Revealing Business Relationships – Eavesdropping Cross-organizational Collaboration in the Internet of Services

André Miede  
*TU Darmstadt,* miede@KOM.tudarmstadt.de

Gökhan S¸ims¸ek  
*TU Darmstadt,* simsek@KOM.tudarmstadt.de

Stefan Schulte  
*TU Darmstadt,* schulte@KOM.tudarmstadt.de

Daniel F. Abawi  
*Hochschule für Technik und Wirtschaft des Saarlandes,* abawi@htw-saarland.de

Julian Eckert  
*TU Darmstadt,* eckert@KOM.tudarmstadt.de

*See next page for additional authors*

Follow this and additional works at: [http://aisel.aisnet.org/wi2011](http://aisel.aisnet.org/wi2011)
Revealing Business Relationships –
Eavesdropping Cross-organizational Collaboration in the
Internet of Services

André Miede  
Multimedia Communications  
Lab (KOM) – TU Darmstadt  
Rundeturmstraße 10, 64283  
Darmstadt, Germany  
miede@KOM.tu-darmstadt.de

Gökhan Şimşek  
Multimedia Communications  
Lab (KOM) – TU Darmstadt  
Rundeturmstraße 10, 64283  
Darmstadt, Germany  
simsek@KOM.tu-darmstadt.de

Stefan Schulte  
Multimedia Communications  
Lab (KOM) – TU Darmstadt  
Rundeturmstraße 10, 64283  
Darmstadt, Germany  
schulte@KOM.tu-darmstadt.de

Daniel F. Abawi  
Hochschule für Technik und Wirtschaft des Saarlandes  
Waldhausweg 14, 66123  
Saarbrücken, Germany  
abawi@htw-saarland.de

Julian Eckert  
Multimedia Communications  
Lab (KOM) – TU Darmstadt  
Rundeturmstraße 10, 64283  
Darmstadt, Germany  
eckert@KOM.tu-darmstadt.de

Ralf Steinmetz  
Multimedia Communications  
Lab (KOM) – TU Darmstadt  
Rundeturmstraße 10, 64283  
Darmstadt, Germany  
steinmetz@KOM.tu-darmstadt.de

ABSTRACT
The Internet of Services is envisioned as a global Service-oriented Architecture enabling collaboration across organizational boundaries. However, by monitoring communication endpoints, attackers can create detailed profiles of service consumers and providers even if typical security mechanisms such as message encryption are used. In a business context, this traffic analysis threatens the relationship anonymity of the participants and can reveal sensitive information about an organization’s underlying business processes or a service provider’s client base. In this paper, we discuss the simulation-based evaluation of different attack scenarios regarding the identification of the service compositions an organization uses. Thus, we offer insights regarding the limits of anonymity for cross-organizational collaboration in the Internet of Services.

Keywords
Security, Anonymity, Internet of Services, Service-oriented Architectures, Cross-organizational Collaboration

1. INTRODUCTION
Modern global economies have become fast-paced and highly competitive, thus, requiring organizations to adapt both quickly and continuously to changing circumstances and requirements. An important factor to achieve this goal is the underlying enterprise Information Technology (IT), which has to integrate both internal and external systems.

The paradigm of Service-oriented Architectures (SOAs) [22] offers technological and organizational means in order to improve the alignment between the functional and the IT side, i.e., by enabling service-based, cross-organizational workflows. In the last years, Web services have become both a mature and successful technology for implementing the SOA paradigm.

For the near future, the Internet of Services is envisioned as a global SOA further facilitating cross-organizational collaboration [4,26]. The Internet of Services provides the foundation for complex business value networks by supporting the composition and aggregation of existing services to value-added services, i.e., using market places as intermediaries between service consumers and providers. Furthermore, it is a business model using the Internet as a medium for the retrieval, combination, and utilization of interoperable services. For example, market places could build compositions using services from different providers and offer these compositions as best practices for recurring process needs to service consumers.

In order to enable such service-based, cross-organizational collaboration, the security of the communication channels used, exchanged messages, and participating systems is a necessity. Regarding the security of Web service technology, substantial advancements have been achieved in the last years as discussed in the standard literature on Web service security [3,12,25]. However, several technology-independent and service-specific attacks on SOA have been identified recently, especially in the Internet of Services context [17,18].

One of these attacks aims at identifying the existence of relationships between collaborating organizations: By ob-
A comprehensive overview of mechanisms and systems in order to achieve different types of anonymity in communication networks is given, e.g., by Edman and Yener [8]. However, even if such standard anonymity mechanisms are deployed and used correctly, attacks mounted at the edges of such networks and aiming at typical long-term business relationships are very likely to be successful. Thus, the goal of the paper at hand is the following: We investigate how an adaptation of a typical anonymity attack with respect to service compositions threatens the relationship anonymity of service consumers and providers in the Internet of Services. This is done by measuring the attacker’s success using metrics from the field of Information Retrieval while varying key system parameters such as the number of service providers, the composition complexity, or the number of observed collaborations.

The rest of the paper is structured as follows: Section 2 and 3 outline the analysis and design of our evaluation, i.e., how it was set-up and why we made certain design decisions. Subsequently, Section 4 analyzes and discusses selected results. In order to place our contributions within the body of existing research, Section 5 discusses the most relevant related work in this area. Section 6 sums up the findings and closes with a brief outlook on future work.

2. ANALYSIS

In this section, the foundation of our research is presented, i.e., the underlying assumptions, the research question, and reasons for the selected means to answer this question.

2.1 Attack Selection

First of all, how does a typical attack on anonymity and anonymity systems look like and where has it to be mounted? Assuming that basic anonymity systems are being used by the communicating participants, there are two basic choices for an attack:

1. Attack the anonymity network itself: The attacker tries to follow the trail of a message along the nodes of an anonymity system, e.g., as described by Guan et al. [9] (cf. Figure 2). However, this is very difficult because of the (usually) high number of participating nodes and the used security mechanisms. Thus, many nodes would have to be compromised in order to cover the whole route. In addition, a single missing node on the route makes this kind of attack even more difficult, because messages are hard to correlate between the nodes.

2. Attack the anonymity network edges, i.e., incoming and outgoing connections of the nodes. This is more straightforward, because the attacker only has to monitor the messages on a single edge, and he only has to be inside the anonymity network to be able to perform this attack. However, he might be outside the network. In this case, he may be able to send out probes to derive the nodes of the network. This strategy has been called relationship anonymity in the standard literature on anonymity research [23]. This means that an adversary cannot sufficiently distinguish whether the sender and recipient of a particular message are related or not. It is important to understand that this kind of anonymity does not apply to the sender and recipient of the message, i.e., they know each other. It refers only to third parties, i.e., parties that are neither sender nor recipient of the message.

A simple, but tangible example from the financial services domain is a generic credit application process, i.e., where credit ratings for customers are retrieved from an external rating agency. More about such an example is shown in Figure 1: The bank works on credit applications from its customers, e.g., first entering and storing the customer’s data in its systems (using an internal access service “Store Data”). A possible next step would be calculating the concrete credit offer, which is a service composition “Calculate Offer” consisting of both internal and external services, e.g., for external credit history ratings. After that, another service would route the resulting information to a human decision maker “Route to Person in Charge” for triggering the next steps. Finally, if this person’s decision is positive, the offer would be made. This results in another service composition “Make Offer”, again consisting of both internal and external services, e.g., for notifying the customer about the decision, processing the payout of the credit, etc.
Unlike the standard disclosure attacks, the so-called “Statistical Disclosure Attack” does not provide the attacker with definite information about the communication relationships but with a probability of each potential relationship. Basically, this requires the attacker to observe a large amount of interactions from which he can calculate the respective relationship probabilities. Selected details of these calculations are discussed as part of our simulation model in Section 3.2.

We chose this attack type for our evaluation, because it is a particular threat for strategic, long-term relationships, i.e., relationships that are custom in the field of service-based cross-organizational collaboration. In addition, the attack is basically independent from the used anonymity system, thus, based on certain assumptions that will be outlined below, it is a threat for most deployed anonymity systems.

For this paper, the Statistical Disclosure Attack is adapted for the Internet of Services scenario, i.e., attackers aim to identify the service compositions that organizations use for executing their processes. More details on these adaptations are given in Section 3.

2.2 Research Question
As outlined above, we assume for our research that organizations use external services (and compositions thereof) for executing their processes. Furthermore, we assume that basic countermeasures against traffic analysis are in place, thus, non-trivial attacks are needed because an attacker cannot just intercept any message in order to retrieve sender and recipient information from its header.

From this and the selected attack type follows the research question we try to answer in this paper: “How does an adap-

3. SIMULATION DESIGN AND SETUP
This section discusses the underlying design decisions of our simulation model, i.e., the general assumptions, an overview of the model, brief implementation information, and the different evaluated configurations are presented.

3.1 General Simulation Assumptions
For our simulation model, we assume the following regarding the different entities: The system uses end-to-end encryption that cannot be broken in time. Furthermore, it delivers messages to recipients in batches, e.g., using a so-called “Threshold Mix” [5,13].

The attacker is passive and static, i.e., the attacker observes only and does not adapt his attack behavior. In addition, he can observe messages leaving and entering the network (not necessarily all messages) and can guess when a message
entering is likely to leave. Furthermore, the attacker knows the anonymity system’s parameters (e.g., batchsize) and the market place’s offerings, i.e., what compositions are available and what providers they consist of. Also, the participants have a consistent communication behavior, i.e., they have strategical, long-term communication relationships and do not change their service providers frequently.

These are typical and well-proven assumptions in the field of anonymity research similar to those in the related work, e.g., assuming nearly worst-case scenarios from the anonymity system’s point of view. Based on this foundation, the model is described in the next section.

3.2 Simulation Model

The attack is modeled as a stochastic model of a time-step simulation. For an overview, the basic flow of the simulation is shown in Figure 4. In addition, these steps are described in the following in more detail:

1. Initialization: The first step of the initialization is the generation of the overall supply of service compositions. Here, a service composition consists of at least one service, i.e., the ID of the respective service provider. The assignment of service providers to compositions is done randomly, in our model based either on a uniform popularity distribution of the providers or a Zipfian one [31]. A Zipfian distribution means that the relative probability of the i-th most popular service provider to be used is proportional to $1/i^\alpha$, leading to a more realistic selection probability of service providers. Breslau et al. showed that the requests of Web pages follow a Zipfian distribution with an exponent $\alpha$ of about 0.75 [2]. This finding is adopted for our simulation because it provides a realistic estimation of service offerings on the Internet. Zipfian distributions were used before in the area of anonymity research, e.g., by Shmatikov and Wang [27]. From the generated service compositions, the organization under observation, here called Alice Corp. (“Alice”) selects a certain number for executing her (business) processes.

2. Cross-organizational Collaboration: With the service offerings and the compositions used by Alice determined, the cross-organizational collaboration starts. At each time-step of the simulation, Alice contacts one of the service providers that are part of her used service compositions. As in real collaboration scenarios, Alice is not the only one communicating with service providers. Thus, there is also the so-called “background”, i.e., other service consumers communicate with different service providers as well at each time-step. This background fills the remaining slots of the anonymity system’s batch of size $b$. The recipients of the background are denoted by the vector $\vec{u}$ and distributed according to the general provider distribution, i.e., either uniform or Zipfian as described above. The attacker is assumed to know or approximate this distribution for his calculations. At each time-step $t$, the attacker intercepts the batch of messages $\vec{s}_t$ or a fraction thereof, depending on the attacker’s spread.

3. Attacker’s Calculations: At regular intervals, e.g., time-step $t$, the attacker performs calculations for identifying Alice’s service providers in general and the corresponding service compositions in particular. The core of the calculations is based on the formal model of the classic Statistical Disclosure Attack [5]. Thus, as proven by Danezis, the attacker approximates Alice’s recipients ($\vec{v}$) after $t$ time-steps based on the observed output of the anonymity system ($\sum_{k=1}^{b} \vec{o}_k$), the batchsize $b$, and the known/approximated background distribution ($\vec{u}$):

$$\vec{v} \approx \frac{\sum_{k=1}^{b} \vec{o}_k}{t} - (b - 1)\vec{u}$$

(1)

Using vector $\vec{v}$ as approximated above and the stored observed vectors $\vec{u}$, the attacker calculates each vector $\vec{r}_k$ by multiplying each element of $\vec{o}_k$ (observed in round $k$) with the respective element of $\vec{v}$, afterwards normalizing the results using their dot product ($||\vec{v} \cdot \vec{o}_k||$):

$$\vec{r}_k = \frac{\vec{v} \cdot \vec{o}_k}{||\vec{v} \cdot \vec{o}_k||}$$

(2)

This then contains the probabilities about the service providers Alice communicated with in time-step $k$, i.e., the higher the resulting value of an element in $\vec{r}_k$, the more likely this service provider was used by Alice in round $k$.

The attacker then uses the maximum probability of each $\vec{r}_k$, i.e., the service provider Alice most likely communicated with in time-step $k$. This knowledge is then combined with the knowledge about the available service compositions, e.g., retrieved from the market places in the Internet of Services. Thus, the probability of each composition containing the most likely provider of time-step $k$ is increased. Because the attacker does this iteratively for all observed time-steps, he builds an internal model of the service compositions Alice is using, assigning a probability to each possible composition.

The validation of the model is done as suggested by North and Macal [20]. It is based on the identified requirements, the plausibility of the assumptions, and the general development process, because the Internet of Services is not yet available for a comparison validation. These necessary aspects were discussed above and found to be valid for our model.

Using this specification as a foundation, a brief overview of
the model’s implementation is given in the next section.

3.3 Model Implementation
The simulation model is implemented using Repast Symphony, an agent-based modeling toolkit\(^1\). Repast has the advantage of providing a frame for the general simulation, such as methods that are executed at each time-step of the simulation, a graphical user interface for configuring simulation parameters, and built-in functionality for tracing and logging simulation results. Furthermore, Repast models can be implemented using the Java programming language, thus, there is no need to learn yet another special modeling language.

A particular implementation aspect is the generation of random numbers based on the Zipfian (or Zeta) distribution in order to achieve more realistic results than using a basic uniform distribution for randomly selecting service providers. We calculated numbers of this distribution based on the following procedure, where \( F(x) \) can be any cumulative distribution function \([14]\):

\[
F(x) = \Pr(X \leq x), \quad y = F(x) \iff x = F^{-1}(y) \quad (3)
\]

Thus, a random number \( X \) of distribution \( F(x) \) can be generated by using \( X = F^{-1}(U) \), where \( U \) is uniformly distributed. In our case, we used the Apache Commons Mathematics Library version 2.1\(^2\) for calculating the inverse cumulative probability \( F^{-1}(U) \) of the Zipfian distribution, extending it regarding much faster random number generation as required for our simulation runs.

For verification purposes, test cases with pre-calculated results of the attack are compared to the (non-stochastic) results of simple attack runs of our model. These test cases can be used, e.g., for verifying the results of the simulation model after changes to the underlying algorithms have been made.

The next section describes how the model and its implementation can be configured in order to reflect different attackers and attack scenarios.

3.4 Configuration
For the evaluation runs of our simulation model, it can be configured in a variety of ways, modeling different attack scenarios and attacker capabilities. The used configuration parameters are described in the following:

**Service compositions** are generated based on the maximum number of services per composition (\(mSC\)) and the total number of (different) service compositions (\(C\)). From these, Alice selects randomly a number of used compositions (\(aC\)).

**Service providers** influence the simulation by their overall number (\(N\)) and their popularity distribution, which can be either uniform or Zipfian. Furthermore, the Zipfian distribution is detailed by its skewness. Based on Breslau et al.’s seminal work on Zipfian distributions in the Web as shown above, we chose a skewness of \(\alpha = 0.75\) for our simulations \([2]\).

The **anonymity system** is characterized by the batchsize \((b)\), i.e., the number of messages leaving the system per time-step.

The **attacker’s capabilities** are modeled by the parameter spread \((S)\), which denotes the percentage of how many outgoing messages the attacker can intercept.

For the attack to have any chance of success, the following relationship must hold, as shown by Danezis \([5]\):

\[
m < \frac{N}{b-1} \quad (4)
\]

However, the parameter \(m\), i.e., the number of Alice’s recipients, is no longer directly available in our model, because it is partially based on random variables. It can be approximated before-hand by \(aC \times mSC\), which serves as an upper bound for \(m\). At run-time, i.e., after the initialization phase, \(m\) can be determined exactly by counting the number of distinct service providers in all service compositions used by Alice.

As a preparation for the simulation runs, we performed a number of calibration runs in order to determine the most important parameters to be observed. The main distinction is the provider popularity, modeled by a uniform or Zipfian distribution. These are then evaluated regarding the impact of the overall number of service providers, the maximum number of services per composition, the number of compositions used by Alice, and the attacker’s spread, i.e., his access to outgoing messages. Based on these configuration decisions, the next section describes the performed simulations and discusses selected results.

4. OUTPUT ANALYSIS AND DISCUSSION
This section discusses the used evaluation metrics and selected results of the performed simulations. Due to space-constraints, some results are omitted here, e.g., the impact of the number of available compositions \((C)\).

For each single configuration, e.g., each different value for \(N\), 100 simulation runs were performed in order to achieve a suitable level of confidence for assessing the results \([11]\).

4.1 Evaluation Metrics
In order to evaluate the attacker’s performance, i.e., his success regarding the identification of Alice’s service compositions, we use well-proven metrics from the field of Information Retrieval \([15]\). These metrics were chosen, because we consider the problem of retrieving a set of “relevant” documents from a larger set of documents to be very similar to the attacker’s goal of identifying certain compositions from the overall supply of service compositions. In addition, our scenario has the advantage of specifying definitely, what “relevant” documents, i.e., compositions, are. The ones used by Alice. Figure 5 shows the respective sets of our scenario in order to apply typical Information Retrieval metrics for our evaluation. Alice’s compositions (relevant) are denoted with \(A\), the attacker’s identifications (retrieved) with \(B\).

\(^1\)http://repast.sourceforge.net, last access on January 3, 2011.

\(^2\)http://commons.apache.org/math/, last access on January 3, 2011.
Thus, the metrics mean in our scenario the following: Precision denotes the fraction of the identified compositions that are actually used by Alice, i.e., $\frac{|A \cap B|}{|B|}$. Recall denotes the fraction of how many of Alice’s compositions could be identified by the attacker, i.e., $\frac{|A \cap B|}{|A|}$.

As the attacker assembles a ranked list of Alice’s compositions, i.e., sorted by their respective probability, this can be considered for the evaluation as well: Mean Average Precision (MAP) considers the position of Alice’s compositions in the attacker’s list of identified compositions. The more of Alice’s compositions are at the top of the list, thus, having a high probability, the higher the MAP. However, even if more of Alice’s compositions are identified correctly, MAP will decrease if these are ranked lower. More details on these metrics can be found in the works by Manning et al. or Moffat and Zobel [15,19].

### 4.2 Impact of the Number of Service Providers

The results for different numbers of service providers, i.e., precision, recall, and MAP, are shown in Figure 6. In all these figures, the three metrics on the left are based on a uniform provider distribution while the three on the right are based on a Zipfian one. Furthermore, the y-axis uses a logarithmic scale in order to facilitate the comparison between the two distributions.

The measurements were taken after a rather short amount of interactions, i.e., 1,000 collaborations between Alice and her service providers. The reason behind this is to investigate how variations of certain system parameters influence the attacker’s results.

**Uniform**: Recall is basically not affected by the number of service providers, as all of Alice’s compositions are identified. However, opposed to what one might expect, precision rises if the number of different service providers increases.

The reason for this might be, that Alice’s and the background’s interactions are distributed over a larger number of possibilities, thus, they stand out more prominently. MAP differs significantly from precision and has a very high and about constant value over the observed $N$. This means, that Alice’s compositions are always at the top of the list of compositions assembled by the attacker. In general, the attacker

**Zipfian**: As above, the attacker’s results against a Zipfian provider distribution are much worse than for the uniform one. The general trend is similar, but the degradation is more graceful than for $N$, e.g., recall decreases with the increase of Alice’s compositions. However, precision is rather unaffected by this increase, but for high numbers of used compositions, it even increases as well (with low and about

---

**Figure 5**: Retrieved/relevant sets of our attack scenario for applying Information Retrieval metrics.

**Figure 6**: Varying the number of service providers $N$, each after 1,000 time-steps, 95% confidence intervals ($b = 125, C = 1,000, aC = 10, mSC = 8, S = 100\%$).
constant values of MAP, though). This trend is due to the rising chances of the attacker identifying compositions because there are more of them.

4.4 Impact of the Maximum Number of Services per Composition
The results for different values for the maximum of services per composition are shown in Figure 8.

4.5 Impact of the Attacker’s Spread
The results for different spread values of the attacker are shown in Figure 9.

4.6 Impact of the Number of Time-Steps
The above evaluations considered a short, fixed amount of time in order to determine the impact of different system parameters. In addition, it is also beneficial to investigate how the metrics evolve over time, i.e., where the limits of anonymity in cross-organizational collaboration or of
the attacker could be. For this, we used the following selected scenarios from above with an underlying Zipfian service provider distribution.

Figure 10: Evolution of attacker’s results with $aC = 100$ and Zipfian distribution, 95% confidence intervals ($N = 200,000$, $b = 125$, $C = 1,000$, $mSC = 8$, $S = 100\%$).

Figure 11: Evolution of attacker’s results with $mSC = 80$ and Zipfian distribution, 95% confidence intervals ($N = 200,000$, $b = 125$, $C = 1,000$, $aC = 10$, $S = 100\%$).

High number of Alice’s compositions ($ac = 100$): As shown in Figure 10, more time, i.e., more observations, gives the attacker a slight advantage. While the additional knowledge regarding MAP from 1,000 to 25,000 time-steps is high, additional 25,000 observations do not contribute much, reaching in total still only about 10%. Precision and recall do not improve much as well, so that a significant improvement after even more observations is unlikely.

High composition complexity ($mSC = 80$): Observing a scenario with a high maximum number of services per composition over a longer time does not improve the attacker’s results as depicted in Figure 11. Precision and MAP remain about constant at their very low values between 1 and 2%. However, the very high recall can be slightly improved from 1,000 to 25,000 time-steps, but not much after that.

Medium spread ($S = 50\%$): Regarding the attacker’s bad results for irregular access to the anonymity system’s messages, additional time does not help as shown in Figure 12. All observed metrics remain nearly constant at their low values, precision and MAP at most reaching 1%. As pointed out above, significant gains cannot be expected even after more time-steps.

In general, making more observations is only one approach for the attacker and not a very good one, i.e., it is most likely only used as a last resort. Other approaches, e.g., improving the internal calculations, the general model, etc. are more likely to threaten the overall anonymity in the Internet of Services. Such possibilities will be discussed below as future work.

A summary and further discussion of the overall results is given in Section 6.

5. RELATED WORK

Regarding specific attacks on anonymity, this paper focuses on attacks on the boundaries of anonymity systems, i.e., the Statistical Disclosure Attack [5] from the general class of Intersection/Disclosure Attacks [13]. Within this attack class, only simple sender-recipient-relationships were investigated so far.

However, the concept of service compositions in the Internet of Services introduces additional complexity, i.e., with respect to the relationships between service consumers, the compositions they use, and the networks of service providers that constitute these compositions. Thus, in order to gain knowledge about an organization’s processes, for example, by identifying the service compositions it uses, an attacker
has to confirm the relationships between service consumers and providers, inferring from this knowledge the service compositions this organization is most likely to use.

As a starting point for our investigations of this new scenario, we adapted the basic variant of the Statistical Disclosure Attack. Therefore, other variants and extensions were not considered so far, e.g., the use of Mix networks or different batching algorithms as described by Mathewson and Dingledine [16] or utilizing graph theory in order to relax specific user behavior assumptions of the attack model as introduced by Troncoso et al. [28].

Furthermore, the attack on relationship anonymity investigated in this paper must not be mixed up with the extensive research on Web service privacy, e.g., [10, 29, 30]. Web service privacy deals with the content of the exchanged messages, e.g., users’ personal data, and how this information is further processed and possibly shared. It is an important aspect of the overall privacy goal “confidentiality”, not of anonymity [1, 7].

On the other hand, the important aspect of anonymous communication between the different organizational participants of an SOA, i.e., with respect to third parties in order to conceal important business relationships has not been addressed so far. Further aspects of anonymity, i.e., the issue of anonymous Web service provision as well as consumption is addressed, e.g., by Papastergiou et al. [21]. However, it is questionable whether this is a desirable functionality for cross-organizational collaboration where it is important that both service consumer and provider know and trust each other, i.e., for legal reasons such as compliance or audit.

6. CONCLUSIONS AND FUTURE WORK

In this paper, we investigated the impact of attacks that aim at revealing business relationships of collaborating organizations. These attacks are of a particular danger in the field of cross-organizational service-based collaboration, because attackers can create detailed profiles of service consumers, providers, and also of market places by monitoring communication endpoints. Thus, sensitive information about the underlying business processes of the communicating organizations can be inferred easily.

Sophisticated countermeasures exist for achieving the required type of anonymity, so that an attacker cannot sufficiently distinguish whether the sender and recipient of a particular message are related or not. However, even if such standard anonymity mechanisms are deployed and used correctly, attacks mounted at the edges of such networks and aiming at typical long-term business relationships are very likely to be successful.

Therefore, this paper investigated the following research question: “How does an adaptation of the Statistical Disclosure Attack regarding service compositions threaten the relationship anonymity of service consumers and providers in the Internet of Services?”

In order to answer this question, the well-known “Statistical Disclosure Attack” was extended regarding service compositions in the Internet of Services scenario. This extension was then evaluated using a simulation model of different attacker models and attack scenarios, which was implemented with the Repast Simphony toolkit.

While the results based on a uniform provider distribution look promising for the attacker’s success, such a distribution cannot necessarily be expected in the real world, i.e., the future Internet of Services. The used Zipfian distribution, whose skew parameter is inspired by the access distribution of Web pages, has the strongest impact on the attacker’s results. This leads to a clear defeat of the attacker for the observed parameters, even if more observations are made.

In addition, if the observed organization (“Alice”) uses many different compositions or mainly ones with a high composition complexity, this makes the attacker’s defeat even clearer. As an organization cannot just increase its process complexity for improved security, this could be achieved by extending the concept of “dummy traffic” with respect to using “dummy compositions” or “dummy services” therein, e.g., obfuscating real compositions with additional (irrelevant) services.

Further impact can be achieved by increasing the number of offered compositions or the batchsize of the anonymity system. However, increasing the batchsize is likely to have serious side-effects, e.g., regarding important Quality of Service parameters such as the response time of service requests.

These findings might suggest anonymity is not that much in danger in an Internet of Services with a suitable provider distribution. However, the attacker can also improve his chances of success by including further information such as the providers’ replies into his internal calculations. In addition, outside knowledge can be used as well, e.g., a bank is more likely to collaborate with other financial service providers than with providers from the logistics, pharmaceutical, or automotive sector. These aspects will be addressed in our future work, because the threat of revealing sensitive process information remains. Thus, our next steps will be to evaluate the existing model with other configuration parameters, e.g., even more observations, and to extend the attacker model regarding industry sector information and replies from service providers, e.g., continuing the work by Danezis et al. [6].

Acknowledgments

This work is partially supported by E-Finance Lab e. V., Frankfurt am Main, Germany (www.efinancelab.de) and BearingPoint Management and Technology Consultants. Furthermore, the authors would like to thank Mr. Enkh Amgalan Ganbaatar for his support regarding the simulation runs.

7. REFERENCES


