World Wide Web (2018) 21:185–200
DOI 10.1007/s11280-017-0465-x

Trustworthy service composition with secure data transmission in sensor networks

Tao Zhang[1] · Lele Zheng[1] · Yongzhi Wang[1] · Yulong Shen[1] · Ning Xi[1] · Jianfeng Ma[1] · Jianming Yong[2]

Received: 30 September 2016 / Revised: 17 April 2017 /
Accepted: 25 April 2017 / Published online: 10 May 2017

Abstract As the basis of the Internet of Things (IoT), sensor networks have materialized its computation and communication capability into anything in our modern lives. Service composition provides us a promising way to cooperate various sensors to build more powerful IoT applications over sensor networks. However, the limited capability of sensor node poses great challenges not only to trustworthy service composition but also to secure data aggregation. The complex composite structure, computation-intensive evaluation, and massive data transmission become burdens for service composition in sensor networks. To overcome these issues, this paper proposes a distributed approach to enable efficient trustworthy service composition with secure data transmission in sensor networks. By analyzing dependency relationships, the rules for computing service trust and data trust are proposed based a multi-level trust model. Then, each target component service can be evaluated independently through a model checker. Moreover, an identity-based aggregate signature is introduced in the composite evaluation to guarantee the secure data transmission among different components. The experimental results show that our approach not only achieves efficient trustworthy service composition with complex invocation structures, but also reduces the costs in the secure data transmission.

Keywords Service composition · Data transmission · Trust · Security · Sensor networks

This article belongs of the Topical Collection: *Special Issue on Security and Privacy of IoT*
Guest Editors: Tarik Taleb, Zonghua Zhang, and Hua Wang

✉ Tao Zhang
taozhang@xidian.edu.cn

[1] School of Computer Science and Technology, Xidian University, Xidian, China

[2] School of Management and Enterprise, University of Southern Queensland, Toowoomba, Australia

1 Introduction

Currently, the Internet of Things (IoT) has been widely adopted in various crucial systems [1], such as urban sensing, highway transportation, intelligent community and granary management. With an increasing number of sensors connected to the Internet, it is possible to achieve more comfortable and safer living conditions. For instance, London has deployed all sorts of sensors to improve the quality of urban life, i.e. congestion control, traffic prediction, weather forecast, and air quality monitoring [2]. In addition to dedicated IoT platforms, mobile devices such as smartphones, cameras and GPS, are also utilized as sensing devices.

In service-oriented sensor networks, the functionality provided by each sensor node is treated as a service [7]. Services can be composed together dynamically and rapidly to develop novel and powerful applications. For a variety of candidate services, consumers can select qualified ones with respect to their specific functional and security requirements. However, some services may be malicious. They may not deliver its task with promised quality, or cause confidential data leakage to the public. Therefore, trust and security are the main concerns of service composition in sensor networks.

Conceptually, trust is the belief of one party in another to act in the interest of the former party in a given situation [16]. Trust mechanism provides an incentive for good behavior and help prevent financial losses during the interaction with an unknown entity. Trust evaluation of service vendors is key to the success of service composition. Traditional service evaluation approaches [8, 11, 15] use ratings from direct perceptions or peers to represent the trust property of a composite service with invocation structures. But a single aggregated trust value cannot ensure secure data transmission through trusted service components. Consequently, some issues still remain unsettled.

The first issue in most previous work is that they do not specifically consider the structures of composite services. A composite service is the service dynamically composed of outsourced service components from multiple domains with complex invocation structures [23]. Even with higher trust level, the service may still be composed of untrustworthy components. Therefore, we should analyze the dependency relationships among different objects to ensure trustworthy data transmission in the service composition.

The second issue is the cost of verification sensor. In general, sensor nodes are resource-limited. It would costs greatly to evaluate the trustworthiness of target service in an centralized way. With the lack of in-network trusted authorities, it is required to evaluate the target invocations in a precise and efficient way. There are various abstract models for data-oriented service invocations. For example, Hutter et al. [20] proposed an information flow control approach that can specify security level to the confidential data based on type-based model [5]. Then, type-labeled confidential data can be dynamic spread among those secure composite services. Yang et al. [20] analyzed the distributed workflow using a hierarchical state machine, with which the information can be propagated in the secure workflow. Moreover, Xi et al. [17] analyzed the intra- and inter-dependencies using a lattice-based information model, then the security properties in service chain can be verified through model checker tools [18]. For trust management in service-oriented architecture, time-constraint service execution workflow can be modeled as multiple basic invocations [23], then the trustworthy composite service can be composed with competitive and cooperative relationships. The invocation relationship models in existing work can be employed to analyze the data dependencies in sequential and conditional workflows, and a distributed evaluation approach is required to be developed for the dependent components, which would further reduce the costs for computation-intensive evaluations.

The last issue is that it is difficult to protect the security of data in the service workflow. It is risky to transmit service invocation data through the composite service workflow. There are various work to protect data transmission among different structured composed entities. For example, Liang et al. [9] proposed a secure review aggregation system that enabled users to submit their review contents in a distributed and cooperative way. Their approaches used aggregate signature on circulated tokens to provide efficient data aggregation and transmission. Moreover, Liu et al. [10] proposed an efficient attribute based sequential aggregate signature for sequential data aggregation. Therefore, it is necessary to integrate aggregate signature during composition steps to guarantee data security and reduce transmission costs.

To solve the above issues, we present a distributed service composition approach to guarantee trustworthy service evaluation and secure data transmission among service components. In our approach, the dynamic dependencies among different objects are analyzed using program dependency graph. Based on the lattice-based trust model, the trust level of each component service can be aggregated and verified in the abstract form. The security of data in service workflow can be further guaranteed with identity-based aggregate signature. Finally, a compositional trustworthy service composition with secure data transmission algorithm is proposed. Through theoretical analysis, we show our approach can construct trustworthy composite services in the distributed way. We further evaluate the performance of our approach. The simulation results show that our approach can work effectively for sensor networks.

Compared with the preliminary version of this paper [24], there are several main additional extensions: 1) We theoretically prove our trust evaluation approach at component and composite level; 2) We model the service function as abstract states with state transitions, and verify the trust constraints defined in these states through model checker; and 3) We introduce a practical identity-based aggregate signature to ensure the secure data transmission in service workflow.

The reminder of this paper is organized as follows. Section 2 introduces the related models and definitions. Sections 3 and 4 propose service evaluation and data transmission approach at both component and composite levels for composing trustworthy services in sensor networks. Experiments are conducted in Section 5 for illustrating the effectiveness of our approach. Finally, we conclude this paper and present future directions in Section 6.

2 Models and definitions

2.1 Network model

A typical sensor network system is illustrated in Figure 1, which consists of multiple sensor domains. These domains are interconnected with network switches to support heterogenous data transmission. In the networks, there are various services s that can be composed together to generate some powerful composite applications. For the sake of focus and simplicity, it is assumed that there is only one trust management authority (TMA) responsible for trust evaluation and security key generation.

Some sensors may offer similar functionalities with different quality-of-service (QoS). Even with the same trust level, services with equivalent functionality may have entirely different implementations. As our focus is on composing trustworthy service components, we assume each service is associated with an evaluated trust level. Some evaluation approaches can be studied in the literature [4, 6, 13, 19, 21].

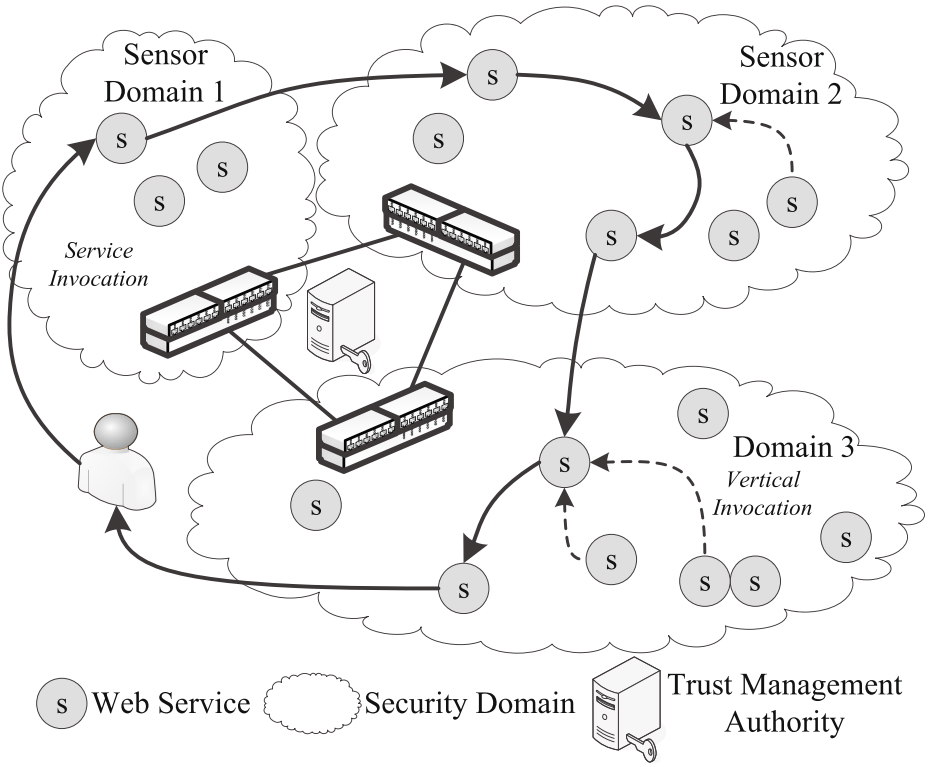

Figure 1 Service-oriented Sensor Networks

2.2 Dataflow service model

The dataflow service model is illustrated in Figure 2, where service s_i which receives its input data $s_i.I$ and generates its output data $s_i.O$. There are three types of data flow: horizontal, vertical and hybrid. Horizontal data flow happens when service s_i receives data from its predecessors and sends data to its successors; vertical data flow is the computation flow when service s_i invokes other service components in its computation function, and hybrid

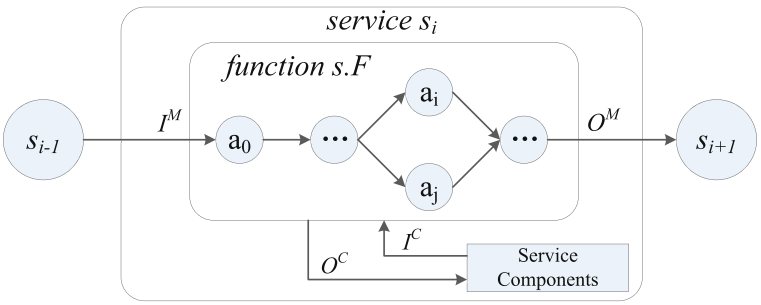

Figure 2 Data Flow Service Model

data flow is composed of both horizontal and vertical flows. The definition of data flow service model can be defined as Definition 1.

Definition 1 (Dataflow Service Model) Service s is defined as a quintuple $\langle id, F, T, I, O\rangle$, where

- $s.id$ is the identifier of s,
- $s.F$ is the computation function of s,
- $s.I$ is the set of all input data objects of s, $s.I = s.I^M \cup s.I^C$ where,
  - $s.I^M = \{s.I_1^M, s.I_2^M, ..., s.I_n^M\}$ is the data objects that receive from its predecessors.
  - $s.I^C = \{s.I_1^C, s.I_2^C, ..., s.I_n^C\}$ is the data objects that receive from its service components in its computation function $s.F$.
- $s.O$ is the set of all output data objects of s, $s.O = s.O^M \cup s.O^C$ where,
  - $s.O^M = \{s.O_1^M, s.O_2^M, ..., s.O_n^M\}$ is the data objects that send to its successors.
  - $s.O^C = \{s.O_1^C, s.O_2^C, ..., s.O_n^C\}$ is the data objects that send to its service components in its computation function $s.F$.

The function $s_i.F$ of service s_i can be abstracted as the following language syntax, which is given in Figure 3.

2.3 Multi-level trust model

We use multi-level trust model [6] to describe the trust property of each associated objects, which include basic service components and the generated data. The multi-level trust model can be defined on a lattice, as Definition 2.

Definition 2 (Multi-level Trust Model) Multi-level trust model is defined as a lattice (TL, $\leq$), where TL is finite set of trust levels that are totally ordered by $\leq$.

In this paper, the trust level of service s is defined as $T(s)$, the trust level of data object transmitted through composite service S_c is defined as $\delta(o)$. Additionally, consumers and vendors are able to specify trust levels for their data, defined as τ.

3 Trustworthy service evaluation for composite services

From the perspective for dataflow service model, data objects are transmitted through different service components in a composite service S_c. S_c is considered as trustworthy if it

$$
\begin{aligned}
&f ::= a; f \\
&a ::= skip | var := e | a; a | input(in, e) | output(out, e) \\
&\qquad | if(e)\ then\ a\ else\ a | while(e)\ a \\
&e ::= var | eRe \\
&R ::= + | - | = | <
\end{aligned}
$$

Figure 3 Language Syntax of $s_i.F$

contains no untrustworthy components. In the process of service composition, a service consumer first submits the functional requirement and trust requirement τ_u, then the composer returns a concrete services S_c satisfying these requirements. This τ_u can be considered as the global constraint on the composite service S_c. Moreover, each service component s_i would specify a trust level $\tau(s)$ for its generated data to prevent trust degradation. This $\tau(s)$ can be considered as local constraints. To guarantee the global and local constraint, we analyze the dependency relationships in composite services to evaluate the trust level of each candidate.

3.1 Program graph-based trustworthy evaluation for component service

From definition 1, service s_i is modeled as a computation function $s_i.F$ with input $s_i.I$ and output $s_i.O$. $s_i.F$, which contains a lot of activities concerned with control and computation dependencies, which can be considered as a reactive system responding to the external events. Based on the abstract language syntax of $s_i.F$ in Figure 3, Program Dependency Graph (PDG) is employed to model the dependencies in $s_i.F$. Then, the relationships among different objects in PDG can be obtained by the program backward slicing [14]. For each object o, we use intra-dependency set $D_a(o)$ to store the internal service dependencies. Based on the intra-dependency set $D_a(o)$, the trust level of a service s_i can be aggregated. For $\forall u \in s_i.O$,

$$T(u) = \underset{\min}{\sqcup}\{T(s), \underset{\min}{\sqcup}\{T(v)\}\} \quad v \in s_i.I \wedge v \in D_a(u) \tag{1}$$

Equation (1) means that the aggregated trust level of service component is set as the lowest trust level (denoted as $\underset{\min}{\sqcup}$) of the dependent service components in $s.F$ of service s. As the operations of service s_i is in the dependency set of its output, i.e. $\forall u \in s_i.O, s_i \in D_a(u)$, the (1) can be simplified as follows. For $\forall u \in s_i.O$,

$$T(u) = \underset{\min}{\sqcup}\{T(v)\} \quad v \in s_i.I \wedge v \in D_a(u) \tag{2}$$

To guarantee the global and local trust constraint, the trust level of recipient services should be equal to or higher than the trust level specified by the service consumer and vendors respectively. In other words, the generated data should be transmitted among the trustworthy subsequent services. From the above analysis, the trust level of data objects can be computed as follows. For $\forall u \in s_i.O$,

$$\begin{cases} \delta(v) = \tau_u \quad v \in s_0.I^M \\ \delta(v) = \tau(s_{i-1}.O^M) \quad v \in s_i.I^M, i > 0 \\ \delta(u) \geq \delta(v) \quad v \in s_i.I \wedge D_a(u), i \geq 0 \end{cases} \tag{3}$$

The above equations mean that the global trust level of composite service S_c is specified by the service consumers as the trust level on initial service s_0, the aggregated trust level of data objects is computed from the dependent service components in $s_i.F$ of service s_i, and the local trust level may be upgraded by the service vendors. Therefore, we use the highest trust level (denoted as $\underset{\max}{\sqcup}$) as the trust level of the output data objects of each component service s_i. As a result, the above equations can be summarized as (4).

$$\delta(u) = \underset{\max}{\sqcup}\{\delta(v)\} \quad v \in s_i.I \wedge D_a(u) \tag{4}$$

From (4), we can obtain the following theorem.

Theorem 1 *For* $\forall u \in s_i.O$, *there is*

$$\delta(u) \geq \delta(v) \quad v \in s_i.I \wedge v \in D_a(u)$$

Based on the multi-level trust model and dependency relationship of service components, if all the data objects are processed in the trustworthy components in the execution of $s_i.F$, the service component s_i is considered trustworthy. Then, the trustworthy service component in s_i can be defined as follows.

Definition 3 (Trustworthy Service Component) A service component s_i is considered trustworthy if it satisfies that for $\forall u \in s_i.O$ and $v \in s_i.I \wedge D_a(u)$, there is

$$T(u) \geq \delta(u) = \underset{\max}{\sqcup}\{\delta(v)\}$$

3.2 Distributed trustworthy evaluation for composite service

For a composite service S_c, data are transmitted through different service components. For example, the input data objects of s_i may come from two sources, $s_i.I^M$ and $s_i.I^C$, where $s_i.I^M$ is the input from its predecessors, and $s_i.I^C$ is from the intra-composed components. The output data objects of s_i may be further processed by its successors, and finally deliver the data to the end service s_n. This kind of dependency relationship is referred to as inter-service dependency, denoted as $D_t(o)$, which can be formally defined as follows.

Definition 4 (Inter-Service Dependency) For objects $v \in s_i$ and $u \in s_j (j > i)$, v is in the set of inter-service dependency $D_t(u)$, if v and u satisfy either of the following two conditions:

(1) $i = j - 1$
$\exists w_1 \in s_i.O, w_2 \in s_j.I, w_1 = w_2,$
$(v \in D_a(w_1) \wedge w_1 = v) \vee (w_2 \in D_a(u) \wedge w_2 = u)$

(2) $i \neq j - 1$
$\exists w \in s_k, i < k < j$
$v \in D_t(w) \wedge w \in D_t(u)$

Based on the $D_t(o)$, we can determine which services consume a give data object in S_c. To guarantee trustworthy data transmission in S_c, we define the trustworthy composite service follows.

Definition 5 (Trustworthy Composite Service) A composite service is considered trustworthy if it satisfies that for $\forall u \in S_c.Out$, there is

$$T(u) \geq \underset{\max}{\sqcup}\{\delta(v)\} \quad v \in S_c.In \wedge (v \in D_a(u) \vee v \in D_t(u))$$

As the trust level of output data is computed by individual service, each service can be distributed evaluated by its successors. To support this, we give the following lemmas and theorem.

Lemma 1 *In a composite service* S_c, *for* $\forall u \in s_i.O$, *and* $v \in s_j.I \wedge (v \in D_a(u) \vee v \in D_t(u))$, $0 \leq j \leq i$, *there is* $\delta(u) \geq \delta(v)$.

Proof First, let n = 1, there are two services s_0 and s_1 in composite service S_c.

For $\forall u \in s_0.O$ and $v \in s_0.I \wedge v \in D_a(u)$, theorem 1 provides that for there is $\delta(u) \geq \delta(v)$. And, since s_0 has no predecessor that $s_0.I = \varnothing$, no inter-service dependency exists. This lemma is proved.

For $\forall u \in s_1.O$, we have to consider the following two cases:

Case 1 $j = 1, v \in s_1.I \wedge v \in D_a(u)$. It can be proved from theorem 1 that $\delta(u) \geq \delta(v)$.

Case 2 $j = 0, v \in s_0.I \wedge v \in D_t(u)$. From the definition 4, we can obtain that $\exists w_1$ and $\exists w_2$, such that

$$w_1 \in s_0.O^M = w_2 \in s_1.I^M \quad (v \in D_a(w_1) \wedge w_2 \in D_a(u))$$

Therefore,

$$\delta(w_1) = \delta(w_2) \tag{5}$$

From Theorem 1, we get

$$\delta(u) \geq \delta(w_2) \tag{6}$$

$$\delta(w_1) \geq \delta(v) \tag{7}$$

From (5), (6) and (7), we get

$$\delta(u) \geq \delta(v)$$

Thus, it has been shown that lemma 1 holds when $n = 1$.

Next, suppose lemma 1 holds when $n = k - 1$, that is,
for $\forall u \in s_i.O^M, 0 \leq i \leq k-1$, there is

$$\begin{aligned} &\delta(u) \geq \delta(v) \\ &v \in s_j.I \wedge (v \in D_a(u) \vee v \in D_t(u)), \quad 0 \leq j \leq i \end{aligned} \tag{8}$$

When $n = k$, the lemma is proved as follows.

For $\forall u \in s_i.O^M, 0 \leq i \leq k$, we consider the following two cases.

Case 1 $j = i$,for $\forall v \in s_i.I \wedge v \in D_a(u)$. Similar to the previous proof, we can get $\delta(u) \geq \delta(v)$.

Case 2 $0 \leq j \leq i$, for $\forall v \in s_j.I \wedge v \in D_t(u)$. From the definition 4, we can obtain that $\exists w_1$ and $\exists w_2$, such that

$$\begin{aligned} &w_1 \in s_{n-1}.O^M = w_2 \in s_n.I^M \\ &(v \in D_a(w_1) \vee v \in D_t(w_1)) \wedge w_2 \in D_a(u) \end{aligned}$$

Therefore,

$$\delta(w_1) = \delta(w_2) \tag{9}$$

From Theorem 1, we get

$$\delta(u) \geq \delta(w_2) \tag{10}$$

And, the hypothesis provides that for $\forall w_1 \in s_i.O^M, 0 \leq i \leq k-1$, there is

$$\delta(w_1) \geq \delta(v) \tag{11}$$

From (9), (10) and (11), we get

$$\delta(u) \geq \delta(v)$$

In conclusion, when $n = k$, the lemma is proved. □

Lemma 2 *If first m component services are trustworthy in S_c, it satisfies that for $\forall u \in s_i.O, 0 \leq i \leq m$, and $v \in s_j.I \wedge (v \in D_a(u) \vee v \in D_t(u), 0 \leq j \leq i$, there is $T(u) \geq \delta(v)$.*

Proof For $\forall u \in s_i.O$, we consider the following two cases.
Case 1 $j = i, \forall v \in s_i.I \wedge v \in D_a(u)$. It follows from Definition 3 that $T(u) \geq \delta(v)$.
Case 2 $0 \leq j < i, \forall v \in s_j.I \wedge v \in D_t(u)$. From the definition 4, we can obtain that $\exists w_1$, and $\exists w_2$, such that

$$w_1 \in s_{n-1}.O^M = w_2 \in s_n.I^M$$
$$(v \in D_a(w_1) \vee v \in D_t(w_1)) \wedge w_2 \in D_a(u)$$

Therefore,

$$\delta(w_1) = \delta(w_2) \tag{12}$$

Definition 3 provides that

$$T(u) \geq \delta(u) \tag{13}$$

From Theorem 1, we get

$$\delta(u) \geq \delta(w_2) \tag{14}$$

Moreover, from Lemma 1, we have

$$\delta(w_1) \geq \delta(v) \tag{15}$$

Finally, from (12), (13), (14) and (15), we get

$$T(u) \geq \delta(v)$$

In conclusion, this lemma is proved. □

Theorem 2 *For a composite service S_c, we say S_c is trustworthy if each component service s_i satisfies the following two conditions:*

(1) In each component service s_i, for $\forall u \in s_i.O$ and $v \in s_i.I \wedge D_a(u)$, there is $T(u) \geq \delta(u) = \underset{\max}{\sqcup}\{\delta(v)\}$.

(2) In services s_i and s_j with invocation relationship, for $\forall w_1 \in s_i.O^M, \forall w_2 \in s_j.I^M$ and $T(w_1) > T(w_2)$, there is $\delta(w_1) > \delta(w_2)$.

Proof Let $m = n + 1$, where n is the number of services in S_c. Theorem 2 can be proved from Lemma 2. □

4 Trustworthy service composition with secure data transmission

4.1 Trustworthy service evaluation framework in sensor networks

In the process of service composition, our approach selects services from the candidate set C_i for each task s_i in S_c. The candidate set is defined as $C_i = \{s_{i,j} | 0 \leq i \leq n, 1 \leq j \leq |C_i|\}$. Based on the trustworthy composite service model, we can evaluate each candidate service in C_i according to the Theorem 2. Service evaluation framework is illustrated as Figure 4.

The framework consists of two elements, 1) candidate services (C), and 2) trust management authority (*TMA*). For each candidate $s_{i,j}$ in C_i, there are two phases to evaluate the trust level, i.e. the component evaluation phase and the composite evaluation phase. In the component evaluation phase, the framework verifies whether a component $s_{i,j}$ is trustworthy by model checking, then generate a signature for the qualified services for the following

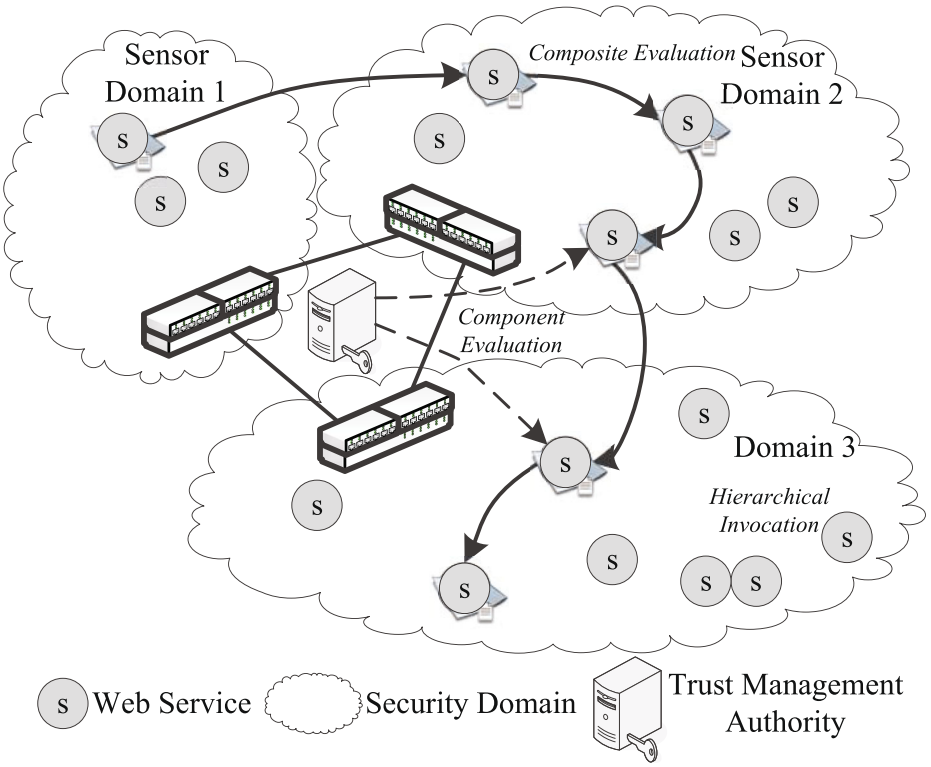

Figure 4 Trust Evaluation Framework

secure data transmission. The composite evaluation happens when a service generates output data objects to its successors. A trustworthy composite service can be deployed after all the services are found in the networks.

TMA plays the role of both model checker and public key generator. And, there is only one *TMA* in our framework. The multi-authority signature schemes are extensively, which are out of scope in our work.

4.2 Component service evaluation by model checking

In this phase, *TMA* evaluates whether each candidate service in C_i satisfies the trust constraints. For the dynamic dependencies in Definition 4, the self-composition procedure is used to evaluated the properties of activities in the functions. The self-composition procedure translates the states of $s_i.F$ into the Label Transition Systems (LTS), which can be defined as follows:

Definition 6 A state of the execution function $s_i.F$ of component service s_i is a triple $\mu = \langle I, O, V \rangle$, where I, O, and V represent the mappings from its input, output and variables to the values respectively, i.e. $I : s_i.I \rightarrow val$, $O : s_i.O \rightarrow val$, $V : s_i.var \rightarrow val$, and val is the range of values in $s_i.F$.

The initial state of $s_i.F$ is μ_0 and the final state is μ_f. Intermediate state μ_i may transit to another state μ_j with the activity a. As some component services may be composed in $s_i.F$, the trust level may be changed based on the data flow in $s_i.var$, which also lead to the state transitions. Therefore, based on the Definition 6, the definition of LTS is given as follows.

Definition 7 The LTS of a service s_i is modeled as $\mathcal{M} = (\alpha, \mu, \rightarrow)$, where μ represents the set of states of $s_i.F$, and $\rightarrow$ represents the transition between states in μ.

In this paper, we employ the state transitions in our previous work [18] for describing the abstract language syntax of service function $s_i.F$ in Figure 3. The transition rules $\Phi(\alpha, n_k, n_l, \rightarrow)$ is given in Figure 5, where n_k and n_l represent the entry and exit state of the activity α in $s_i.F$.

TMA applies self-composition procedure to verify the trust properties of each candidate. For a service model $\mathcal{M}$ with initial state μ_0, *TMA* creates a copy of $\mathcal{M}$ as $\mathcal{M}'$ with μ'_0, and also copies the trust level of input objects from M to M', i.e. for $\forall o \in s_i.I$, such that $\delta_{\mu'_0}(o) = \delta_{\mu_0}(o)$.

The model checker verifies whether two abstractions of properties would stay at the equivalent states. For two states $\mu = \langle I, O, V\rangle$ and $\mu' = \langle I, O, V\rangle$, $\mu.I \sim \mu'.I$ means for $\forall o \in s_i.I$, there is $\mu.I(o) = \mu'.I(o)$, and $\mu.O \sim \mu'.O$ means for $\forall o \in s_i.O$, there is $\mu.O(o) = \mu'.O(o)$. Then, we can obtain the following theorem from Definition 5.

Theorem 3 *The component service s_i is trustworthy if and only if that for $\forall\mu_0$ and μ'_0, if $\mu_0.I \sim \mu'_0.I$, there is $\mu_f.O \sim \mu'_f.O$, where $\mu_f = \wedge\{s_i.F(\mu_0)\}$ and $\mu'_f = \wedge\{s_i.F(\mu'_0)\}$.*

According to Theorem 3, we can verify whether all output data objects with different trust levels can reach the same state using the following assertion s_i, i.e. for $\forall u_x \in s_i.O$,

$$\textbf{assert}\left(\bigwedge_{L\le x\le\delta(u)}\left(\mu_f.O(u_x) == \mu'_f.O(u_x)\right)\right).$$

Next, we use model checking tool SPIN [3] to verify the trust properties of each component services s_i. SPIN accepts the self-composition service models $\mathcal{M}$ and $\mathcal{M}'$ as input, and takes security property expressions as assertions. If no error returns, component service s_i satisfies the trust constraints; otherwise it returns some traces of counter-examples.

If component service s_i is trustworthy, it generates a message m_i for its successor evaluation. m_i contains two parts of information, i.e. service output and data trust requirement δ. In order to guarantee the s_i can invoke s_j in a secure channel, we sign the message m_i for each s_i, which will be described in the next subsection. As the component service evaluation does not rely on the signature, the properties can be verified by model checker offline. The computation cost of compositional evaluation will decrease accordingly.

4.3 Composite evaluation with secure data transmission

For trustworthy service s_i, the identity-based (ID-based) aggregate signature scheme in [12] is used to record the invocation path, which enables secure data transmission in composed component services. The ID-based aggregate signature scheme generates the private key sk_i for service s_i with $s_i.id$, then service s_i uses this private key sk_i to sign data message

$$
\begin{aligned}
&\Phi(skip, n_k, n_l, \rightarrow) = \{\langle n_k\rangle \rightarrow \langle n_l\rangle | I' = I \wedge O' = O \wedge V' = V\}\\
&\Phi(input(in, var), n_k, n_l, \rightarrow)\\
&\qquad = \{\langle n_k\rangle \rightarrow \langle n_l\rangle | I' = I \wedge O' = O \wedge V'(var)\\
&\qquad = I(in) \wedge (\forall var' \neq var, V'(var') = V(var'))\}\\
&\Phi(output(out, var), n_k, n_l, \rightarrow)\\
&\qquad = \{\langle n_k\rangle \rightarrow \langle n_l\rangle | I' = I \wedge O'(out)\\
&\qquad = V(var) \wedge V' = V')\}\\
&\Phi(var := e, n_k, n_l, \rightarrow)\\
&\qquad = \{\langle n_k\rangle \rightarrow \langle n_l\rangle | I' = I \wedge O' = O \wedge V'(var)\\
&\qquad = V(e) \wedge (\forall var' \neq var, V'(var') = V(var'))\}\\
&\Phi(a; a', n_k, n_l, \rightarrow) = \Phi(a, n_k, n_l, \rightarrow) \cup \Phi(a, n_k, n_l, \rightarrow)\\
&\Phi(if\ (e)\ then\ \alpha\ else\ \alpha', n_k, n_l, \rightarrow)\\
&\qquad = \{\langle n_k\rangle \rightarrow \langle n_r\rangle | I' = I \wedge O' = O \wedge V' = V \wedge e\}\\
&\qquad \cup \{\langle n_k\rangle \rightarrow \langle n_t\rangle | I' = I \wedge O' = O \wedge V' = V \wedge \neg e\}\\
&\qquad \cup \Phi(\alpha, n_r, n_l, \rightarrow) \cup \Phi(\alpha', n_t, n_l, \rightarrow)\\
&\Phi(while\ (e)\ \alpha, n_k, n_l, \rightarrow)\\
&\qquad = \{\langle n_k\rangle \rightarrow \langle n_r\rangle | I' = I \wedge O' = O \wedge V' = V \wedge e\}\\
&\qquad \cup \{\langle n_k\rangle \rightarrow \langle n_l\rangle | I' = I \wedge O' = O \wedge V' = V \wedge \neg e\}\\
&\qquad \cup \Phi(\alpha, n_r, n_k, \rightarrow)
\end{aligned}
$$

Figure 5 State Transitions of $s_i.F$

m_i with signature $\sigma_i = sign_{sk_i}(m_i)$, and sends the signature to its successors. Then the composed service can generate its aggregate signature as

$$(\{s_j.id, m_j, \sigma_j, 0 \leq j \leq n\}, \sigma*).$$

The aggregated signature $\sigma*$ is valid if the function $verify(\{s_j.id, m_j, \sigma_j, 0 \leq j \leq n\})$ outputs *True*.

If $s_{i,j}$ passes the evaluation, TMA gives a signature σ_i to instantiate the s_i. Then composed services play the role of requestor to invoke the candidate s_i and sign the m_i with ID-based aggregate signature. Then the TMA verifies the trust properties of each invocations based on the Theorem 2.

For illegal candidate, it will be removed from the candidate set. Moreover, if there is no legal composite service, we will examine which candidates would result in the data trust level aggradation.

4.4 Service composition algorithm in sensor networks

Based on the component verification and composite evaluation, we propose a distributed trustworthy service composition algorithm with secure data transmission in sensor networks. Given the functional and trust requirement, TMA first evaluate each candidate evaluated candidates C_i to obtain the qualified candidates C_i^p, then each service in C_i^p continues its stepwise composite evaluation. As service execution graph is an end-to-end graph, s_0 and s_n are the deterministic initiator and ending nodes, where s_0 receives consumer's input while s_n generates the computation results. The details of the service composition algorithm is presented in Algorithm 1.

Algorithm 1: Trustworthy Service Composition

Input: composed service s_{com} , Candidate service set C_i for s_i, Ending service s_n
Output: Composite Service: S_c

```
1  TMA waits for the signal to start evaluation;
2  if Signal is INVOKE then
3      if s_com = s_0 then
4          s_0 pushes its successor candidates in C_i to the waiting queue succ;
5      if s_{i,j} == s_n then
6          if succ == ∅ then
7              TMA sends signal SUCCESS and trustworthy S_c to consumer;
8              return S_c;
9          else
10             Pop succ to s_{i,j};
11             TMA evaluates each candidate s_{i,j}, and the illegal ones is removed;
12             TMA verifies the invocation trust level between s_com and legal
               successors based on Theorem 2;
13             TMA issues security keys to sign the trustworthy service s_{i,j} using
               ID-based signature and aggregates the signatures in set s_com;
14     if There is no trustworthy services then
15         TMA sends signal FAIL ;
16     TMA sends signal INVOKE to the newly composed service;
17 if Signal is FAIL then
18     C_i.counter increases;
19     if C_i.counter == |C_i| then
20         TMA notices that there is no trustworthy service composition;
21         return null;
22     else
23         TMA sends FAIL to its predecessors;
24 return S_c;
```

In Algorithm 1, we use three types of signals to represent the different steps in the composition, i.e. INVOKE, SUCCESS and FAIL. The signal **INVOKE** is used when the composed service is ready to compose its successors. In this step, the invocation relationships are considered. If all the candidates passed evaluations, the signal **SUCCESS** is used to deliver the final computation results to the end consumer. In addition, the signal **FAIL** tells that the current service is untrustworthy, we will judge whether it is caused by the upgraded data trust level specified by either service itself, or some components in its computation function. If there is no available candidates, there is no trustworthy composition results for the end consumer. The consumer may need to modify the global trust constraint for the next composition.

5 Experiments and evaluations

5.1 Performance evaluation

According to Algorithm 1, for each verification, the time complexity is $O(m)$, where $m = |C_i|$ is the number of candidates in C_i. Therefore, at each verification step, the time complexity is $O(m * p) \approx O(m^2)$, where $p = |C_i^p|$ is the number of candidates in C_i^p. For each composite service, there are n tasks to be executed (n is a constant). Therefore, the total complexity for service evaluation is $O(nm) + O(nm^2) \approx O(nm^2)$. However, the global

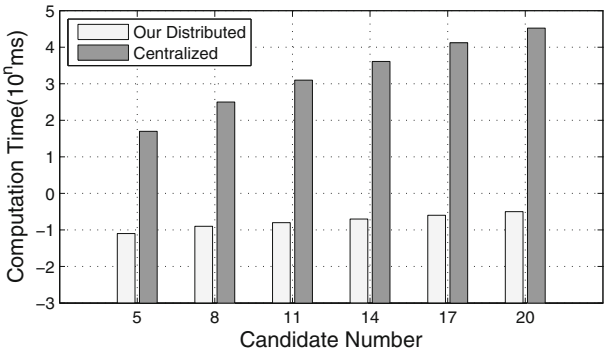

Figure 6 Evaluation Time

model checking has the complexity of $O(m^n)$, which is greatly larger than ours $O(nm^2)$. In conclusion, our approach can reduce the cost in service evaluations.

5.2 Experimental evaluation

We use the service composition structure in [22] as the test case, which supports the complex input and output relationships. We use SPIN to verify the abstract states in service functionality and NS-3 to simulate our composition algorithm.

Figure 6 plots the evaluation time on the requestor service in relation with different number of candidates. It can be observed that evaluation time increases significantly with the number of candidate in the centralized way, while in our approach it just increases slightly. This is because in the centralized evaluation, the authority has to search for all the possible states of the composed services, and the search range would rise sharply at an exponential rate. However, in our approach, each service can evaluate its successor directly, thus needs far less evaluation time.

Regarding the communication cost on the requestor, our approach also outperforms the centralized approach (See Figure 7). This is because in centralized evaluation, requestors have to communicate with all other nodes, while in our approach requestors only need to send evaluation and invocation information to its successor in each composition step, which leads to the low communication cost.

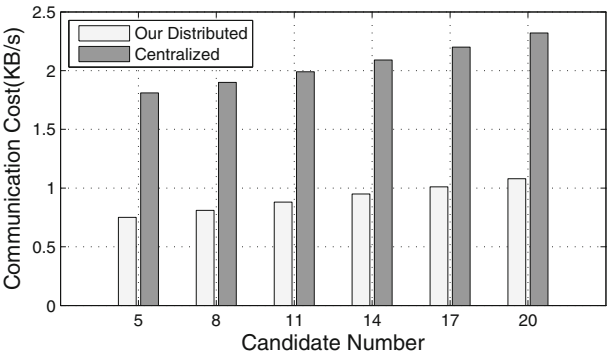

Figure 7 Communication Cost

6 Conclusion

This paper has proposed a trustworthy service composition approach with secure data transmission in sensor networks. Our approach supports the following features:

1) Engage program dependency graph to analyze the dependency relationship and enable distributed trustworthy evaluation;

2) Based on the lattice-based trust model, the property abstraction and state transition for each component service can be verified through model checker.

3) Introduce ID-based aggregate signature in the composition steps, the execution step can be evaluated stepwise by each component.

For future work, we would like to improve our current evaluation framework to be more adaptive for mobile environment. Besides, adversary may act as an virtualized sensor nodes to launch sybil attack, we will also develop a sybil-defensive trust-aware framework for IoT applications.

Acknowledgements We wish to thank the anonymous reviewers for their highly valuable and constructive comments. This paper is supported by the National Natural Science Foundation of China (No. 61602365, U1536202, 61571352, 61373173 and 61602364), the Open Fund of the Chinese Key Laboratory of the Grain Information Processing and Control (No. KFJJ-2015-202), and the Fundamental Research Funds for the Central Universities (No. XJS15075).

References

1. Atzori, L., Iera, A., Morabito, G.: The internet of things: A survey. Comput. Netw. **54**(15), 2787 – 2805 (2010). doi:10.1016/j.comnet.2010.05.010
2. Boyle, D.E., Yates, D.C., Yeatman, E.M.: Urban sensor data streams: London 2013. IEEE Internet Comput. **17**(6), 12–20 (2013). doi:10.1109/MIC.2013.85
3. Holzmann, G.J.: The model checker spin. IEEE Trans. Softw. Eng. **23**(5), 279–295 (1997). doi:10.1109/32.588521
4. Huang, J., Peng, M., Wang, H., Cao, J., Gao, W., Zhang, X.: A probabilistic method for emerging topic tracking in microblog stream. World Wide Web **20**(2), 325–350 (2017). doi:10.1007/s11280-016-0390-4
5. Hutter, D., Volkamer, M.: Information flow control to secure dynamic web service composition, pp. 196–210. Springer Berlin Heidelberg, Berlin, Heidelberg (2006)
6. Jøsang, A., Ismail, R., Boyd, C.: A survey of trust and reputation systems for online service provision. Decis. Support. Syst. **43**(2), 618 – 644 (2007). doi:10.1016/j.dss.2005.05.019
7. Kyusakov, R., Eliasson, J., Delsing, J., Van Deventer, J., Gustafsson, J.: Integration of wireless sensor and actuator nodes with it infrastructure using service-oriented architecture. IEEE Trans. Indust. Inform. **9**(1), 43–51 (2013). doi:10.1109/TII.2012.2198655
8. Li, L., Wang, Y.: Trust evaluation in composite services selection and discovery. In: SCC, pp. 482–485 (2009). doi:10.1109/SCC.2009.70
9. Liang, X., Li, X., Lu, R., Lin, X., Shen, X.: Seer: A secure and efficient service review system for service-oriented mobile social networks. In: 2012 IEEE 32nd International Conference on Distributed Computing Systems (ICDCS), pp. 647–656 (2012). doi:10.1109/ICDCS.2012.46
10. Liu, X., Zhu, H., Ma, J., Li, Q., Xiong, J.: Efficient attribute based sequential aggregate signature for wireless sensor networks. Int. J. Sensor Netw. **16**(3), 172–184 (2014)
11. Malik, Z., Bouguettaya, A.: Rateweb: Reputation assessment for trust establishment among web services. VLDB J. **18**(4), 885–911 (2009)
12. Shen, L., Ma, J., Liu, X., Wei, F., Miao, M.: A secure and efficient id-based aggregate signature scheme for wireless sensor networks. IEEE Internet Things J. **PP**(99), 1–1 (2016). doi:10.1109/JIOT.2016.2557487
13. Sherchan, W., Nepal, S., Paris, C.: A survey of trust in social networks. ACM Comput. Surv. **45**(4), 47:1–47:33 (2013). doi:10.1145/2501654.2501661
14. Snelting, G., Robschink, T., Krinke, J.: Efficient path conditions in dependence graphs for software safety analysis. ACM Trans. Softw. Eng. Methodol. **15**(4), 410–457 (2006). doi:10.1145/1178625.1178628

15. Wang, Y., Li, L.: Two-dimensional trust rating aggregations in service-oriented applications. IEEE Trans. Serv. Comput. **4**(4), 257–271 (2011). doi:10.1109/TSC.2010.39
16. Wang, Y., Lin, K.J.: Reputation-oriented trustworthy computing in e-commerce environments. IEEE Internet Comput. **12**(4), 55–59 (2008). doi:10.1109/MIC.2008.84
17. Xi, N., Ma, J., Sun, C., Zhang, T.: Decentralized information flow verification framework for the service chain composition in mobile computing environments. In: 2013 IEEE 20th International Conference on Web Services (ICWS), pp. 563–570 (2013). doi:10.1109/ICWS.2013.81
18. Xi, N., Sun, C., Ma, J., Shen, Y.: Secure service composition with information flow control in service clouds. Futur. Gener. Comput. Syst. **49**, 142 – 148 (2015). doi:10.1016/j.future.2014.12.009
19. Yan, Z., Zhang, P., Vasilakos, A.V.: A survey on trust management for internet of things. J. Netw. Comput. Appl. **42**, 120 – 134 (2014). doi:10.1016/j.jnca.2014.01.014
20. Yang, P., Yang, Z., Lu, S.: Formal modeling and analysis of scientific workflows using hierarchical state machines. In: IEEE International Conference on e-Science and Grid Computing (2007). doi:10.1109/E-SCIENCE.2007.35
21. Yu, H., Shen, Z., Leung, C., Miao, C., Lesser, V.R.: A survey of multi-agent trust management systems. IEEE Access **1**, 35–50 (2013). doi:10.1109/ACCESS.2013.2259892
22. Zhang, T., Ma, J., Li, Q., Xi, N., Sun, C.: Trust-based service composition in multi-domain environments under time constraint. Sci. China Inf. Sci. **57**(9), 1–16 (2014). doi:10.1007/s11432-014-5104-x
23. Zhang, T., Ma, J., Sun, C., Li, Q., Xi, N.: Service composition in multi-domain environment under time constraint. In: ICWS, pp. 227–234 (2013). doi:10.1109/ICWS.2013.39
24. Zhang, T., Ma, J., Xi, N., Liu, X., Liu, Z., Xiong, J.: Trustworthy service composition in service-oriented mobile social networks. In: 2014 IEEE International Conference on Web Services (ICWS), pp. 684–687 (2014). doi:10.1109/ICWS.2014.102