Securing Cooperative Spectrum Sensing against Rational SSDF Attack in Cognitive Radio Networks

Jingyu Feng[1, 2], Yuqing Zhang[2], Guangyue Lu[1] and Liang Zhang[1]

[1] Department of Communication Engineering, Xi'an University of Post & Telecommunication
Xi'an 710121, China
[e-mail: fjy.fengjingyu@gmail.com]
[2] National Computer Network Intrusion Protection Center, University of Chinese Academy of Sciences
Beijing 100190, China
[e-mail: zhangyq@ucas.org.cn]
*Corresponding author: Jingyu Feng

Received August 6, 2013; revised October 20, 2013; revised December 11, 2013; accepted January 4, 2014; published January 29, 2014

Abstract

Cooperative spectrum sensing (CSS) is considered as a powerful approach to improve the utilization of scarce radio spectrum resources. However, most of CSS schemes assume all secondary users (SU) are honest, and thus offering opportunities for malicious SUs to launch the spectrum sensing data falsification attack (SSDF attack). To combat such misbehaved behaviors, recent efforts have been made to trust schemes. In this paper, we argue that powering CSS with traditional trust schemes is not enough. The rational SSDF attack is found in this paper. Unlike the simple SSDF attack, rational SSDF attackers send out false sensing data on a small number of interested primary users (PUs) rather than all PUs. In this case, rational SSDF attackers can keep up high trustworthiness, resulting in difficultly detecting malicious SUs in the traditional trust schemes. Meanwhile, a defense scheme using a novel trust approach is proposed to counter rational SSDF attack. Simulation results show that this scheme can successfully reduce the power of rational SSDF, and thus ensure the performance of CSS.

Keywords: Cooperative spectrum sensing; cognitive radio networks; trust; SSDF attack.

A preliminary version of this paper appeared in IEEE TrustCom 2013, July 16-18, Melbourne, Australia. This version includes a concrete analysis of rational SSDF attack and the functional modules of its defense scheme. This research was supported in part by the National Science Foundation of China (61271276, 61272481, 61301091), the Key Program for International S&T Cooperation Projects of Shaanxi Province(No: 2013KW01-03), the China Postdoctoral Science Foundation(2013M541013), the Youth Natural Science Research Program of XUPT (ZL2012-06).

http://dx.doi.org/10.3837/tiis.2014.01.001

1. Introduction

With the rapid development of wireless communication technology and the huge demand of the capacity for wireless applications, spectrum resources have become increasingly scarce. However, a large portion of the assigned spectrum is not utilized efficiently. According to the Federal Communications Commission (FCC) 오류! 참조 원본을 찾을 수 없습니다., temporal and geographical variations in the utilization of the assigned spectrum range from 15% to 85%. To solve the contradiction between the spectrum scarcity and low spectrum utilization, cognitive radio networks (CRNs) [2-3] have been proposed to make effective use of the frequency spectrum by opportunistically using the spectrum of the licensed users. The licensed users are called the primary users (PUs) and the unlicensed users of the CRNs are the secondary users (SUs).

Cooperative spectrum sensing (CSS) is one of the key technologies in the realization of CRNs, since it enables SUs to find the unused spectrum bands without causing harmful interference to PUs. In a CSS architecture, all the participating SUs forward their observations regarding the presence or absence of a PU to a fusion center (FC), which makes the final decision about whether the PU is transmitting or not. By cooperation, SUs can share their sensing data to make a combined decision with increased accuracy as comparing with the individual decisions [4].

On the other hand, the cognitive radio paradigm imposes human-like characteristics (e.g., learning, adaptation and cooperation) in wireless networks [5]. A CSS action is often established randomly among SUs that are unrelated and unknown to each other. This offers opportunities to malicious SUs who launch SSDF attack by sending false spectrum sensing data [6], causing the FC to make a wrong spectrum sensing decision. How to efficiently and effectively counter SSDF attack has become a very challenging issue to achieve better performance of CSS.

Nowadays, trust is used as a popular and yet effective approach to encourage real sensing data sharing among SUs. Various trust schemes have been proposed [7-10]. They estimate whether an SU is trustworthy or not by it past behaviors concerning all PUs as a whole, and give low weights to the sensing data from less trustworthy SUs when generating the final decision.

These trust schemes can make an adversary's SSDF attack more difficult to succeed. But, this successful foundation is built on the fact that SSDF attackers always report false sensing data on all PUs. They cannot prevent the attackers from abusing the trust evaluation. That is, the attackers can strategically report false sensing data on a small number of PUs they are interested in, but provide real sensing data on most PUs for the purpose of boosting their trustworthiness. Such attack is found in this paper named as rational SSDF (hereinafter "RSSDF").

To defend against RSSDF attack in the CSS environment, we propose a trust scheme from a novel angle, called SensingGuard. Unlike the traditional trust schemes, the trustworthiness of each SU is evaluated by his past behaviors concerning different PUs respectively. That is, the calculation of an SU's trust value is bound to each PU. The higher trust value of an SU just shows it is trustworthy on a PU not all PUs. Only when the set of trust values concerning all PUs are higher will the SU be recognized as trustworthy. Meanwhile, considering that the

individual sensing report of each SU is a binary variable, we can evaluate the trustworthiness of each SU through analyzing such binary variables to detect RSSDF attackers, resulting in less mathematical analysis and computation.

The organization of this paper is as follows: In section2, preliminaries related on CSS and trust Schemes are described. In section 3, we analyze RSSDF attack and constructs the SensingGuard scheme to defend against it. Simulation analysis of SensingGuard is given in section 4. Finally, we conclude the paper in section 5.

2. Preliminaries

In this section, we first describe background of CSS. We then review the related work of trust schemes, in which a basic trust scheme of CSS is concluded.

2.1 Cooperative Spectrum Sensing (CSS)

Carrying out reliable spectrum sensing is a challenging task for an SU. The spectrum sensing can basically be classified as individual spectrum sensing and cooperative spectrum sensing. In the case of deep shadowing and multipath fading, it is very difficult for an SU to distinguish a white space from a deep shadowing effect. Therefore, an individual spectrum sensing system may not work well in this case, and a cooperative scheme can solve the problem effectively by sharing the spectrum sensing data among SUs.

The CSS process can be modeled as a parallel fusion network [11]. As shown in **Fig. 1**, a central identity called fusion center (FC) controls the process of CSS: individual sensing, data reporting and decision making [12]. First, each SU exploits the energy detection to sense the signal of a PU via the sensing link. Second, all SUs report their sensing data to the FC via the reporting link. Then the FC combines the received local sensing information and determines the presence of PU. The final decision can be made according to three typical CSS fusion schemes, such as the *AND*, *OR* and *Majority* rule [13].

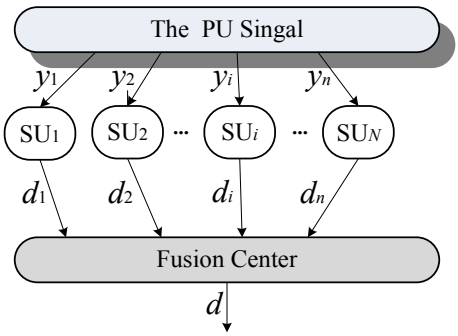

Fig. 1. Modeling CSS into a parallel fusion network.

Typically, individual sensing for primary signal energy detection can be formulated as a binary hypothesis problem as follows [14]:

$$y(t)=\begin{cases} n(t), & H_0 \\ h(t)\cdot s(t)+n(t), & H_1 \end{cases} \tag{1}$$

where y(t) represents the detected signal at each SU, $s(t)$ is the transmitted PU signal, $h(t)$ is the channel gain of the sensing channel, $n(t)$ is the zero-mean additive white Gaussian noise

(AWGN), and t is the sample index. H_0 and H_1 denote the hypothesis of the absence and the presence of the PU signal, respectively.

For simplicity of derivation, $s(t)$ is assumed to be independent [15]. The detected signal y(t) is independent since $n(t)$ is also independent. Based on this independence, the decision metric for the energy detection can be written as:

$$M = \sum_{t=0}^{D} \left| y(t) \right|^2 \tag{2}$$

where D is the size of the observation vector. The decision on the occupancy of a PU singal can be obtained by comparing the decision metric M against a energy decision threshold λ_E. The existence of PU would be declared when $M>\lambda_E$. Otherwise, there is no PU singal when $M<\lambda_E$. However, if the sensing channels are facing deep fading or shadowing, individuals will fail to detect the presence of PU. To improve the performance of spectrum sensing, SUs spread out in the spatial distance and observe a PU signal via multi-path sensing links between the PU transmitter and each SU, then propagate their observations to the FC synchronously.

After the individual sensing, the individual sensing report of each SU is determined. d_i indicates the individual sensing report of SU_i, which is usually expressed as a binary variable:

$$d_i = \begin{cases} 0, & H_0 \\ 1, & H_1 \end{cases}$$

where "0" and "1" denote the hypothesis of the absence and the presence of the PU singal, respectively. The spectrum sensing problem therefore can be regarded as a binary. Correspondingly, the final decision d is also binary under the *AND*, *OR* and *Majority* fusion rule. In the *AND* rule, the FC determines d=1 if all individual sensing d_i=1. The *OR* rule refers to d=1 if an individual sensing d_i=1. The *Majority* rule requires at least a half of SUs to report ''1''. The *OR* rule works best when the number of SUs is large, whereas the *AND* rule works well when the number of cooperating users is small, and the *Majority* rule can be obtained from the k out of N rule under the condition when $k \geq N/2$ [12].

For the evaluation of the detection performance, the probabilities of individual detection P_d (d_i=1 when the PU singal is using) and false alarm P_f (d_i=1 when the PU singal is free) are defined.

In cooperative sensing, the probabilities of detection and false alarms for evaluating the performance of cooperative decisions are denoted by Q_d and Q_f, respectively, which can be written as follows[16]:

$$Q_d = Prob\left(H_1 \middle| H_1\right) = \sum_{l=k}^{N} \binom{N}{l} P_d^l \left(1 - P_d\right)^{N-l} \tag{3}$$

$$Q_f = Prob\left(H_1 \middle| H_0\right) = \sum_{l=k}^{N} \binom{N}{l} P_f^l \left(1 - P_f\right)^{N-l} \tag{4}$$

It can be seen that the *OR* rule corresponds to the case of k=1, the *AND* rule corresponds to the case of k=N and the *Majority* rule corresponds to the case of $k \geq N/2$.

2.2 Trust Schemes

Trust schemes are having increasing influence on many application scenarios, including e-commerce [17], P2P file-sharing [18], ad hoc routing [19], social networks [20], and so on.

Trust schemes also play significant roles in CSS area，such as 1) assisting FC in accurate decision making, 2) encouraging trustworthy behavior, and 3) deterring participation by malicious SUs. Representative schemes are as follows. In [7], the authors proposed a novel trust-aware hybrid spectrum sensing scheme, in which the Beta Reputation System is applied to construct trust scheme. Zeng et al proposed a reputation-based cooperative spectrum sensing scheme in [8], and categorize the trustworthiness of each SU into three states. In [9], the authors proposed a novel trust-aware resource allocation scheme in a centralized cognitive radio network with a system-level trust scheme to detect misbehaving SUs and filter out the malicious attack for CSS, in which trustworthiness is used as social capital to gain system resources. In [10], the authors measured the trustworthiness of SUs in CSS during the cognition cycle, and incorporate it into the sensing data fusion to reduce the effect of malicious SUs on the final decision making.

The common property of these traditional trust schemes is that the trustworthiness of an SU is evaluated by it past behaviors concerning all PUs as a whole. Such property gives a chance to RSSDF attack. The attackers report false sensing data on a small number of PUs, but real sensing data on most PUs. In this case, the attackers can hold high trustworthiness in these trust schemes.

With the common property, we present a basic trust scheme to abstract the traditional trust schemes. In the subsequent sections, the basic trust scheme will be used to compare simple SSDF attack with RSSDF attack, and demonstrate RSSDF attack as well as experimental results.

It is important to note that the individual sensing report of each SU is a binary variable. In a CSS action, each SU plays two types of sensing behaviors: real or false. Based on this, the evaluation of trust value depends on the two factors: the number of real sensing (r) and the number of false sensing (f), and thus the beta function method is well suitable for CSS trust evaluation. The beta function denoted by $Beta(r, f)$ takes binary ratings as input, which can be expressed using the gamma function [21]:

$$Beta(r,s)=\frac{\Gamma(r+f)}{\Gamma(r)\Gamma(f)}\theta^{r-1}(1-\theta)^{s-1},\ 0\le\theta\le 1 \tag{5}$$

where θ is the probability of sensing behaviors.

Take example for the i-th SU (SU_i), its trust value denoted by t_i can be evaluated with beta function as: $t_i = Beta(r_i+1, f_i+1)$. Without any prior observations, $r_i=f_i=0$ and hence, $t_i=$ Beta(1,1). Consider the case $\Gamma(x)=(x-1)!$ when x is an integer [22]. It can be deduced that the expectation value of the beta function is given by: $E[Beta(r, f)]=r/(r+f)$. Thus, t_i can be further described is as follows:

$$t_i=\frac{r_i+1}{r_i+f_i+2} \tag{6}$$

In the basic trust scheme, t_i is a real number ranging from 0 (complete distrust) to 1 (complete trust). For $r_i=f_i=0$, SU_i is recognized as a newcomer and t_i is initialize as 0.5.

Afterwards, the more SU_i often provides honest sensing data, the higher trust value he will get, and vice verse.

3. RSSDF Attack and Defense Scheme

In this section, we first describe RSSDF attack, and then design the SensingGuard scheme to defend against the attack. Finally, a case study is created to further illustrate the design idea of SensingGuard.

3.1 RSSDF Attack Overview

Since the individual sensing report is usually regarded as a binary variable, it is very easy for malicious SUs to take advantage of CSS and launch SSDF attack by faking individual spectrum sensing data, resulting in a wrong final sensing decision.

Generally speaking, the basic goal of SSDF attackers against CSS is to illegally occupy or disturb the PU spectrum bands. Such attackers can be classified according to their attack goal [23].

- **Always-using**: The attackers declare that the primary user is active, although there is no PU spectrum bands. In this case other SUs make a wrong decision that PUs are present and will not use the spectrum. The intention of the attackers is to gain exclusive access to the target spectrum.
- **Always-free**: The attackers report an absent primary signal, although there are PUs using their spectrum bands. In this case other SUs make a wrong decision that the PU spectrum bands are free and will use the spectrum. The intention of the attackers is to give interference to PUs.

These two kinds of attackers are the most dangerous. Fortunately, they can be easily detected by current trust schemes if the attacks always send false sensing data to the FC in order to alter the final decision. This is because the attackers will obtain a lower trust value when they always report false sensing data. This SSDF strategy is known as the simple SSDF (hereinafter " SSSDF ") in this paper.

To avoid the detection of trust schemes, the attackers have to adopt new SSDF strategies. This question leads to the discovery of RSSDF. Such attack is more difficult to counter in traditional trust schemes, because attackers can exhibit rational behavior that allows them to partially hide through reporting real sensing data sometimes.

Unlike the SSSDF attackers, RSSDF attackers are extremely sensitive to their trust value. Assuming SU_k is a RSSDF attacker, he launches RSSDF attack under the constraint

$$\varepsilon \le t_k < \varepsilon + 0.1$$

ε is the threshold of trustworthiness. As each $t_k \in [0,1]$, ε is usually set to a moderate value, such as 0.5. For $t_k \ge \varepsilon$, SU_k will be not identified by trust schemes since he is marked as honest. This inspires RSSDF attackers to find an attack procedure with trust-boost. That is, SU_k should begin to boost its trust value when $\varepsilon \le t_k < \varepsilon + 0.1$. It is late for boosting trust when $t_k < \varepsilon$. In this case, SU_k is marked as malicious by trust schemes and anyone won't trust him again. Actually 0.1 is the yellow warning line of RSSDF attackers. It is not necessary to set the yellow warning line in a larger value. Otherwise, RSSDF attackers will be busy boosting trust even if a small reduction in their trust value appears. Under the above constraint, the RSSDF attack procedure is conducted in a round mode.

Step 1. Launching RSSDF. SU_k reports false sensing data to illegally occupy or disturb the interested PU spectrum bands via disguising an honest one.
Step 2. Evaluating t_k. SU_k also stores the two factors(r_k, f_k) in his machine and evaluate his trust value after an attack.
Step 3. Checking $\varepsilon \le t_k < \varepsilon + 0.1$. It means his trust value will be lower than the threshold immediately. Yes, continue Step 4. No, go to Step1.
Step 4. Boosting $t_k > \varepsilon$. SU_k searches for the PU spectrum bands in which he is not interesed and reports real sensing data until his trust value is far more than the threshold.

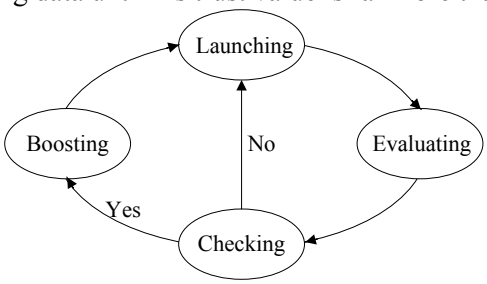

Fig. 2. A round of RSSDF attack procedure

The RSSDF attack is usually launched in the case where several PUs exist or multiple spectrum channels have been assigned to a cognitive radio network. Currently, several small-scale PUs such as wireless microphones can be found in IEEE 802.22 or radios can be found in emergency and military networks [12]. Akyildiz et al [14] also introduced the concept of the primary network which is consisted of PUs and controlled through a primary base station. Specially in the future as the spectrum resources become more and more scarce, multiple spectrum channels would be assigned to a wireless region rather than some specific users. In this environment, each mobile device would become a secondary user to sense or even access these channels opportunistically.

3.2 Design of SensingGuard

We have known that the RSSDF attackers fake sensing data on their interested PUs and report real sensing data to boost their trust value. In the SensingGuard scheme, our design idea is that the trustworthiness of each SU should be evaluated by his past behaviors concerning different PUs respectively. As shown in **Fig. 3**, this scheme is built with three functional modules: Data Management, Trustworthiness Evaluation and Attackers Detection module.

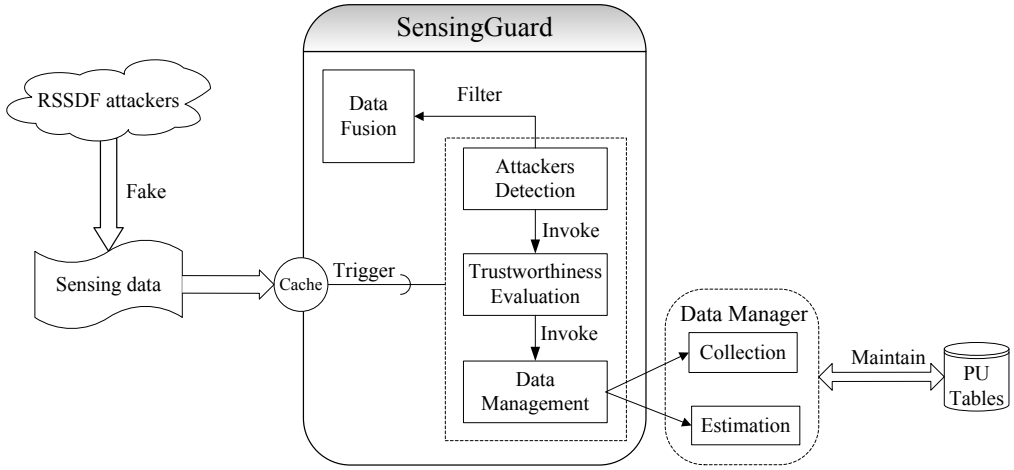

Fig. 3. Functional modules in the SensingGuard scheme

Data Management. In **Fig. 3**, this module has a data manager that is responsible for performing two main tasks. It first collects the sensing data during each CSS action. This task can be built on the two-steps process of CSS [12]: data reporting and decision making. For data reporting, the FC is required to store the sensing data to a small database rather than discarding them. Considering the demand of defending against RSSDF attack, the sensing data should be not saved as a whole on all PUs, but based on the viewpoint of differentiating PUs. Therefore, the small database is made up of multiple PU tables. That is, each PU is assigned to a table that saves the sensing data previously provided by all SUs on the PU. Take PU_j as an example, the sensing data provided by all SUs and PU_j's practical spectrum status (*pss*) are stored in the PU_j table, as shown in **Table 1**.

Table 1. Description of the PU_j tableTable Styles

Times	**SU_1**	**SU_2**	**…**	**SU_i**	**…**	**SU_n**	**PU_j**
1	d_1	d_2	...	d_i	...	d_n	pss_j
2	d_1	d_2	...	d_i	...	d_n	pss_j
…	...	...	...	...	...	...	...
N	d_1	d_2	...	d_i	...	d_n	pss_j

In the PU tables, *pss* is stored during the process of decision making. That value is basically the final sensing decision of the FC. But, we could not ignore a fact that CSS enables SUs to find the unused spectrum bands without causing harmful interference to PUs. However, this scenario is usually violated by malicious SUs who inject false reports into the sensing data. In this case, there is a real-time cooperation between PUs. To avoid the harmful interference, this real-time cooperation is basically the communication from PUs to the FC. That is, PUs should have the right to send a complaint to the FC while causing harmful interference from SUs. Then the practical spectrum status (*pss*) is updated to a correct value when a complaint arrives at the FC. To mitigate the harmful interference to PUs, the FC is also required to identify the attacker who fake sensing data and filter out them in the future data fusion.

The task of data collection would not add any burden to the FC by improving the existing process of CSS, as it only requires that the FC stores the sensing data during each CSS action rather than discarding them when the presence of PU is determined. By doing so, another benefit is that this task is independent of users mobility. This is because all SUs just send sensing data in the process of data reporting. Afterwards, they are not required to report data again, and then they can move freely.

To obtain the factors for evaluating trustworthiness, the second task of the data manager is to estimate the number of real sensing and the number of false sensing from the sensing data stored in the PU tables. Such estimation lies on the PU's practical spectrum status. Let (r_{ij}, f_{ij}) denote the number of real sensing and the number of false sensing performed by SU_i on PU_j, respectively. The two factors (r_{ij}, f_{ij}) can be estimated by Procedure 1.

Procedure 1 Estimating (r_{ij}, f_{ij})

Input: PU_j table
Output: r_{ij}, f_{ij}
1: Initialize $r_{ij}=f_{ij}=0$
2: **for** each sensing time **do**
3: **if** (d_i==pss_j) **then**
4: r_{ij}++
5: **else**

6: f_{ij}++
7: **end if**
8: **end for**

Trustworthiness Evaluation. As we know, the RSSDF attackers can keep up high trustworthiness in the traditional trust schemes, since each SU is evaluated by considering all PUs as a whole. Unlike the traditional trust schemes, the design idea of differentiating PUs is used in our SensingGuard scheme to evaluate the trustworthiness of each SU related on different PUs. This SU-PU relationship graph of SensingGuard compared with the traditional trust schemes is shown in **Fig. 4**.

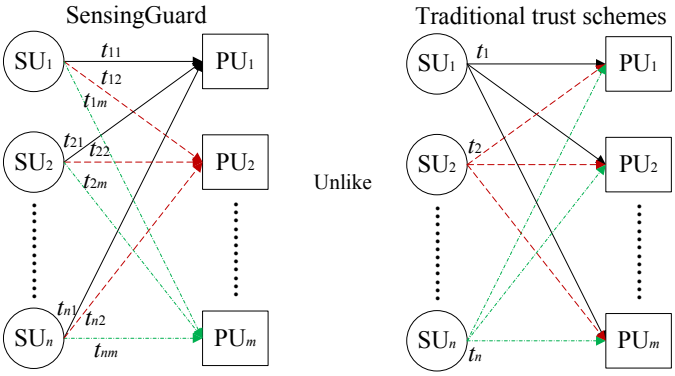

Fig. 4. SU-PU relationship graph of SensingGuard compared with the traditional trust schemes

The advantage of SensingGuard over the traditional schemes is to prevent the RSSDF attackers from keeping up high trustworthiness. In the SensingGuard scheme, the result of trustworthiness evaluation is not a signal value, but a set of trust values. Take SU_i as an example again, the trust value of SU_i related on PU_j (t_{ij}) can be calculated as:

$$t_{ij} = \frac{r_{ij} + 1}{r_{ij} + f_{ij} + 2} \tag{7}$$

Analogously, we can evaluate the trust values of other SUs who have reported sensing data on PU_j, and thus generating a trust vector related on PU_j, which is expressed as

$$T_j = \left[t_{1j}, \cdots \quad \cdots \right.$$

For all PUs, their trust vectors compose a matrix $T_{m\times n}$, where m is the number of PUs in a cognitive radio network and n is the number of SUs.

$$T_{m\times n} = \begin{bmatrix} t_{11} & \cdots \\ \vdots & \ddots & \vdots \\ t_{m1} & \cdots \end{bmatrix}$$

Attackers Detection. The key to ensuring the performance of CSS is to filter the false reports from SUs who have suspicious behaviors before the data fusion. In this module, the two factors (τ, ψ) are employed to detect the RSSDF attackers through analyzing the matrix $T_{m\times n}$.

Table 2. Description of two factors

Factors	SU_1	SU_2	…	SU_i	…	SU_n
τ	τ_1	τ_2	…	τ_i	…	τ_n
ψ	ψ_1	ψ_2	…	ψ_i	…	ψ_n

As shown in **Table 2**, τ denotes the mean value of an SU's trust values and ψ is the amount of low trust values on some particular PUs. For example, SU_i sends out real sensing data on three PUs and sends out false sensing data on two PUs. By Eq.(6) and the matrix $T_{m\times n}$, τ_i may be great than the threshold ε , but we can identify the RSSDF attacker by ψ_i=2.

Specially, we find that the SensingGuard scheme can also counter the SSSDF attack. As we know, an SSSDF attacker can be detected under the case $t_i<\varepsilon$, in that he always fakes sensing data on all PUs. In the SensingGuard scheme, such attacker will get a low value in τ_i when $\psi_i=m$.

Proof: $$\sum_{1\le j\le m} t_{ij} < \sum_{1\le j\le m} \varepsilon \Rightarrow \sum_{1\le j\le m} t_{ij} < m*\varepsilon \Rightarrow \frac{1}{m}\sum_{1\le j\le m} t_{ij} < \varepsilon \Rightarrow \tau_i < \varepsilon$$

In other words, the case $\tau_i<\varepsilon$ appears when an SSSDF attacker fakes sensing data on all PUs. In section 4, the first simulation is also given to validate this feature.

To detect the two types of attackers simultaneously, we apply Procedure 2 to separate the set of SUs (Ψ) into three clusters: the cluster Ψ_s which consists of SSSDF attackers, the cluster Ψ_r which consists of RSSDF attackers and the cluster Ψ_h which consists of honest SUs.

Procedure 2 Attackers detection
Input: Ψ
Output: Ψ_s, Ψ_r, Ψ_h
1: Initialize $\Psi_s=\Psi_r=\Psi_h=\varnothing$;
2: **for** each $SU_i\in\Psi$ **do**
3: *Calculate* τ_i *and* ψ_i
4: **if** ($\tau_i<\varepsilon\&\&\psi_i=m$) **then**
5: SU_i is an SSSDF attacker
6: $\Psi_s\leftarrow SU_i$
7: **elseif** ($\tau_i>\varepsilon\&\&1\le\psi_i<m$) **then**
8: SU_i is a RSSDF attacker
9: $\Psi_r\leftarrow SU_i$
10: **elseif** ($\tau_i>\varepsilon\&\&\psi_i=0$) **then**
11: SU_i is an honest SUs
12: $\Psi_h\leftarrow SU_i$
13: **end if**
14: **end for**

For SSSDF attackers, their reports should be screened out before the data fusion since they always fake sensing data on all PUs. For RSSDF attackers, their reports could be screened out selectively in either case. When the number of cooperating SUs to a CSS exchange is large, such sufficient sensing data can be ensured to make a final decision on a PU spectrum band. So, there's no need to fuse the reports from RSSDF attackers if they are the minority in the cooperating SUs. But when the number of cooperating SUs is small, every report is important for the data fusion. It is necessary to check whether RSSDF attackers faked sensing data on the PU before. If not, their reports can be used in the data fusion.

3.3 Case Study

To further illustrate the SensingGuard scheme, we also create a case study with 5 cooperating SUs (SU_1, SU_2, SU_3, SU_4 and SU_5) and 3 PUs (PU_1, PU_2 and PU_3).

Firstly, the PU tables with respect to 3 PUs are assumed as follows:

Table 3. Description of the PU1 table

Times	SU_1	SU_2	SU_3	SU_4	SU_5	PU_1
1	1	0	1	1	1	1
2	-	1	-	0	1	1
3	1	0	1	-	1	1
4	1	1	0	0	0	0
5	0	0	1	-	1	1
6	1	-	0	-	-	0
7	1	0	-	1	0	1
8	0	0	1	1	1	1

Table 4. Description of the PU2 table

Times	SU_1	SU_2	SU_3	SU_4	SU_5	PU_2
1	0	1	1	1	-	1
2	-	1	0	-	0	0
3	1	-	0	0	-	0
4	-	0	-	1	1	1
5	1	1	0	0	0	0
6	1	-	0	1	0	0
7	0	0	0	-	-	1
8	1	1	1	1	0	1

Table 5. Description of the PU3 table

Times	SU_1	SU_2	SU_3	SU_4	SU_5	PU_3
1	0	1	1	0	1	1
2	1	0	-	-	-	0
3	0	0	0	0	1	1
4	1	1	0	0	1	1
5	0	-	-	0	-	1
6	0	0	0	1	1	1
7	-	1	-	-	0	0
8	1	0	0	1	1	1

where “0”and “1” denote the absence and the presence of the PU spectrum band, respectively. “-”denotes an SU sends out nothing.

By the Trustworthiness Evaluation module, the matrix $T_{3\times5}$ is calculated as:

$$T_{3\times5} = \begin{bmatrix} 0.444 & 0.222 & 0.875 & 0.714 & 0.667 \\ 0.25 & 0.375 & 0.778 & 0.75 & 0.714 \\ 0.333 & 0.444 & 0.256 & 0.375 & 0.875 \end{bmatrix}$$

Finally, we apply Procedure 2 built on the matrix $T_{3\times5}$ to detect the SSSDF and RSSDF attackers.

Table 6. Description of two factors with respect to indentifying SSDF attackers from 5 cooperating SUs

Factors	SU_1	SU_2	SU_3	SU_4	SU_5
τ	0.342	0.347	0.551	0.613	0.752
r	3	3	1	1	0

As shown in **Table 6**, we can see that:

- For τ_1=0.342&&ψ_1=3 and τ_2=0.347&&ψ_2=3, SU_1 and SU_2 are identified as the SSSDF attackers.
- For τ_3=0.551&&ψ_3=1 and τ_4=0.613&&ψ_4=1, SU_3 and SU_4 are identified as the RSSDF attackers, who send out real sensing data on PU_1, PU_2 and sends out false sensing data on PU_3.
- For τ_5=0.752&&ψ_5=0, SU_5 is identified as an honest SU who sends out real sensing data on PU_1, PU_2 and PU_3.

4. Simulation Analysis

We demonstrate the effectiveness of the SensingGuard scheme by using the Monte Carlo simulation.

4.1 Simulation Setup

We implemented the simulations based on the energy detection, in which the primary signal is a baseband QPSK modulated signal under the AWGN (additive white Gaussian noise) environment. The general simulation setup is shown in **Table 7.**

Table 7. Description of simulation elements

	Description	Default
Environment Setting	Number of PUs in the network	5
	Number of SUs in the network	30
	Percentage of attackers	40%
	Sampling frequency	1KHz
	SNR	-8dB
	Time-bandwidth product	50
	Trustworthiness Threshold	0.5

In the simulations, the SUs are split into three types: SSSDF attackers, RSSDF attackers SUs and honest SUs. The behavior pattern for SSSDF attackers is to fake sensing data on all PUs. RSSDF attackers always fake sensing data on two PUs but send out real sensing data on three PUs. Considering the the case of deep shadowing and multipath fading, the behavior pattern for honest SUs is modeled to provide real sensing data at the probability of 0.8.

The simulations initiate by cycle-based fashion. At each cycle, all SUs are selected to perform CSS actions with each other randomly. After a few cycles, a trusted network topology is gradually formed by SensingGurad. The FC then uses the scheme to perform CSS actions at each cycle, and update the trust values on the corresponding SUs.

4.2 Simulation Results

We performed four simulations to validate the SensingGuard scheme in terms of comparing SensingGuard with the basic trust scheme evaluating the trustworthiness of each SU by

considering all PUs on behalf of Traditional Trust. By doing so, we can test that SensingGuard evaluating the trustworthiness of each SU by considering different PUs is more effective than Traditional Trust to suppress RSSDF attack.

The key to defending against SSDF attack is to detect the attackers and filter out them in the process of data fusion. The first simulation validates the effectiveness of SensingGuard in terms of its detecting the SSSDF attackers and RSSDF attackers. As shown in **Fig. 5**, the curve of SensingGuard and Traditional Trust are similar to detecting the SSSDF attackers after 9 cycles. The reason is that the SSSDF attackers can get a low value in τ if they fake sensing data on all PUs. In this case, the trustworthiness of the attackers in both SensingGuard and Traditional Trust are similar to SSSDF attack. Due to evaluating the trustworthiness of each SU by considering different PUs, our SensingGuard scheme is better than Traditional Trust towards detecting the RSSDF attackers.

The rest of three simulations are performed to analyze the probabilities of detection(Q_d) and false alarms(Q_f) of the *AND*, *OR* and *Majority* rule at the RSSDF attack, obtained from 5000 rounds of Monte Carlo detection. Generally, the higher Q_d and lower Q_f a CSS rule holds, the better performance will be achieved.

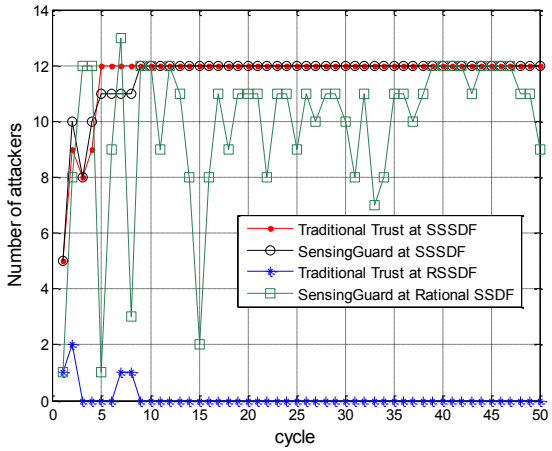

Fig. 5. SensingGuard vs. Traditional Trust under detecting types of SSDF attackers.

As we know, in the Always-using, the attackers always report the false sensing data "1" although there is no PU spectrum bands. Such threat can increase Q_f significantly.

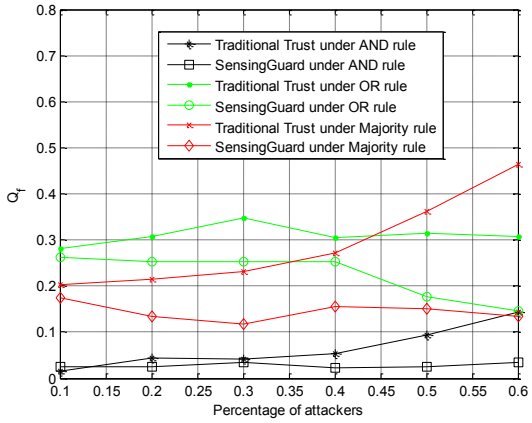

Fig. 6. SensingGuard vs. Traditional Trust at Always-using

In the second simulation, we vary the percentage of the RSSDF attackers to observe Q_f. As shown in **Fig. 6**, SensingGuard filters out some RSSDF attackers through analyzing the two factors (τ, ψ) before the data fusion, thus the Q_f curve of SensingGuard is better than Traditional Trust in guarding the *AND*, *OR* and *Majority* rule. For the *AND* rule, the damage of RSSDF is limited in Q_f. The reason is that the final decision under the *AND* rule is "1" only when the sensing data are all "1". For the *OR* rule, the damage of RSSDF is biggest. The final decision under the *OR* rule is "1" so long as an attacker fake the sensing data "1". Without effective preventive measures, it can be seen that the damage of RSSDF under the *Majority* rule amplifies with the percentage of attackers. The attackers can change the final decision when they become the majority. This is because the majority of sensing data are "1" under the *Majority* rule, the final decision is "1".

As we know, in the Always-free, the attackers always report the false sensing data "0", although there are PUs using their spectrum bands. Such threat can decrease Q_d significantly. In the third simulation, we vary the percentage of RSSDF attackers to observe Q_d. As shown in **Fig. 7**, the Q_d curve of SensingGuard is better than Traditional Trust in guarding the *AND* and *Majority* rule. For the *AND* rule, the damage of RSSDF is biggest in Q_d. The final decision under the *AND* rule is "0" so long as an attacker fake the sensing data "0". For the *Majority* rule, the Q_f curve decreases with the percentage of attackers if not any effective preventive measures are adopted. We can also observe that the damage of RSSDF to the *OR* rule is limited. The reason is that the final decision under the *OR* rule is "1" only when an SU reports real sensing data "1".

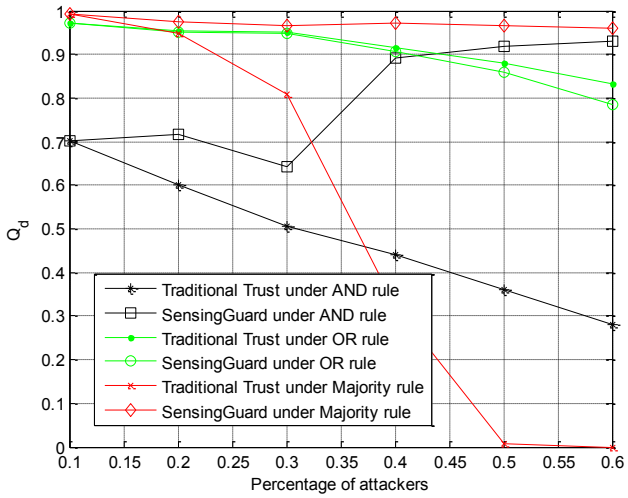

Fig. 7. SensingGuard vs. Traditional Trust at Always-free

Finally, we analyze the receiver operating characteristic (ROC) curves, the relationship between Q_f and Q_d, which is usually to validate the entire performance of CSS. In the fourth simulation, RSSDF attackers launch the Always-using or Always-free threat at random, and the percentage of attackers is fixed at 40%. As shown in **Fig. 8**, we can see that the ROC curves of SensingGuard under the *AND* and *Majority* rule are better than Traditional Trust, which indicates that SensingGuard can enhance the performance of the two fusion rules significantly after filtering out the RSSDF attackers. When one SU reports "1" under the *OR* rule, the PU signal is considered to be present. It can be seen that the RSSDF attackers have a little influence on the *OR* rule by analyzing the ROC curves.

5. Conclusion

In this paper, we describe the the discover of RSSDF attack in which attackers can keep up highe trustworhiness and thus being difficultly detected in the traditional trust schemes. Meanwhile, a novel trust scheme called SensingGuard is proposed to mitigate the harmful effect of RSSDF attackers and enhance the performance of CSS. Unlike the traditional trust schemes, SensingGuard evaluates the trustworthiness of each SU by considering different PUs and filters out RSSDF attackers through analyzing the two factors (τ, ψ). Simulation results show that the proposed scheme can counter RSSDF attack effectively and achieve better performance under the *AND*, *OR* and *Majority* rule.

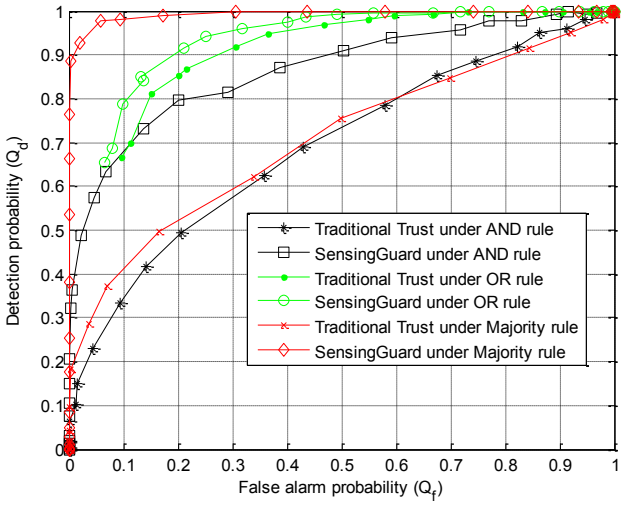

Fig. 8. ROC curves of SensingGuard vs. Traditional Trust

References

[1] Federal Communications Commission, "Spectrum Policy Task Force," Rep. ET Docket no. 02-135, Nov. 2002. http://www.fcc.gov/sptf/files/SEWGFinalReport_1.pdf

[2] J. Mitola and G. Q. Maguire, "Cognitive radio: making software radios more personal," *IEEE Personal on Communication*, vol. 6, no. 4, pp. 13-18, August, 1999. Article (CrossRef Link)

[3] G. U, Hwang and S. Roy, "Design and analysis of optimal random access policies in cognitive radio networks," *IEEE Transactions on Communications*, vol. 60, no. 1, pp. 121-131, January, 2012. Article (CrossRef Link)

[4] D. Cabric, S. Mishra and R. Brodersen, "Implementation issues in spectrum sensing for cognitive radios," in *Proc. of Asilomar Conference on Signals, Systems, and Computers*, pp. 772–776, November 7-10, 2004. Article (CrossRef Link)

[5] F. R Yu, M. Huang and H. Tang, "Biologically inspired consensus-based spectrum sensing in mobile Ad hoc networks with cognitive radios," *IEEE Network*, vol. 24, no. 3, pp. 26-30, June, 2010. Article (CrossRef Link)

[6] R. L Chen, J. M Park and Y. T Hou, "Toward Secure Distributed Spectrum Sensing in Cognitive Radio Networks," *IEEE Communications Magazine*, vol. 46, no. 4, pp. 50-55, April, 2008. Article (CrossRef Link)

[7] T. Qin, H. Yu and C. Leung, "Towards a trust-aware cognitive radio architecture," *ACM SIGMOBILE Mobile Computing and Communications Review*, vol. 13, no. 2, pp. 86-95, April, 2009. Article (CrossRef Link)

[8] K. Zeng, P. Pawełczak and D. Cabri, "Reputation-based cooperative spectrum sensing with trusted nodes assistance," *IEEE Communications Letters*, vol. 14, no. 3, pp. 26-228, March, 2010. Article (CrossRef Link)
[9] T. Qin, C, Leung, C. Y Mao and Y. Q Chen, "Trust-aware resource allocation in a cognitive radio system," in *Proc. of 2012 IEEE International Conference on Computer Science and Automation Engineering (CSAE),* pp. 797-801, May 25-27, 2012. Article (CrossRef Link)
[10] Q. Q Pei, B. B Yuan, L. Li and H. N Li, "A sensing and etiquette reputation-based trust management for centralized cognitive radio networks," *Neurocomputing*, vol. 101, no. 4, pp. 129-138, July, 2013. Article (CrossRef Link)
[11] R. Chen, J. M. Park and K. Bian, "Robust distributed spectrum sensing in cognitive radio networks," in *Proc. of 27th IEEE INFOCOM Conference*, pp. 1876–1884, April 13-18, 2008. Article (CrossRef Link)
[12] I. F. Akyildiz, B. F. Lo and R. Balakrishnan, "Cooperative spectrum sensing in cognitive radio networks: A survey," *Physical Communication*, vol. 4, no. 1, February, pp. 40-62, 2011. Article (CrossRef Link)
[13] E. Peh, Y. C Liang, Y. L Guan and Y. G Zeng, "Optimization of Cooperative Sensing in Cognitive Radio Networks: A Sensing-Throughput Tradeoff View," *IEEE Transactions on Vehicular Technology*, vol. 58, no. 9, pp. 5294-5299, November, 2009. Article (CrossRef Link)
[14] I. F. Akyildiz, W. Y Lee and K. R Chowdhury, "CRAHNs: cognitive radio ad hoc networks," *Ad Hoc Networks*, vol. 7, no.5, pp. 810-836, October, 2009. Article (CrossRef Link)
[15] H. Tang, "Some physical layer issues of wide-band cognitive radio systems," in *Proc. of 1st IEEE International Symposium on New Frontiers in Dynamic Spectrum Access Networks*, pp. 151–159, November 8-11, 2005. Article (CrossRef Link)
[16] W. Zhang and R. K Mallik, "Cooperative Spectrum Sensing Optimization in Cognitive Radio Networks," in *Proc. of IEEE International Conference on Communications*, pp. 3411-3415, May 19-23, 2008. Article (CrossRef Link)
[17] M. A Morid and M. Shajari, "An enhanced e-commerce trust model for community based centralized systems," *Electronic Commerce Research*, vol. 12, no. 4, pp. 409-427, November, 2012. Article (CrossRef Link)
[18] X. Y Li, F. Zhou and X. D Yang. Scalable Feedback Aggregating (SFA) Overlay for Large-Scale P2P Trust Management. *IEEE Transactions on Parallel and Distributed Systems*, vol. 23, no. 10, pp. 1944-1957, October, 2012. Article (CrossRef Link)
[19] A. Boukerche, Y. Ren and R. Pazzi. "An adaptive computational trust model for mobile ad hoc networks," in *Proc. the 5th International Conference on Wireless Communications and Mobile Computing*, pp. 191-195, June 21-24, 2009. Article (CrossRef Link)
[20] A. Mohaisen, N. Hopper and Y. Kim, "Keep Your Friends Close: Incorporating Trust into Social-Network-based Sybil Defenses," in *Proc. of 30th IEEE INFOCOM Conference*, pp. 1943-1951, April 10-15, 2011. Article (CrossRef Link)
[21] A. Jøsang and R. Ismail, "The beta reputation system", in *Proc. the 15th Bled Electronic Commence Conference*, pp. 1-14, June 17-19, 2002. Article (CrossRef Link)
[22] Gamma function. http://en.wikipedia.org/wiki/Gamma_function
[23] H. Rif-Pous, M. Blasco and C. Garrigues, "Review of Robust Cooperative Spectrum Sensing Techniques for Cognitive Radio Networks," *Wireless Personal Communications*, vol. 67, no. 2, pp. 175-198, November, 2011. Article (CrossRef Link)

Jingyu Feng received his B.S. degree in electrical information science and technology from Lanzhou University of Technology, China, in 2006. He received his Ph.D. degree from Xidian University, China, in 2011. He is currently a lecturer in Department of Communication Engineering, Xi'an University of Post & Telecommunication, China. He is also a Postdoctor of University of Chinese Academy of Sciences, China. His main research interests include wireless security, trust management and cooperative spectrum sensing.

Yuqing Zhang is a professor and supervisor of Ph.D. candidates of Graduate University of Chinese Academy of Sciences. He received his B.S. and M.S. degree in computer science from Xidian University in 1987 and 1990 respectively. He received his Ph.D. degree in Cryptography from Xidian University in 2000. His research interests include cryptography, information security and network protocol security.

Guangyue Lu is a professor in Department of Communication Engineering, Xi'an University of Post & Telecommunication. He received his B.S. and M.S. degree from Yangtze University, China, in 1992 and 1995 respectively. He received his Ph.D. degree from Xidian University in 1999. His research interests include wireless communication, cognitive radio and cooperative spectrum sensing.

Liang Zhang received his B.S. degree in Department of Computer Science and Technology, from Xi'an University of Post & Telecommunications in 2009. He is currently working toward the master degree in Department of Communication Engineering, Xi'an University of Post & Telecommunication. His interests include cognitive radio networks and cooperative spectrum sensing.