move a device from VLAN k to VLAN j with MTD
$m2(d,k) \rightarrow 0,1$	function to deploy a decoy d into VLAN k with deception
$c1(k,j)$	cost of moving a device from VLAN k to VLAN j when $m1(k,j) = 1$
$c2(d,k)$	cost of deploying a decoy d into VLAN k when $m2(d,k) = 1$

Obfuscation of VLANs: Differential privacy is a framework that provides theoretical guarantee to protect any individual record in a dataset from leaking or being exploited by all possible informed attackers [27]. DP has been successfully applied to the network security field [21, 28]. However, existing studies on DP-based network security solutions still have limitations. In this work, we try to address the privacy security issues caused by leakage of statistical information of IoT networks and overcome the limitations we have found in [21] when applying DP to obfuscate the statistical information of networks.

Briefly speaking, DP operates on two neighbouring datasets D and D' which differ by only one record. For a query function f that maps dataset D to an output value range $\mathbb{R}$, $f : D \rightarrow \mathbb{R}$, DP randomises this query f such that its output from D and D' cannot be distinguished by any attacker. The maximal difference of the query function f is called sensitivity $\varDelta S$ ($\varDelta S = \max_{D,D'} \|f(D) - f(D')\|_1$), which affects the level of noise added during randomisation. The formal definition of DP is given as follows.

Definition 1. ***(ϵ-Differential Privacy [29]).*** *An algorithm $\mathcal{A}$ provides ϵ-differential privacy for any pair of neighbouring datasets D and D' if the algorithm $\mathcal{A}$ satisfies:*

$$\Pr[\mathcal{A}(D) \in \varOmega] \leq \exp(\epsilon) \cdot \Pr[\mathcal{A}(D') \in \varOmega] \tag{1}$$

where $\varOmega$ is the output of algorithm $\mathcal{A}$ performing on D (or the neighbouring D'), ϵ is the privacy budget, ranging from 0 to 1 for counting queries. It controls how much noise or randomness is added to the raw data. The higher the ϵ value, the lower the noise is; or the lower the ϵ value, the higher the noise is.

There are two most widely used mechanisms to achieve DP (i.e., Laplace and Exponential). We focus on the Laplace mechanism, which adds Laplace noise to the true query answer (i.e., the number of devices per VLAN). Denote Lap(b) the noise sampled from the Laplace distribution with scaling b, we have the following definition and properties.

Algorithm 1 Strawman obfuscation approach of VLANs

```
Input: $|N_1|, \cdots, |N_k|$
Output: $|N'_1|, \cdots, |N'_k|$ and $\Delta N^*_1, \cdots, \Delta N^*_k$
1: Calculate the minimum value of set of devices on VLANs $a_{min} : |N_1|, \cdots, |N_k|$
2: for $k = 1 \to K$ do
3:     $|N'_k| \leftarrow |N_k| + \lceil Lap\left(\frac{\Delta \cdot K}{\epsilon}\right)\rceil$
4:     if $|N'_k| \leq 0$ then
5:         $|N'_k| \leftarrow a_{min}/2$
6:     end if
7:     $\Delta N^*_k \leftarrow |N'_k| - |N_k|$
8: end for
9: Obtain $|N'_1|, \cdots, |N'_k|, a_{min}, \Delta N^*_1, \cdots, \Delta N^*_k$.
```

Definition 2. ***(The Laplace Mechanism [29]).*** *Given a function $f :\to \mathbb{R}$ over a dataset D, then*

$$\widehat{f}(D) = f(D) + \text{Lap}\left(\frac{\Delta S}{\epsilon}\right) \tag{2}$$

satisfies ϵ-DP.

Theorem 1. ***Sequential composition [29]:*** *Suppose there are k algorithms $\mathcal{A}_1, \mathcal{A}_2, \cdots, \mathcal{A}_k$ that satisfy ϵ_1-DP, ϵ_2-DP, $\cdots$, ϵ_k-DP, respectively, with respect to the input dataset D. Publishing $t = (o_1, o_2, \cdots, o_k)$, which is the output value of the algorithms performing on D, satisfies $(\sum_{i=1}^{k} \epsilon_i)$-DP.*

Theorem 2. ***Post-processing [29]:*** *Suppose an algorithm $\mathcal{A}_1(\cdot)$ satisfies ϵ-DP, then for any algorithm $\mathcal{A}_2$, the composition of $\mathcal{A}_1$ and $\mathcal{A}_2$, i.e., $\mathcal{A}_2(\mathcal{A}_1(\cdot))$, satisfies ϵ-DP.*

Based on DP's definition and properties, we present a strawman obfuscation approach for the VLANs in Algorithm 1. In line 3, Laplace noise is added to each $|N_k|$ to obfuscate the number of devices on VLAN k. In our case, Δ is the the maximum change (i.e., shuffling a device's IP address or deploying a decoy) made to VLANs. By definition, $\Delta = 1$.

However, as the Laplace noise is a random variant, there is a possibility that $|N'_k| < 0$, which means the number of devices needs to be moved out of one VLAN (i.e., by IP address shuffling) may be greater than the number of devices in that VLAN, which violates reality. Therefore, lines 4-6 are used to limit the number of devices left in VLANs (after moving devices out) to be one half of the original minimum number of devices in the VLANs. In this way, the issue of how to guarantee the number of systems to be taken offline fewer than that of the current live devices from [21], can be avoided naturally.

On the other hand, another extreme situation, where the total number of devices to-be-moved is greater than the total number of available decoys (under a certain budget), may occur. In this case, the extra devices to-be-moved need to be taken offline to ensure DP. Clearly, in our example application, it is irrational

Algorithm 2 Improved VLAN obfuscation approach

```
Input: $\mathit{\Delta} N_1^*, \cdots, \mathit{\Delta} N_k^*$
Output: Updated values of $\mathit{\Delta} N_1^*, \cdots, \mathit{\Delta} N_k^*$
 1: for $k = 1 \to K$ do
 2:     if $\mathit{\Delta} N_k^* < 0$ then
 3:         $moveOut \leftarrow N_k^*$
 4:     else
 5:         $moveIn \leftarrow N_k^*$
 6:     end if
 7: end for
 8: $sumOut \leftarrow \mathrm{abs}(\mathrm{sum}(moveOut))$;
 9: $sumIn \leftarrow \mathrm{sum}(moveIn)$
10: if $sumOut > sumIn$ then
11:     for $i = 0 \to \mathrm{len}(moveOut)$ do
12:         $temp = moveOut[i]$
13:         $moveOut[i] = (\frac{moveOut[i]}{sumOut}) * sumIn$
14:         $\mathit{\Delta} N^*_{\mathit{\Delta} N^*.index(temp)} = moveOut[i]$
15:     end for
16: end if
17: Obtain $\mathit{\Delta} N_1^*, \cdots, \mathit{\Delta} N_k^*$.
```

to take devices offline as this will disrupt services for patients. Therefore, we design Algorithm 2 to deal with this situation. In line 13, we regard the number of devices moved per VLAN as being proportional to the total number of fake assets added.

In this way, our approach not only ensures the satisfaction of DP but also avoids taking devices offline, which is an irrational defence method to protect devices from [21].

Proposition 1. *Both Algorithm 1 and Algorithm 2 satisfy ϵ-DP.*

Deployment of defence mechanisms: After obtaining the outcomes of VLAN obfuscation, the remaining task is to implement the obfuscation of VLANs to deceive attackers. As discussed earlier, we use MTD and deception technologies to achieve the obfuscation. Without loss of generality, we assume the cost of moving one device out of one VLAN to another VLAN is cheaper than adding a decoy, so the defender prefers IP-shuffling based MTD to cyber deception.

Deception: The defender utilises deception technology if the number of devices in VLAN k increases (i.e., $\mathit{\Delta} N_k^* > 0$). The defender deploys $\mathit{\Delta} N_k^*$ number of additional devices on VLAN k for deceiving possible attackers. The additional devices can be decoys or devices that are moved from other VLANs if those VLANs have a reduced number of devices after obfuscation, or a mix of decoys and devices from other VLANs.

To minimise the cost while maximising security and preserving privacy, we need to decide (1) which VLAN should be selected first for deploying decoys and (2) what decoys should be deployed. In particular, we select a VLAN based on its criticality. Take the healthcare network as an example, we consider the total cost of devices per VLAN as the VLAN's criticality because critical devices like servers and medical devices often have a higher price. We consider the cost of decoy deployment based on the license fees by decoy type [14,30]. A decoy at a lower price will have a higher priority being deployed into a VLAN.

To increase the chance of decoy interacting with attackers, the decoys should resemble the devices in the real network [14]. In our case study, we choose real OS for server decoys (i.e., full OS-based decoy) and emulator software for other device decoys (i.e., emulated decoy). Full OS-based decoys have higher interaction capability with attackers but will incur a higher cost than the emulated ones.

Shuffling: The defender resorts to MTD if the number of devices in VLAN k decreases (i.e., $\Delta N_k^* < 0$). The defender moves $|\Delta N_k^*|$ number of devices from VLAN k to another VLAN by shuffling IP addresses of these devices in that VLAN.

In particular, we select devices based on their criticality. Betwenness centrality (BC) is considered which measures the centrality of a node (device) in a graph (topology) based on the shortest path [31]. BC can be formulated as

$$BS(t_i) = \sum_{t_s \neq t_d} \frac{\sigma_{sd}(t_i)}{\sigma_{sd}}, \tag{3}$$

where σ_{sd} is the total number of shortest paths from the source nodes t_s to the destination node t_d. The $\sigma_{sd}(t_i)$ is the number of those paths that pass through the node t_i. The node/device with a higher BC plays a more important role in the network and is more likely to be an attack target. Therefore, in this work, a device with a higher BC will be shuffled first.

Motivated by the studies on IP address shuffling [24,25], we adopt this MTD technology in our work. When needed, a device's rIP can be randomly mapped to one vIP from a pool of $(|K| - 1)$ vIPs. As the noise introduced by DP is random, the device of the VLAN under study can be shuffled to any other $(|K| - 1)$ VLANs. Before shuffling, a new set of $(|K| - 1)$ vIPs for each device is randomly generated. The communication process between the source device and destination device under IP shuffling-based MTD (i.e., rIP-to-vIP mapping and vIP-to-rIP mapping) as bellow.

As we mentioned in **Phase 3** in Sec. 3, only the SDN controller and the device itself know their rIPs while other devices use vIPs to communicate with each other. When the source device sends a packet to the nearest SDN switch, the SDN switch transmits the packet to the SDN controller. The SDN controller receives the packet to map vIP to rIP from the packet header information, and updates the flow-table entry of all SDN switches (e.g., Open-Flow-Switches). Each switch uses the flow rules to convert the rIP into the vIP in the packet header. The SDN switch near the the destination device convert the vIP of the destination device in a packet header to its rIP. Hence, both the source device and destination device do not know each other's rIP and the mapping is transparent to an end device with no service disruption since rIPs of devices remain unchanged [24, 25].

Optimal Defence Deployment: Deploying any defence technology can reduce the security risk of a given IoT network, but it may incur costs as well. We treat this cost-increasing problem as an optimisation problem. Due to the discrete nature of our problem formulation, we use greedy algorithm to find the trade-off between the defence cost and the privacy protection. The objective function is

$$\begin{aligned} &\underset{C1(k,j),C2(d,k)}{\arg\min} \quad M1(k,j) * C1(k,j) + M2(d,k) * C2(d,k) \\ &\text{s.t.} \quad \begin{cases} M1(k,j) \in \{0,1\}, \\ M2(d,k) \in \{0,1\}, \end{cases} \end{aligned}$$

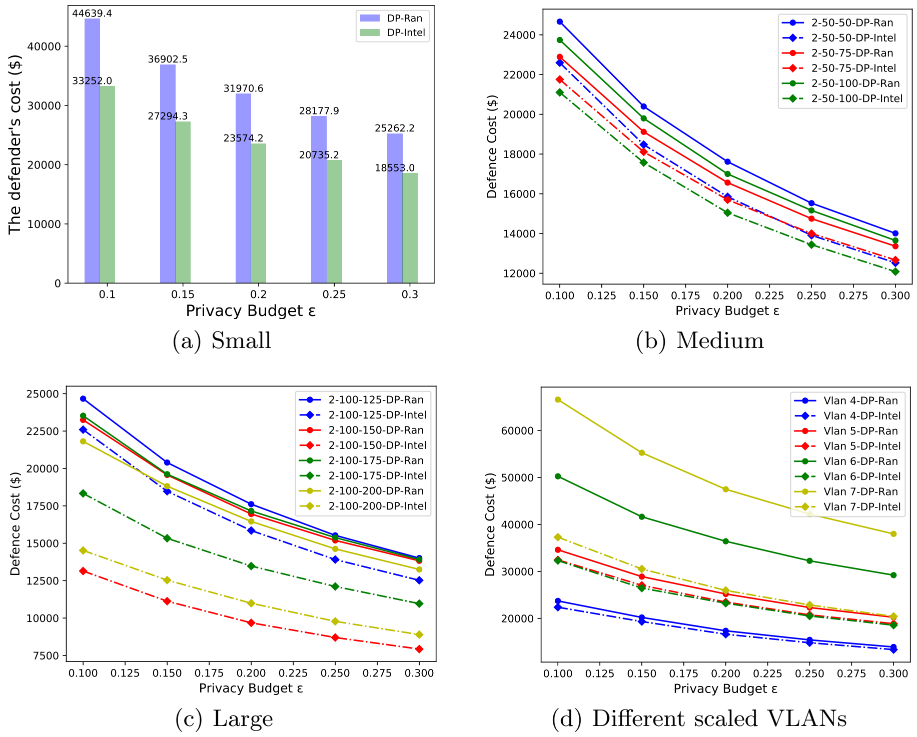
where $M1(k',k) * C1(k',k)$ represents the cost of moving devices from VLAN k to VLAN j, and $M2(d,k) * C2(d,k)$ is the cost of adding decoys d into VLAN k.

5 Evaluation

5.1 Simulation Setup

Fig. 3 shows an example smart healthcare system where IoT technologies are heavily adopted [32] [33]. It consists of 4 VLANs, with 4 IoMT devices in VLAN2 (e.g., Ultrasound, X-Ray, MRI and CT Scanner in the Radiology Department that send images to servers), 10 user machines in VLAN3 (i.e., staff office), and 5 IoMT devices in VLAN4 (e.g., Electroencephalography Monitor and Neuron Endoscopes sensor in the Neurology Department) and 2 servers in VLAN5 (i.e., server room). VLAN5 can be accessed by other three VLANs as IoT devices need to send patients' information to the servers for storage or processing. VLAN2 and VLAN4 are connected to VLAN3 for administration purposes.

Based on the above structure, we consider 3 different scaled IoT networks (i.e., small, medium and large) in the case study. For each network scale, we run 1000 rounds of simulations to evaluate the scalability of our method. In each round of simulation, the numbers of servers and user machines are fixed, while the IoT device number varies. For small-scale, we consider 2 servers, 10 user machines, 9 IoMT devices. The network shown in Fig. 3 is of small-scale. For medium-scale, we consider 2 servers, 50 user machines, and IoMT device number ranging from 50 to 100 per VLAN with an increment of 25 in each simulation. For large-scale, we consider 2 severs, 100 user machines, and the IoMT device number ranging from 125 to 200 per VLAN with an increment of 25 in each simulation. To evaluate the adaptability of the method, we also consider 2 servers, 100 user machines, 400 IoMT devices, and increase the number of VLANs from 4 to 7 with an increment of 1 in each simulation.



(a) Small

(b) Medium

(c) Large

(d) Different scaled VLANs

Fig. 4. Defender's cost under different privacy budgets on different scaled networks

5.2 Deception Cost

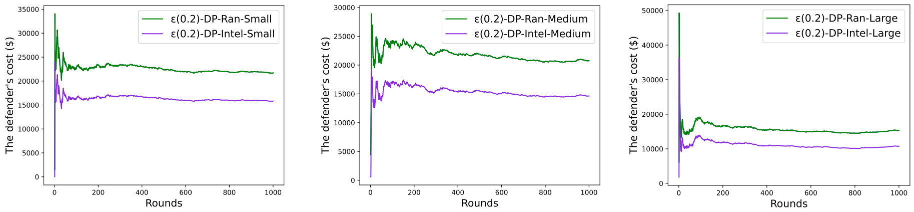
We consider the deception costs of two defence mechanisms. Each decoy is purchased individually with an annual license fee by type [34] [35]. The estimated decoy prices from different manufacturers are shown in Table 2. Assume each IoT device added to VLAN is of the same type and provided by the same manufacturer. Prices for the deception products are the same as the prices shown in Table 2. In the example network, if one device is moved from one VLAN to another, it is switched off first before the move. During the moving period, there is moving cost incurred as well. For simplicity, we treat the moving cost within one hour as the cost of one shuffling operation, i.e., 150$ per hour per device.

5.3 Strategies to Deploy Defence Mechanisms

Two strategies are considered when deploying defence mechanisms based on the obfuscation results: random selection of VLAN, decoy, or device (*DP-Ran*) and intelligent selection of VLAN, decoy, or device based on their criticality as mentioned in Sec. 4.3 (*DP-Intel*). We set the privacy budget ranging from 0.1

Table 2. Annual license fees for decoys [34] [35]

Products	Types	Annual License Fees ($)
Ultrasound	Emulation, Windows 8	200
X-Ray	Emulation, Windows 8	200
Magnetic Resonance Imaging (MRI)	Emulation, Windows 8	200
CT Scanner	Emulation, Windows 8	200
Electroencephalographs Monitor	Emulation, Windows 8	200
Nerve Monitor	Emulation, Windows 8	200
Neuron Endoscopes	Emulation, Windows 8	200
Spinal endoscopes	Emulation, Windows 8	200
Electromyography Monitor	Emulation, Windows 8	200
Computer	Emulation, Windows 8	300
Server	Full OS, Linux	1500



(a) Small (b) Medium (c) Large

Fig. 5. Defender's cost under privacy budget $\epsilon = 0.2$ on different scaled networks

to 0.3 with an increment of 0.05. Each simulation has 1000 rounds with random noise added.

Fig. 4(a) shows the results of the small-scale IoT network. It can be seen that the defence costs under both strategies decrease with the increasing privacy budget. The intelligent selection strategy has a better defence performance with less defence cost under the same privacy budget. According to [30], if the healthcare provider has a defence budget of $25,000.00, the intelligent strategy with a privacy budget of 0.15-0.2 is the best defence option. By adding a small noise to the number of devices per VLAN, we obfuscate the attacker, and the defence cost is well under the budget. That suggests a smaller privacy budget provides a higher protection level.

Fig. 4(b) and 4(c) show the defender's costs under different privacy budgets for medium-scale and large-scale networks. The defence cost decreases with the increasing privacy budget under *DP-Ran* and *DP-Intel*. This is because fewer devices to be moved or fewer decoys to be added into the networks for obfuscation. We can also see that the defence cost under *DP-Intel* is lower than that under *DP-Ran* with the same privacy budget. It verifies that our approach performs well on different scaled IoT networks and the defence cost decreases with the increasing privacy budget.

In Fig. 4(b), it can be seen that the *2-50-100-DP-Ran* has a higher defence cost than *2-50-75-DP-Ran* but a lower cost than *2-50-50-DP-Ran*. It is because the level of noise added to the number of devices per VLAN is random and the random strategy is used to select devices to be shuffled or decoys to be added. For the same reason, a similar case with *DP-Ran* can be seen in Fig. 4(c) where random noise has an impact on *DP-Intel* and causes varying defence costs.

Fig. 4(d) shows the defender's cost under different privacy budgets and a fixed number of IoT devices in varying VLANs (from 4 to 7). When the privacy budget increases, the defence cost decreases. The reason is because small noise is added when the privacy budget is larger and fewer assets are moved or added. On the other hand, with the increase of VLAN number, the defender's cost also increases under the same privacy budget and two selection strategies *DP-Ran* and *DP-Intel*.

Particularly, in Fig. 4, for either *DP-Ran* or *DP-Intel*, when the privacy budget increases, the difference between the defence costs under different sizes of networks decreases. For example, when $\epsilon = 0.3$, under each strategy, the difference between defence costs with different network sizes are minimal. It is because when the privacy budget ϵ is larger, the level of added noise is higher. It means the number of devices per VLAN can be very different with noise and without and the network's statistical information can not be obfuscated. The attacker can obtain the true operational information of IoT networks by analysing the collocated statistical information which is non-obfuscated, and then launch attack on real devices

It can be seen in Fig. 4(a), a good trade-off is achieved between the defence cost and the privacy budget when the privacy budget is 0.2. We also conduct experiments with 1000 rounds of simulation and privacy budget being 0.2 to observe the trend of defence cost. The results are shown in Fig. 5. It is obvious that the defence costs of *DP-Intel* are lower than those of *DP-Ran*, indicating our approach performs well on protecting the IoT networks under a defence budget.

6 Conclusion

In this work, we are motivated by the attack model where attackers may exploit the statistical information of IoT networks (e.g., the number of devices per VLAN), infer the operational information of VLANs and launch attacks. To address this problem, we utilise a differential privacy mechanism to obfuscate the network information by adding Laplace noise to change the number of devices per VLAN. We then use two defence technologies to achieve the obfuscation. We evaluate our approach by considering different scaled networks to find the trade-off between defence cost and privacy budget. The simulation results show that our approach with intelligent selection strategy has a better performance compared to the random selection strategy. In our work, as we focus more on applying differential privacy mechanism to obfuscate VLANs and protect the network, the greedy strategy used for implementing MTD and deception technologies is simplified. In our future work, we will consider more sophisticated and effective strategies to select VLANs, devices and decoys for deletion or addition, as well as different MTD techniques for obfuscation. Using different privacy-preserving solutions, such as (ϵ, δ)-DP and Gaussian DP, is also worth exploring.

References

1. P. Gokhale, O. Bhat, and S. Bhat, "Introduction to IoT," *International Advanced Research Journal in Science, Engineering and Technology*, vol. 5, no. 1, pp. 41–44, 2018.
2. L. Atzori, A. Iera, and G. Morabito, "The internet of things: A survey," *Computer Networks*, vol. 54, no. 15, pp. 2787–2805, 2010.
3. Help Net Security, "Threat highlight: Analysis of 5+ million unmanaged, iot, and iomt devices." `https://www.helpnetsecurity.com/2020/07/24/analysis-of-5-million-unmanaged-iot-and-iomt-devices/`, 2020.
4. THALES, "IoT security issues in 2022: A business perspective." `https://www.thalesgroup.com/en/markets/digital-identity-and-security/iot/magazine/internet-threats`, 2022.
5. M. Ge and D. S. Kim, "A framework for modeling and assessing security of the internet of things," in *2015 IEEE 21st International Conference on Parallel and Distributed Systems (ICPADS)*, pp. 776–781, IEEE, 2015.
6. A. K. Nayak, A. Reimers, N. Feamster, and R. Clark, "Resonance: Dynamic access control for enterprise networks," in *Proceedings of the 1st ACM workshop on Research on enterprise networking*, pp. 11–18, 2009.
7. M. H. Almeshekah and E. H. Spafford, "Planning and integrating deception into computer security defenses," in *Proceedings of the 2014 New Security Paradigms Workshop*, pp. 127–138, 2014.
8. S. Jajodia, A. K. Ghosh, V. Swarup, C. Wang, and X. S. Wang, *Moving target defense: creating asymmetric uncertainty for cyber threats*, vol. 54. Springer Science & Business Media, 2011.
9. M. Crouse, B. Prosser, and E. W. Fulp, "Probabilistic performance analysis of moving target and deception reconnaissance defenses," in *Proceedings of the Second ACM Workshop on Moving Target Defense*, pp. 21–29, 2015.
10. C. Wang and Z. Lu, "Cyber deception: Overview and the road ahead," *IEEE Security & Privacy*, vol. 16, no. 2, pp. 80–85, 2018.
11. J. H. Cho, D. P. Sharma, H. Alavizadeh, S. Yoon, N. Ben-Asher, T. J. Moore, D. S. Kim, H. Lim, and F. F. Nelson, "Toward proactive, adaptive defense: A survey on moving target defense," *IEEE Communications Surveys and Tutorials*, vol. 22, no. 1, pp. 709–745, 2020.
12. M. Ge, J. Cho, B. Ishfaq, and S. K. Dong, *Modeling and analysis of integrated proactive defence mechanisms for internet of things.* Modeling and Design of Secure Internet of Things, 2020.
13. M. Ge, J. B. Hong, S. E. Yusuf, and D. S. Kim, "Proactive defense mechanisms for the software-defined internet of things with non-patchable vulnerabilities," *Future Generation Computer Systems*, vol. 78, pp. 568–582, 2018.
14. M. Ge, J.-H. Cho, C. A. Kamhoua, and D. S. Kim, "Optimal deployments of defense mechanisms for the internet of things," in *2018 International Workshop on Secure Internet of Things (SIoT)*, pp. 8–17, IEEE, 2018.

15. M. Ge, J.-H. Cho, D. S. Kim, G. Dixit, and I.-R. Chen, "Proactive defense for internet-of-things: Integrating moving target defense with cyberdeception," *arXiv preprint arXiv:2005.04220*, 2020.
16. A. A. Mercado-Velázquez, P. J. Escamilla-Ambrosio, and F. Ortiz-Rodriguez, "A moving target defense strategy for internet of things cybersecurity," *IEEE Access*, vol. 9, pp. 118406–118418, 2021.
17. Z. Lu, C. Wang, and S. Zhao, "Cyber deception for computer and network security: Survey and challenges," *arXiv preprint arXiv:2007.14497*, 2020.
18. A. Juels and R. L. Rivest, "Honeywords: Making password-cracking detectable," in *Proceedings of the 2013 ACM SIGSAC*, pp. 145–160, 2013.
19. Q. D. La, T. Q. Quek, J. Lee, S. Jin, and H. Zhu, "Deceptive attack and defense game in honeypot-enabled networks for the internet of things," *IEEE Internet of Things Journal*, vol. 3, no. 6, pp. 1025–1035, 2016.
20. O. Tsemogne, Y. Hayel, C. Kamhoua, and G. Deugoué, "Game theoretic modeling of cyber deception against epidemic botnets in internet of things," *IEEE Internet of Things Journal*, 2021.
21. D. Ye, T. Zhu, S. Shen, and W. Zhou, "A differentially private game theoretic approach for deceiving cyber adversaries," *IEEE TIFS*, vol. 16, pp. 569–584, 2020.
22. ONF, "Openflow switch specification." `https://opennetworking.org/sdn-resources/openflow-switch-specification/`, 2017.
23. F. Cadini, E. Zio, and C.-A. Petrescu, "Using centrality measures to rank the importance of the components of a complex network infrastructure," in *International Workshop on Critical Information Infrastructures Security*, pp. 155–167, Springer, 2008.
24. S. Yoon, J.-H. Cho, D. S. Kim, T. J. Moore, F. Free-Nelson, and H. Lim, "Attack graph-based moving target defense in software-defined networks," *IEEE Trans on Network and Service Management*, vol. 17, no. 3, pp. 1653–1668, 2020.
25. D. P. Sharma, D. S. Kim, S. Yoon, H. Lim, J.-H. Cho, and T. J. Moore, "Frvm: Flexible random virtual ip multiplexing in software-defined networks," in *12th IEEE International Conference On Big Data Science And Engineering (TrustCom/BigDataSE)*, pp. 579–587, IEEE, 2018.
26. TrapX, "Security's deception grid. [online]." `https://www.scmagazine.com/trapx-security-deceptiongrid/article/681820`, 2017.
27. C. Dwork, "Differential privacy: A survey of results," in *International conference on theory and applications of models of computation*, pp. 1–19, Springer, 2008.
28. T. Zhu, P. Xiong, G. Li, W. Zhou, and S. Y. Philip, "Differentially private model publishing in cyber physical systems," *Future Generation Computer Systems*, vol. 108, pp. 1297–1306, 2020.
29. N. Li, M. Lyu, D. Su, and W. Yang, "Differential privacy: From theory to practice," *Synthesis Lectures on Information Security, Privacy, and Trust*, vol. 8, no. 4, pp. 1–138, 2016.
30. Attivo Networks, "Attivo botsink deception platform." `https://www.scmagazine.com/product-test/-/attivo-botsink-deception-platform`, 2016.
31. H. Alavizadeh, J. B. Hong, D. S. Kim, and J. Jang-Jaccard, "Evaluating the effectiveness of shuffle and redundancy mtd techniques in the cloud," *Computers & Security*, vol. 102, p. 102091, 2021.
32. A. James and M. B. Simon, "Medjack. 3 medical device hijack cyber attacks evolve," in *Proc. RSA Conf.(San Francisco, CA, USA)*, 2017.
33. S. Meggitt, "Medjack attacks: The scariest part of the hospital," 2018.
34. Medical Equipment Leasing Cost, "Medical equipment leasing cost." `https://costhack.com/medical-equipment-leasing-cost/`, 2020.

35. Computer, “How much does it cost to lease it equipment?.” `https://www.costowl.com/rental/equipment-leasing/equipment-leasing-computer-cost/`, 2022.

A Proof of Proposition 1

Proof. In the case of **Algorithm 1**, for neighbouring datasets N_k and N'_k, without loss of generality, let $\mathcal{A}$ be the step of Algorithm 1 that injects Laplace noise (i.e., Line 3 of Algorithm 1) and X be a random variable that follows $\mathrm{Lap}((\frac{|K|\cdot\mathit{\Delta}}{\epsilon}))$. For any output value z, we have:

$$\frac{\Pr[\mathcal{A}(N_k) = z]}{\Pr[\mathcal{A}(N'_k) = z]} \leq \exp^{\frac{\epsilon}{|K|}}. \tag{4}$$

For any count function f, $A_{f(N_k)} = f(N_k) + \mathrm{Lap}(\frac{|K|\cdot\mathit{\Delta}}{\epsilon})$, it is easy to conclude

$$\begin{aligned}
\frac{\Pr[\mathcal{A}_{f(N_k)} = z]}{\Pr[\mathcal{A}_{f(N'_k)} = z]} &= \frac{\Pr[f(N_k) + X = z]}{\Pr[f(N'_k) + X = z]} \\
&= \frac{\Pr[X = z - f(N_k)]}{\Pr[X = z - f(N'_k)]} \\
&= \frac{\frac{\epsilon}{2\cdot|K|\cdot\mathit{\Delta}} \cdot \exp^{\frac{-\epsilon\cdot|z-f(N_k)|}{|K|\cdot\mathit{\Delta}}}}{\frac{\epsilon}{2\cdot|K|\cdot\mathit{\Delta}} \cdot \exp^{\frac{-\epsilon\cdot|z-f(N'_k)|}{|K|\cdot\mathit{\Delta}}}} \\
&= \exp^{(\frac{-\epsilon\cdot|z-f(N_k)|}{|K|\cdot\mathit{\Delta}} - \frac{-\epsilon\cdot|z-f(N'_k)|}{|K|\cdot\mathit{\Delta}})} \\
&= \exp^{(\epsilon\cdot(\frac{|z-f(N'_k)|-|z-f(N_k)|}{|K|\cdot\mathit{\Delta}}))} \\
&\leq \exp^{(\frac{\epsilon\cdot|f(N'_k)-f(N_k)|}{|K|\cdot\mathit{\Delta}})}.
\end{aligned} \tag{5}$$

Since the sensitivity $\mathit{\Delta}$ is 1 as mentioned before, and by definition of sensitivity, $\mathit{\Delta} = \max_{N_k, N'_k} \|f(N'_k) - f(N_k)\|_1$. Hence, Eq. (5) becomes

$$\begin{aligned}
\frac{\Pr[\mathcal{A}_{f(N_k)} = z]}{\Pr[\mathcal{A}_{f(N'_k)} = z]} &= \frac{\Pr[f(N_k) + X = z]}{\Pr[f(N'_k) + X = z]} \\
&\leq \exp^{(\frac{\epsilon\cdot|f(N'_k)-f(N_k)|}{|K|\cdot\mathit{\Delta}})} \\
&\leq \exp^{(\frac{\epsilon}{|K|})}.
\end{aligned} \tag{6}$$

Thus, each step of Algorithm 1 satisfies $\frac{\epsilon}{|K|}$-DP. As there are $|K|$ steps in Algorithm 1, based on Theorem 1, Algorithm 1 satisfies $(\sum_{i=1}^{|K|} \frac{\epsilon}{|K|})$-$DP$. Therefore, Algorithm 1 satisfies ϵ-DP.

Without loss of generality, denote Algorithm 1 as $\mathcal{A}_1$ and Algorithm 2 as $\mathcal{A}_2$. In the case of **Algorithm 2**, for neighbouring N_k and N'_k, let z be the output value of algorithm $\mathcal{A}_1$ and O be the set of output value of algorithm $\mathcal{A}_2$. According to the discussion above, we have proved $\mathcal{A}_1$ satisfies ϵ-DP, so we have

$$\frac{\Pr[\mathcal{A}_1(N_k) = z]}{\Pr[\mathcal{A}_1(N'_k) = z]} \leq \exp^{\epsilon}. \tag{7}$$

For any $o \in O$, we have

$$\begin{aligned}
\Pr[\mathcal{A}_2\left(\mathcal{A}_1(N_k)\right) = o] &= \sum_{o \in O} \Pr[\mathcal{A}_1(N_k) = z]\Pr[\mathcal{A}_2(z) = o] \\
&\leq \sum_{o \in O} \exp^{\epsilon} \Pr[\mathcal{A}_1(N'_k) = z]\Pr[\mathcal{A}_2(z) = o] \\
&= \sum_{o \in O} \exp^{\epsilon} \Pr[\mathcal{A}_2\left(\mathcal{A}_1(N'_k)\right) = o]. \qquad (8)
\end{aligned}$$

Hence, according to Eq. (8), we have

$$\frac{\Pr[\mathcal{A}_2\left(\mathcal{A}_1(N_k)\right) = o]}{\Pr[\mathcal{A}_2\left(\mathcal{A}_1(N'_k)\right) = o]} \leq \exp^{\epsilon}, \qquad (9)$$

Therefore, Algorithm 2 also satisfies ϵ-DP based on the Post-processing Theorem 2. □