Undefined 1 (2009) 1–5
IOS Press

Trusted Tiny Things: Making Devices in Smart Cities More Transparent

Editor(s): Name Surname, University, Country
Solicited review(s): Name Surname, University, Country
Open review(s): Name Surname, University, Country

Edoardo Pignotti [a], Peter Edwards [a] and Stanislav Beran [a]
[a] *Computing Science & dot.rural Digital Economy Hub, University of Aberdeen, AB24 5UA, Aberdeen, United Kingdom*
E-mail: {e.pignotti,p.edwards,s.beran}@abdn.ac.uk

Abstract. The deployment of Internet of Things devices are increasingly commonplace in smart city environments to capture, analyse and exchange streams of information. Such devices and their associated services have the potential to generate and consume large amounts of personal data raising a host of privacy concerns. In this paper we present the *Trusted Tiny Things* system that can be used to interrogate IoT devices and present users with information about their characteristics and capabilities in a transparent manner. The system consists of a mobile application used to retrieve information about IoT devices supported by RESTful web services. In order to infer IoT device capabilities our services perform reasoning over the provenance of devices represented using a number of Semantic Web technologies. We illustrate the use of the system and evaluate it with two distinct scenarios: an NFC tag used at bus stops to provide a means to access real-time bus timetables, and a telemetry device installed into vehicles by insurance companies to track driving behaviour.

Keywords: internet of things, transparency, provenance

1. Introduction

The vision of the Internet of Things (IoT) is a dynamic global network based on standard and interoperable communication protocols where physical and virtual 'things' have identities, physical attributes, and capabilities and are seamlessly integrated into the existing internet infrastructure. The IoT is thus built upon a range of sensors and other devices [1] that together represent the 'things'. These range from passive radio tags to internet connected sensor platforms and embedded computers. Deployments of such devices are increasingly commonplace in Smart City environments to capture, analyse and exchange streams of information. For example, passive NFC (Near Field Communication) tags are currently in use by Aberdeenshire Council in the UK to provide smartphone access to timetable information for a particular bus stop. Active IoT devices include in-car black boxes [2] being introduced by insurance companies to assess the behaviour of drivers and affect their premiums. Such devices and their associated services could potentially generate and store vast amounts of data, much of it personal or other confidential data that users may not wish to share. Alternatively, they may be agreeable to data sharing, but wish to know how the data are used and by whom. Questions that a user might ask about an IoT device include: *What kind of data does the thing collect? Are the data transmitted? If so, how and to whom? For what purposes are the data used? What control do I have over any aspects related to the generation and use of this data?*. These questions were reflected in the "TRUSTe Internet of Things Privacy Index - GB Edition[1]" study where more than 80% of the 2,005 people interviewed were concerned about such issues. In-

[1] https://www.truste.com/gb-internet-of-things-index-2014/

0000-0000/09/$00.00 © 2009 – IOS Press and the authors. All rights reserved

spired by this study, we conducted a number of participatory design activities which helped shape our proposed solution; these are described in Section 3. Bartoli et al.[4] identified privacy and sensitive data management as one of the key issues to be addressed during the design of smart city systems. The authors discuss the different perspectives to be taken when assessing security or insecurity of a particular service and highlight that services should be configurable to best suit specific user needs based on sensitive data they provide. Bartoli et al. also state: “When personal data is collected by smart meters, smart phones, connected plug-in hybrid electric vehicles, and other types of ubiquitous sensors, privacy becomes all the more important." The authors also noted that one of the main challenges in this area is to protect the identity of users while providing high-quality personalised services.

The Trusted Tiny Things(T^3) project investigated how Semantic Web technologies can be used to describe the characteristics of IoT devices (e.g. manufacturer, owner, security method) and to reason about device capabilities (e.g. data sharing methods, data capture rate, data anonymisation method). As "things" become more interconnected it is also important to describe provenance information: a record of the entities (devices or services) and processes (data transmission, data analysis, decision making) involved. A formal representation of provenance has been identified as essential to support users (and machines) to better understand and trust data [5]. For example, in the car telemetry scenario, provenance could be used in order to understand what kind of data the telemetry device is collecting, what agents or services are using this data, and for what purpose.

In the remainder of this paper we introduce two transport related case studies. We then continue by discussing a trusted things framework based on Semantic Web technologies and provenance. We describe the components of the system architecture used to reason about the capabilities of IoT devices and to present this information to the user via a smartphone app. We discuss the evaluation of the system by presenting the results of two experiments based on the case-studies. We conclude by highlighting the potential impact of the project and our future plans.

2. Case-studies

The first case study relates to the deployment of passive NFC tags by Aberdeenshire Council to provide smartphone access to timetable information for bus stops. Passengers interested in obtaining real-time bus information can scan the NFC tag with a capable smartphone. The tag embeds a URL containing a unique ID identifying the bus stop. This URL is used to redirect the smartphone web browser to a third party website displaying live timetable information. This scenario raises some questions regarding the privacy of the user. For example: *Where is the web browser re-directed? Who is running the service? What information are they collecting from my smartphone? Are there any charges for the service? Would the service take contact-less payments?* (a feature normally associated with NFC technology).

In its current form, the existing timetable service does not provide the information required to answer the questions above. For example, the user is not aware that their web browser is redirected to a website (*www.rslpublic.co.uk*) and that the service is not managed by Aberdeenshire Council. There is also no explicit indication that the service is free of charge. Furthermore, the user is unaware that the service is collecting the IP address of the device used and details about the smartphone’s operating system and version.

The second case-study focuses on the use of car telemetry devices. Such devices record information about driving style and location of the vehicle using a range of sensors including GPS and accelerometers. The information captured by the sensors is then transmitted to an insurance company, typically via a 3G connection. This scenario also raises a number of questions regarding privacy. For example: *What kind of data are being recorded? When and where are the data transmitted? Who uses the data? Are the data being sent to other third-party companies? For what purpose?*

3. Participatory Design Activities

As a result of a collaborative R&D roadmapping activity between the UK’s Technology Strategy Board and the UK Research Councils conducted in 2012, a report was produced highlighting the priorities for research and innovation in the IoT. The report [9] identified several priorities in this area including the need to understand how researchers, developers and end users can become involved in co-designing IoT services, especially with respect to information interfaces. In the T^3 project we have embraced this idea by involving

users in co-design of certain aspects of our semantic models and software applications.

We conducted a number of design sessions involving a total of 77 participants with different technological backgrounds. The events involved the participants themselves determining the direction of the group discussion, through their answers to some initial questions. Each event lasted for 90 minutes and was divided into two stages. We began by exploring with participants the capabilities of IoT devices. They were presented with a number of pictures illustrating IoT devices such as an internet enabled alarm clock, a telemetry blackbox, a number of smart appliances, the NFC tags used at bus stops and an internet enabled toy. Questions were then posed such as: *What do you think are the capabilities of these devices?* and *What kind of capabilities would you want to be aware of before interacting with such devices?* Information points and other thoughts were captured on post-it notes by participants themselves and by the event facilitator. In the second stage we asked participants to design a mobile app to visualise the kinds of information identified during the earlier discussion. Participants recorded app design ideas on A3 sheets of paper, using words or images as they preferred. Information extracted from the first stage was categorised in a process where similar statements were clustered together, and subsequently formed into categories describing attitudes towards the capabilities of IoT devices. The categories identified were as follows:

- **Understanding Who** - It is important to understand who controls the devices and who has access to the data generated (people, organisations or software agents). It is also important to make explicit if the owner of the device is different from individuals and organisations that have access to the data.
- **Understanding Why** - It is important to know for what purposes personal data are used (e.g. statistics, quality of service, advertising, etc.)
- **Accessing Data** - It is important to have access to personal data generated by the devices.
- **Exercising Control** - It is important to know if it is possible to limit the data being sent by devices and if it is possible to turn the device off (and how to do this).
- **Receiving Notifications** - If the capabilities of the device change, users should be notified and additional consent should be required.

In the subsequent analysis, the key categories identify during the study were translated into requirements that informed the components of a transparency framework for IoT devices (described in Section 4) including the rules required to infer capabilities of devices. The app design ideas identified during the second stage were also analysed and used to inform development of a prototype mobile app (described in Section 5).

4. The Trusted Tiny Things Semantic Framework

The Semantic Web[2] is a vision in which today's Web will be extended with machine readable content, and where every resource will be marked-up using machine readable meta-data; a family of XML-based technologies, most notably the Resource Description Framework[3] (RDF) provide a mechanism for representing that meta-data. The Web Ontology Language[4] (OWL) is used to capture the meaning of meta-data terms and their interrelationships.

Bandara et al. [6] previously presented a semantic model for describing IoT devices. While this model is capable of describing device characteristics, it cannot describe relationships between the devices, services and their usage. Kortis et al. [7] describe an ontology that represents knowledge about "Things" in the IoT domain and the way they should interoperate. The authors have created a model describing IoT concepts by introducing ontological definitions such as *Physical Entity, Control Entity, Electronic Device, Smart Entity Cluster* and *Smart Network*. While this ontology is capable of capturing some relationships it is mostly focused on finding a common framework to allow deployment of IoT devices into the existing Internet infrastructure for service discovery and is therefore not suitable for our needs as it is too focused on describing Web service invocation methods and parameters.

The Internet of Things Architecture[5](IoTa) project is working towards a common architecture for the future Internet of Things. They have developed a conceptual model [8] to describe the IoT domain based on previous work from Serbanati et al. [10] and Haller [11]. The main aim of the IoTa Domain model is to characterise the different entities in the IoT domain (e.g. *User, Service, Device, Physical Entity, Virtual En-*

[2]http://www.w3.org/2001/sw/
[3]http://www.w3.org/RDF/
[4]http://www.w3.org/TR/owl-features/
[5]http://www.iot-a.eu

tity and *Resource*). Figure 1 illustrates the use of this model to describe a car telemetry device (*iota:Device*) that contains a number of sensors namely GPS and accelerometer. The telemetry device is attached to a car which is represented as an *iota:PhysicalEntity*. In addition, the device is associated with an insurance company service which provides tailored insurance premiums for the car owner.

In order to enhance the transparency of IoT devices we have developed a semantic framework able to characterise information such as: capabilities, security properties, ownership and provenance of devices and their use. This framework is summarised in Figure 2.

In order to reason about device capabilities we need to be able to represent the provenance of an IoT device. This might include information about the activities performed by devices, what agents (human or computational) are responsible for such activities and what data are being generated and used. In our model we describe such provenance using the W3C PROV-O[6] ontology as it is designed to be applicable to a wide range of applications and domains. PROV-O defines the following concepts: *prov:Entity* (physical, digital, conceptual); *prov:Activity* (something that occurs over a period of time and acts upon or with entities); and *prov:Agent* (something that bears some form of responsibility for an activity). Using this ontology it is possible to describe the agent (*prov:Agent*) responsible for a specific action (*prov:Activity*) taken by a device (e.g. location data collection activity). It is also possible to describe those entities (including data) that were generated by a device as a result of a *prov:Activity* (e.g. the device generated information about speed during the location data collection activity). The PROV-O model is domain agnostic and consequently the properties of entities, activities and agent are generic and not tailored to a specific application domain. While this is important in order to facilitate the exchange of provenance between systems, it would be difficult to describe provenance that is useful for a specific application using PROV-O alone. We were able to align PROV-O with the IoTa Domain Model - in order to use properties in the provenance record that are more specific to IoT devices. For example, we can use the properties of an *iota:Sensor* to describe an IoT device which is also defined as a *prov:Entity* in the provenance record.

Figure 3 shows an example of our framework being used to represent a car telemetry device. With the provenance support provided by PROV-O we are able to identify not only the high level concepts of the IoT system (i.e. Devices, Sensors, Services, Virtual Entities, Tags as identified in IoTa Domain Model), but also specific data and relationships between them. In this case there is a process (*prov:Activity*) which used the GPS sensor to calculate the current speed of the vehicle and generated another *prov:Entity* which contains specific information about the driver's speed in miles per hour (mph). This information is then used by an external process which calculates an insurance premium. Note that both the telemetry device controller and insurance service acted on behalf of the same company. Participants during our design exercises highlighted the need to provide contact information about agents (individuals or organisations) responsible for certain devices. This information is described using the Friend-of-a-Friend FOAF[7] ontology. The class foaf:Organization was defined as a subclass of *prov:Agent* so the contact details of organisations could be represented in our model.

4.1. T^3 Model and Capability Inference Rules

As discussed earlier in this paper, the use of PROV-O allows us to address some of the issues relating to transparency of IoT devices in smart environments, such as: Who is responsible for a device? What data are generated? and Where are the data transmitted? However, in PROV-O it is not possible to distinguish between personal and non-personal data, neither it is possible to describe the purpose behind the use of such data. We have therefore designed an ontology (T^3 ontology) to support two important aspects of the reasoning required to infer and detect device capabilities. The ontology provides annotations over PROV-O concepts to capture the kind of information identified as important by our user participants. These annotations include:

- The *ttt:PersonalData* class is used to identify if a *prov:Entity* represents personal data. The *ttt:description* property is used to provide a description of the data and why it is classified as personal.

[6] http://www.w3.org/TR/prov-o/

[7] http://xmlns.com/foaf/spec/

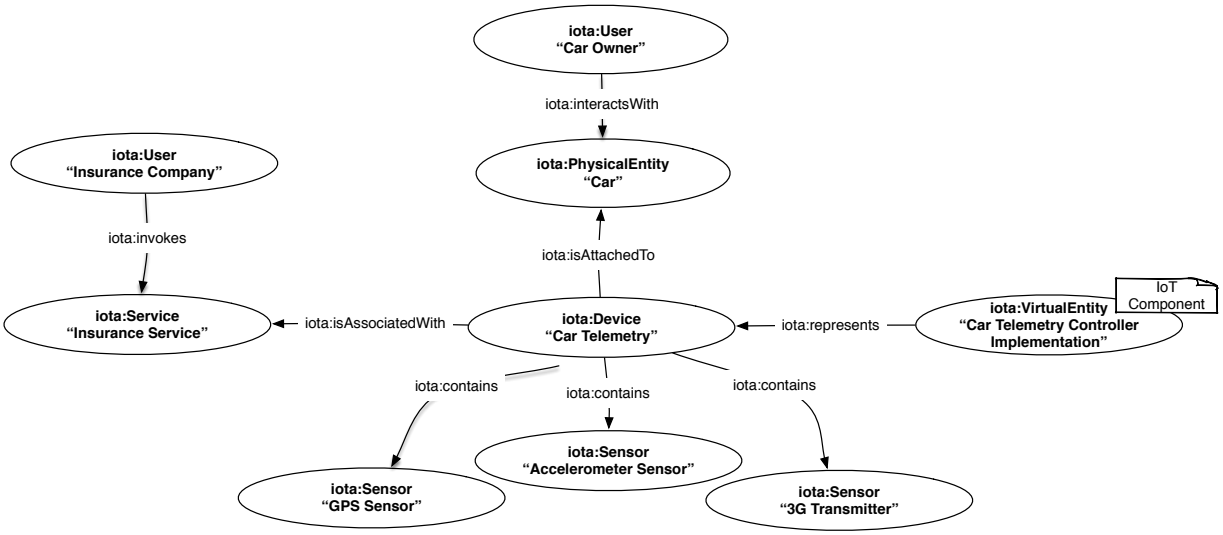

Fig. 1. Semantic model of the car telemetry scenario modelled using the IoTa [8] domain model.

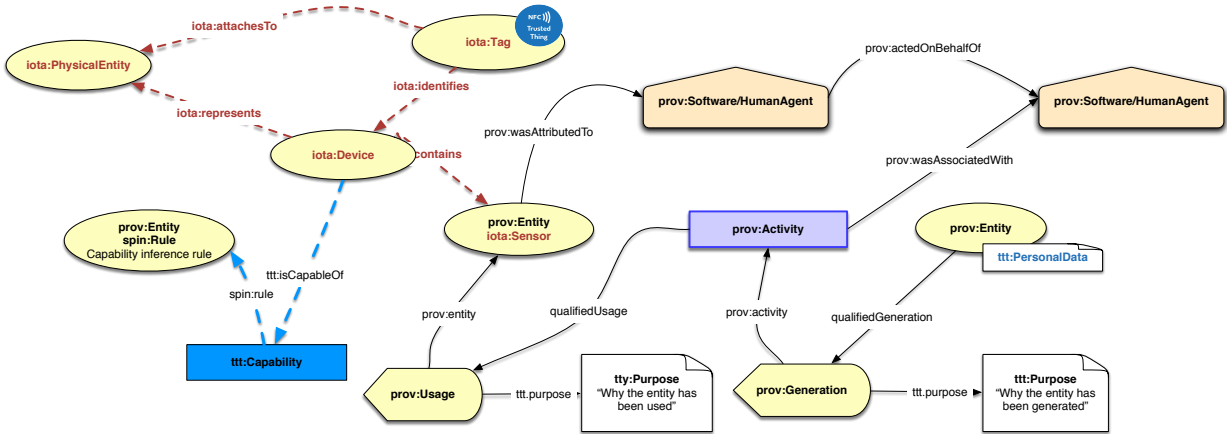

Fig. 2. An extract of the ontological framework describing the provenance of IoT devices.

- The *ttt:purpose* property is used to provide a description of why certain entities (described as personal data) are being generated or used.
- The *ttt:Capability* class defines different kinds of capabilities (e.g. *ttt:DataCollection, ttt:DataGeneration, ttt:DataSharing*) that can be associated with *iota:Devices*. These associations (described by the *ttt:isCapableOf* property) are made on the basis of a number of inference rules described later in this section.
- Properties such as *ttt:deviceDescription*, *ttt:device Name* were defined to provide additional descriptions of device characteristics relevant to our user group.

As noted above, to infer the capabilities of IoT devices using our ontological framework we can associate rules to specific classes of *ttt:Capability*. We make use of the SPIN ontology[8] to support the use of SPARQL to specify rules and logical constraints necessary to reason about capabilities. Such rules are designed to explore a PROV-O provenance graph starting from an instance of an *iota:Device* and identifying any device capability that is relevant to a specific user. To date, we have implemented five inference rules, which are closely related to personal information:

[8] http://spinrdf.org/spin.html

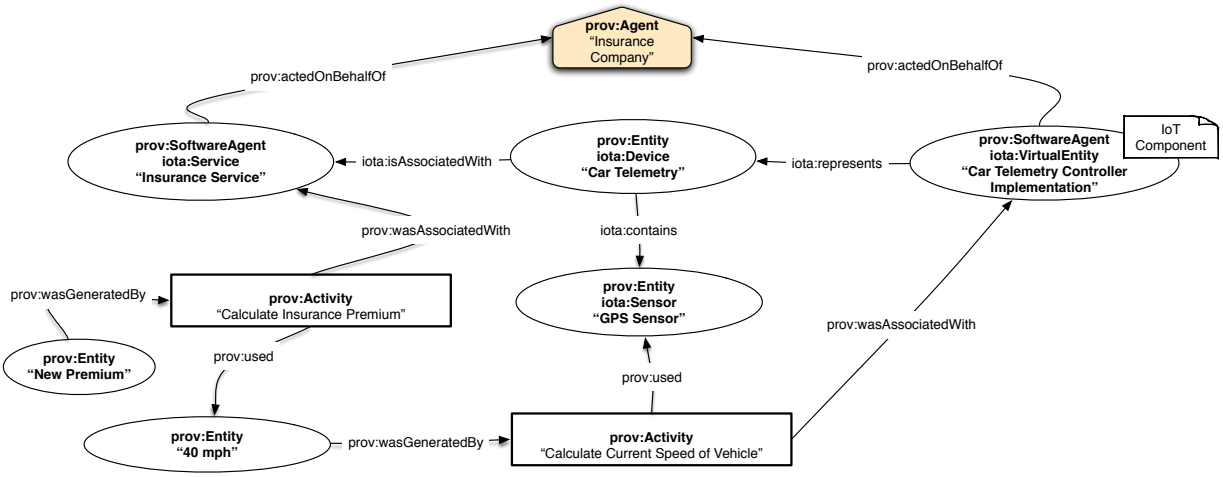

Fig. 3. Provenance extract from the car telemetry scenario.

- Rule to check if any IoT device or service is capable of charging the user.
- Rule to check if personal data are being collected.
- Rule to check if personal data are being used and if so for what purpose.
- Rule to detect if personal data are shared with third-party organisations.
- Rule to detect what data the device or service is generating (e.g. using collected data).

Implementation of some of these rules can be seen in Figure 4.

If a rule activates, it constructs new inferred facts about the device in question and forms them into capabilities (subclasses of *ttt:Capability*). If a certain capability is already attached to a device (e.g. *ttt:DataSharing*) the inferencing will avoid recreating the same one, unless there is a new purpose or actor (e.g. personal data is being shared with a new agent).

Using the provenance record our system can understand how data flows within an IoT system. This will permit identification of all the agents and processes, that played any part in creation, generation, modification and aggregation of personally identifiable data. For example, rules can determine if the inferred device capabilities are *direct* (about the device itself) or *indirect* (about associated services or devices).

5. The Trusted Tiny Things Architecture

We have developed a software infrastructure and mobile app (see Figures 5 and 6) that can be used to register, query and update IoT devices and to notify the user of any changes in the capabilities of a registered device.

5.1. System Architectural Layers

The system is composed of the following layers:

- **Storage layer** - used to store and retrieve device metadata, provenance records and information about users. We employ two different Jena TDB[9] repositories. One repository holds publicly available provenance of IoT devices (e.g. information on device manufacturer and ownership). The other repository holds a record of user preferences and the list of devices he/she agreed to use with current capabilities. We also utilise a MySql database to store system data (e.g reasoner logs).
- **Ontological layer** - contains a number of ontologies used to support the metadata in the Storage layer. These ontologies were discussed earlier in Section 4.
- **Service Layer** - defines methods for querying, updating, and synchronising data from the devices. It is also capable of notifying users of any changes in an IoT device or service (e.g. a new organisation is using information provided by an existing IoT device).
- **RESTful API Service Layer** - provides uniform access to the system from external applications such us our Trusted Tiny Things mobile app.

[9] http://jena.apache.org/documentation/tdb/

```
CONSTRUCT {
   _:b0 a :PersonalDataSharing .
   _:b0 :consumes ?data .
   _:b0 :consumer ?agentResponsibleCol .
   _:b0 :provider ?agentResponsibleGen .
   _:b0 :purpose ?purpose .
   ?this :isCapableOf _:b0 .
}
WHERE {
   ?genactivity (prov:wasAssociatedWith)* ?service .
   ?service a prov:Agent .
   ?service prov:actedOnBehalfOf ?agentResponsibleGen .
   ?agentResponsibleGen a foaf:Organization .
   ?data prov:wasGeneratedBy ?genactivity .
   ?this :owner ?agentResponsibleGen .
   ?data a :PersonalData .
   ?data :description ?dataDesc .
   ?collectingActivity (prov:wasAssociatedWith)* ?collectingService .
   ?collectingService prov:actedOnBehalfOf ?agentResponsibleCol .
   ?agentResponsibleCol a foaf:Organization .
   ?collectingActivity prov:used ?data .
   ?genData prov:wasGeneratedBy ?collectingActivity .
   ?collectingActivity prov:qualifiedUsage ?usage .
   ?usage :purpose ?purpose .
   FILTER (?agentResponsibleGen != ?agentResponsibleCol) .
   NOT EXISTS {
      ?pds a :PersonalDataSharing .
   } .
}
```

```
CONSTRUCT {
   _:b0 a :PersonalDataGeneration .
   _:b0 :generatedBy ?agent .
   _:b0 :generates ?data .
   ?this :isCapableOf _:b0 .
}
WHERE {
   ?data prov:wasGeneratedBy ?activity .
   ?data a :PersonalData .
   ?data :description ?dataDesc .
   ?activity (prov:wasAssociatedWith)* ?agentController .
   ?agentController prov:actedOnBehalfOf ?agent .
   ?agentController iota:represents ?this .
   NOT EXISTS {
      ?pdg a :PersonalDataGeneration .
      ?pdg :generates ?dt .
      ?dt :description ?dataDesc .
   } .
}
```

```
CONSTRUCT {
   _:b0 a :BillingCap .
   _:b0 :provider ?provider .
   _:b0 :consumes ?data .
   ?this :isCapableOf _:b0 .
}
WHERE {
   ?data a :BillingData .
   ?data prov:wasGeneratedBy ?activity .
   ?data :description ?dataDesc .
   ?activity prov:wasAssociatedWith ?server .
   ?server prov:actedOnBehalfOf ?provider .
   NOT EXISTS {
      ?bc a :BillingCap .
      ?bc :consumes ?dta .
      ?dta :description ?dataDesc .
   } .
}
```

Fig. 4. Personal data sharing, personal data usage and billing inference rules.

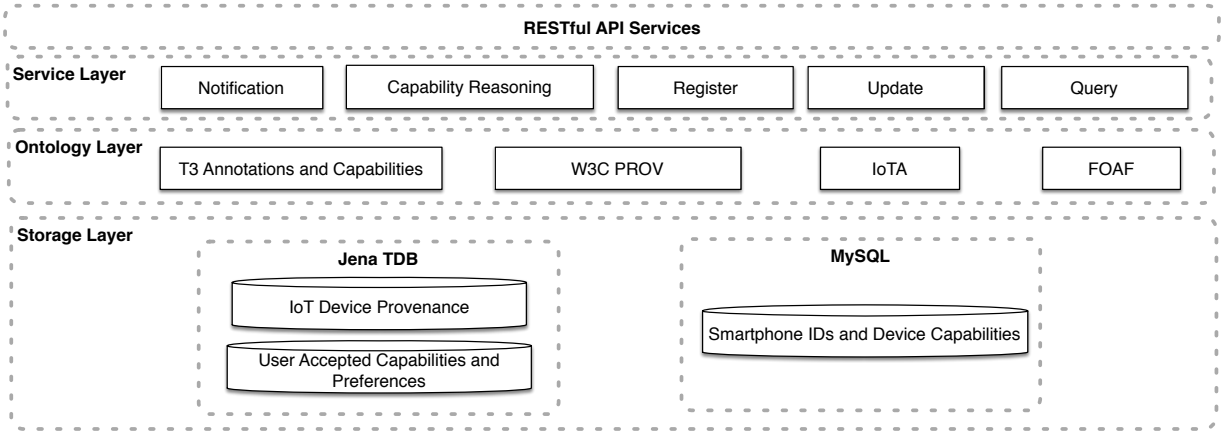

Fig. 5. Trusted Tiny Things system architecture.

5.2. *Core Services*

The system consists of five core Java EE based services:

The *Register* service is used to register new devices and associations between devices and users into the system. It provides methods for generating links between devices and the physical entities they represent. Both devices and device users are given a unique ID, which is used to identify them as part of our provenance record. For example, after registering a car telemetry device the manufacturer will be given an unique Trusted Tiny Things ID which identifies the device in our system. When a user (e.g. car driver) uses our system for the first time it will be also given a unique ID which corresponds to the ID of the smartphone used to interrogate our system.

The *Update* service provides methods for updating information about devices in the system. This provides an API that can be used to update the provenance of the device during its use. For example new activities performed by a device or any of the associated services can be recorded via this API. This service also provides a method for associating smartphone IDs (used for identifying device users) with IoT devices. Using the app a device user can obtain information about the capabilities of an IoT device. The user can then de-

cide to accept or decline such capabilities which will be recorded in the provenance record.

The *Capability Reasoning and Filtering* service is designed to reason about the provenance information associated with devices in order to infer direct or indirect capabilities. This service makes use of a rule-based reasoner implemented using the TopBraid SPIN API[10] to evaluate rules of the kind described in Section 4.1. Since the smart devices, their services and third-party smart applications can generate vast amounts of provenance metadata, it was necessary to implement an efficient mechanism to filter redundant repeated capabilities that have already been recorded. For example, consider a public security camera capable of motion detection. For each motion event activation the camera could generate provenance, which would then be sent to our system. However, in a case such as this a user typically only wishes to know that the camera has motion detection capabilities, rather than having this capability highlighted each time it detects a specific motion event.

The *Query* service is used to extract information about an IoT device based on the metadata stored in our system (including provenance and capability inferences). For example, with the bus stop case study, a user could query for the owner of the device, details of the manufacturer, what data are being collected when using the service and what capabilities are associated with the device.

The *Notification* service checks for changes in the capabilities of registered devices every hour. If the service detects such changes push notification messages are sent to the relevant users informing them of the newly detected capabilities.

The *Policy* service is used to determine if certain activities are permitted under a set of user defined policies. For example, the insurance company might want to check if information about a specific car telemetry device can be shared with the car manufacturer according to the policies defined by the car owner. Activities are expressed using the PROV model (*prov:Activity*) and used by this service as a proposition (e.g. the insurance company is proposing to send device information to the car manufacturer). Policies are also expressed in PROV as a set of undesired provenance states (e.g. the car owner does not wish that his/her data is shared to other companies). The *Policy* service reasons about the proposed activity against the existing provenance record to determine if the combination would result in a valid or undesirable provenance state. The outcome of this reasoning is returned to the client which then decides whether on not it should proceed with the activity.

5.3. Trusted Tiny Things App

The services described above are accessible from our mobile app via a RESTful API layer and JSON[11] as the data interchange format. Devices in our API are recognised by a custom URL *http://t3domain/devices/{DeviceID}* where the DeviceID represents the device identifier (e.g. the ID identifying a specific car telemetry device) encoded in a secure Trusted Thing NFC tag, which device owners should attach to their devices. The use of a custom NFC tag is to ensure that the user recognises that a device has been registered with the T^3 system and that it will be possible to query the device capabilities with our app. Using the device URL we can support different types of GET and POST actions to retrieve or create information about devices. The app listens for NFC events in a smartphone in order to determine if they represent a registered device. If so, it retrieves the device characteristics and a list of inferred capabilities based upon the past behaviour of the device. The app also uses a pictorial representation to summarise all the organizations associated with the IoT device and the type of data that are being collected. If the user is happy with the capabilities, he/she has the option to either Accept or Decline the use of the IoT device or service associated with it. Once the user has accepted the device capabilities, he/she subscribes to receive notifications about any changes/additions to device capabilities.

6. Evaluation

In this section we discuss the evaluation of the Trusted Tiny Things system through two experiments conducted using the case-studies introduced in Section 2. The first experiment involved monitoring and gathering feedback from a small group of public transport users utilising the system for a period of one week. The second experiment involved analysing the performance of the reasoning component of the system using data collected by a car telemetry device. The overall aim of the evaluation was to determine the usefulness

[10] http://topbraid.org/spin/api/

[11] http://www.json.org/

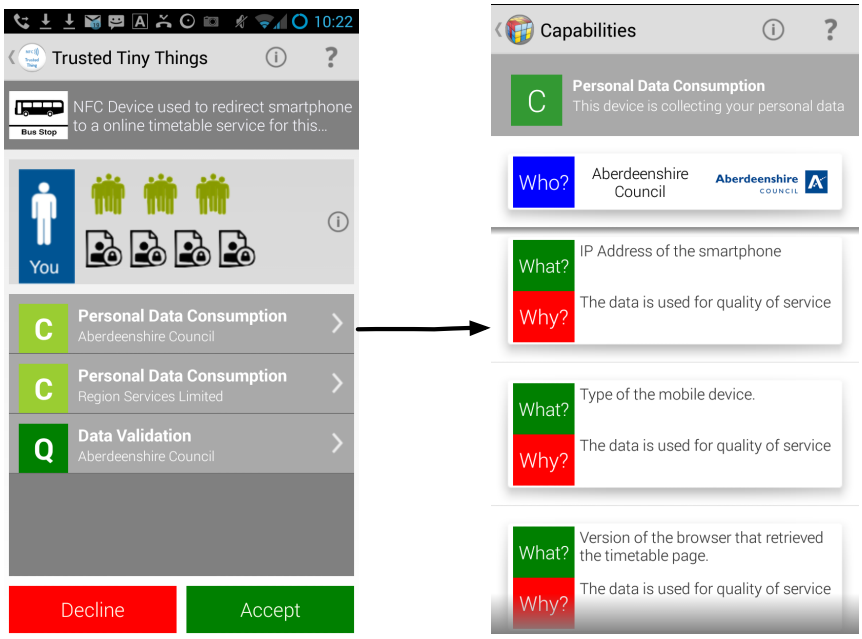

Fig. 6. Main screen (left) and capability screen (right) of the Trusted Tiny Things mobile app.

of the T^3 system and to understand the impact of provenance and semantic reasoning on the performance of an IoT system.

6.1. Experiment 1: Public Transport Users

The purpose of this experiment was to assess the usefulness of the system to potential users. More specifically, with this experiment we were aiming to answer the following questions:

Q1 Would users be interested in exploring the capabilities of IoT devices?
Q2 Would users like to have control over the data generated by IoT devices?
Q3 Would users find the application and the information presented easy to understand?

We recruited 12 public transport users based on the conditions defined below:

- The participant should own or have access to an Android mobile phone with OS version greater that 2.3.3.
- The mobile phone must be capable of reading NFC tags.
- The participant must be a regular user of public transport in the area of Aberdeen city.

Participants were asked to install the Trusted Tiny Things app from the Android Play Store and were given a brief introduction on how to use the app and how to recognise the Trusted Thing NFC tag. Following a successful installation, participants were asked to use the app for one week by scanning Trusted Things tags deployed on over 2,400 bus stops across Aberdeen. At the end of the study period, participants received an email invitation to complete an online questionnaire asking them to give feedback about their experience with the system.

During the study, and with consent from participants, we instrumented the T^3 app and system to collect usage data. Specifically, we recorded what information the users explored by logging events on the mobile interface (click, scroll, open, close and back) and the API calls made from the app to the T^3 system. We used such data in order to determine which of the ten possible items of information specific to the bus stop scenario were explored by users. These items are summarised below:

Main This is the main screen presented to the user after scanning the Trusted Thing NFC tag for the first time. An example of this screen is illustrated in Figure 6 (left). This includes a graphical representation of "you" (the user of the IoT device), what personal information will be used and by

whom. This is represented by graphical symbols (people or documents) that can be clicked on in order to obtain information. These include:

IP The IP address of the mobile phone.
Type Information about the mobile device model.
Browser The name and version of the web browser.
OS The name and version of the mobile device.
ASC Aberdeenshire Council is one of the organisations that collects the data.
NXP NXP semiconductors manufactured the device.
RSL RSL Limited collects data.

Capabilities of Aberdeenshire Council This screen is displayed when the user is interested in additional information about the personal data consumption capability associated with the bus timetable service. Specifically this information item describes what personal information Aberdeenshire Councils uses and for what purpose. An example of this screen is illustrated in Figure 6 (right).

Capabilities of RSL Describes what personal information RSL limited uses and for what purpose.

We now present some of the evidence collected from both the questionnaire instrument and the system logs, and discuss these with respect to questions **Q1**, **Q2** and **Q3** introduced at the beginning of this section. 75% of the participants were aware of what they could do with the NFC technology before they started using the app, the remainder of the participants learnt about the technology for the first time. During the experiment, the Trusted Things NFC tags at bus stops were scanned 71 times. Participants used the smartphone application a total of 118 times. One of the aims of this study was to determine if users are interested in exploring the capabilities of IoT devices (**Q1**). Results from the survey suggested that more that 70% of the participants are concerned with what personal information they share online especially if the information contains biometric or behavioural data. 57% of the participants indicated that they would be more inclined to share personal data if they knew how the data were going to be used. We can therefore deduce that for the participants in our study, understanding what information a device collects and who uses it would impact on how they share their personal information within a future IoT ecosystem. This is also consistent with the issues identified during the participatory design activities (see Section 3).

To further illustrate this point, all of the participants indicated that before using the T^3 app they were not aware of the capabilities of the services operating the NFC enabled bus timetable. While analysing the logs, we discovered that participants looked at 5 of the 10 information items available on the app. 62% of users looked at "Capabilities of Aberdeenshire Council" and "Capabilities of RSL". A summary of this data is presented in Table 1. During the first interaction with the app, the majority of the participants looked at the "Capabilities RSL" information item. Reflecting on this, we suggest that participants were not expecting a third party company to be involved in the timetable service, especially one that could collect information about their smartphone.

Users	12
Scanned Tags	71
App Usage	118
Item(Main)	9
Item(IP)	6
Item(Type)	6
Item(Browser)	6
Item(OS)	6
Item(ASC)	4
Item(NXP)	5
Item(RSL)	5
Item(Capabilities of A. C.)	7
Item(Capabilities of RSL)	8

Table 1
Summary of the mobile app usage statistics for each of the bus stop information items held in the T^3 system.

In this study we were also interested in the question of user control over the data generated by IoT devices (**Q2**). From our survey, 90% of the participants strongly believed that, as user of an IoT device, they should have some control over its behaviour. Moreover, 88% of them would like to have control over what data is collected about them and would like to be notified if there are any changes in the underlying service. One of the participants stated that "through the Trusted Tiny Things app I was able to see where my data is going and I would welcome having the opportunity to deny this and not provide the data about myself. But even if I choose not to share I would still want to get the service that the device provides."

The final aspect of the study explored if the information presented by the app was easy to understand (**Q3**). The majority of the participants (87%) stated

in the survey that they found the app easy to use and the language used to describe device capabilities was clear. Interestingly, 70% of the participants also indicated that they are now more aware of the implications that technology could have on their own privacy.

6.2. Experiment 2: Car Telemetry Device Experiment

The purpose of this second experiment was to evaluate what would be the computational cost associated with running the Trusted Tiny Things system on top of an existing IoT ecosystem. For this we created an experimental IoT system based on the car telemetry case-study described in Section 2. The ecosystem consists of a car telemetry device installed by insurance companies, a web service using information from the telemetry device to calculate a tailored insurance premium and a car-manufacturer service using data from the telemetry device to determine the current status of the vehicle. Using an official telemetry device was beyond the scope of this project. We therefore developed a mobile app (called Simbox) replicating the functionality of telemetry devices issues by insurance companies.

The methodology used for this experiment was to measure and compare the time required to run the mobile app and/or the insurance service under the following conditions:

C1 Basic Simbox app sending raw data to the insurance company service.
C2 Simbox app sending raw data to the insurance service and provenance metadata to the T^3 System.
C3 Basic insurance service consuming raw data from the IoT device.
C4 C3 + sending provenance metadata to the T^3 System.
C5 C3 + sharing data to the car manufacturer service.
C6 C5 + sending provenance metadata to the T^3 system.
C7 C6 + policy support (e.g. insurance service decides to share data with the car manufacturer service, if user consented).
C8 C6 + policy support where all users in the system refused consent to share data with the car manufacturer.

In order to test condition C1 and C2, we conducted 36 system runs and recorded the time required to generate and send raw data from the Simbox to the insurance service with and without the provenance required by the T^3 system. On average, the T^3 system increases the time the Simbox app needed to generate a datum by 10%. 24 triples per datum described provenance. This included annotations detailing the sensor used to generate personal data, e.g. accelerometer, GPS.

To investigate conditions C3-8 we conducted 50 system runs measuring the time required by the insurance service to compute a set of inputs from the car telemetry device. The outcomes of this experiment are summarised in Table 2. The T^3 service imposes a time overhead of 14% over the basic service as demonstrated by the time difference between C4 and C3. In addition, an average of 15 triples were needed to describe the data used to calculate the insurance premium, the purpose of using that data, and any additional personal data generated (Billing Entity).

Sending provenance information from both the car manufacturer and insurance services took 26% longer than the basic services (C6 - C5). Both services had a similar overhead associated with computing and transmitting provenance to the T^3 system.

Reasoning about provenance and policies as part of the insurance company service increases the time taken by up to 40%. The policy in this experiment checked if the user consented to their personal data being shared with the vehicle manufacturer, and if not, prevented the insurance service from sending such data. We have also tested the scenario in which every insurance holder refuses consent to share data with the car manufacturer (C8) and observed a 14% time increase over the existing insurance company service.

In conclusion, we identified that in order to run the T^3 system (with no policy support) in an IoT application it is necessary to account for an overhead of between 10 and 15% compared to the IoT application alone. This has been tested in a mobile app and supporting web services. The cost is much greater (up to 40%) if policies are used. This was expected as the insurance system delegates the decision to invoke services to the T^3 system which uses semantic reasoning to determine if the data can be shared between services. However, this cost could be reduced as policies might have an effect on the number of third-party service invocations (e.g. when a user does now allow data to be shared with the car manufacturer).

7. Conclusions

The Trusted Tiny Things project investigated how Semantic Web technologies can be used to manage information about IoT devices so that their capabilities

	C3	C4	C5	C6	C7	C8
Time(ms)	248	284	280	354	392	321
Increase(%)		+14% C3		+ 26% C5	+ 40% C5	+ 14% C5

Table 2

Table summarising the time required to run the insurance premium calculation service under experimental condition C3-8.

are transparent to users. Our motivation was to assist users to make informed decisions when using an IoT device.

During the participatory design activities we learnt what users want to know before interacting with IoT devices. This is summarised in five high-level requirements: understanding who controls the device and has access to the data; understanding why the data are used; being able to access personal data; being able to exercise control over the device; and receiving notifications if the capabilities of a device change. We argue that in order to create a "transparent" IoT ecosystem it is important to consider how to capture provenance, how to monitor activities within the ecosystem and how to control the behaviour of devices so these five requirements can be fulfilled.

To demonstrate how such an ecosystem can be created we have designed a semantic software infrastructure and a secure Trusted Thing NFC tag that can be attached to any physical IoT device in order to identify it within our system. We have built an app and supporting services that can provide users with information about the capabilities of IoT devices (e.g. what personal data are used, by whom, and for what purpose). We have evaluated our solution with real users based on the bus stop scenario - with the T^3 NFC tag deployed across 2,400 bus stops in the Aberdeen and Aberdeenshire region. We have also assessed the computational overhead of the T^3 solution, using a scenario based on car telemetry devices used by insurance companies.

While the Trusted Tiny Things infrastructure is designed to provide greater transparency about the capabilities of IoT devices, there are several limitations to our approach. One of these is being able to ensure that device provenance and service provenance used to derive capabilities is truthful. We provide a set of rules in the form of guidelines (*http://t3.abdn.ac.uk/guidelines*) for registering and managing information about IoT devices in the Trusted Tiny Things system. However, such rules will have to be enforced by a trusted authority in order to guarantee the reliability of the provenance represented in our system. Understanding the role of this authority in policing such a system will require further research.

In the future we are interested in exploring how policy-based reasoning will scale in smart city environments where potentially hundreds of devices could have access to information about individual citizens. It is important that a user is able to specify high-level privacy preferences, and that all the surrounding devices are able to enact such preferences by providing or denying access and notifying users of violations.

8. Acknowledgements

The research discussed was is supported by the Research Councils UK Digital Economy IT as a Utility Network+ (EP/K003569/1) and the dot.rural Digital Economy Hub (EP/G066051/1).

References

[1] Gusmeroli, S., Haller, S., Harrison, M., Kalaboukas, K., Tomasella, M., Vermesan, O., Vogt, H., Wouters, K.: Vision and Challenges for Realising the Internet of Things. In Friess, P., Guillemin, P., Sundmaeker, H., Woelfflé, S. (Eds.) European Commission (2010)

[2] Hossain, E., Chow, G., Leung, V.C.M., McLeod, R.D., Mišić, J., Wong, V.W.S., Yang, O.: Vehicular Telematics over Heterogeneous Wireless Networks: A Survey. Comput. Commun. **33**(7) (May 2010) 775–793

[3] Pignotti, E., Beran, S., Edwards, P.: What Does This Device Do? In: The First International Conference on IoT in Urban Space, October 27–28, 2014 Rome, Italy (2014) 56–61

[4] Bartoli, A., Hernández-Serrano, J., Soriano, M., Dohler, M., Kountouris, A., Barthel, D.: Security and Privacy in Your Smart City. In: Proceedings of the Barcelona Smart Cities Congress. (2011)

[5] Moreau, L.: The Foundations for Provenance on the Web. Found. Trends Web Sci. **2**(2-3) (February 2010) 99–241

[6] Bandara, A., Payne, T.R., de Roure, D., Clemo, G.: An Ontological Framework for Semantic Description of Devices. McIlraith, S.A., Plexousakis, D., van Harmelen, F. (Eds.) ISWC 2004. LNCS, Vol. 3298, Springer, Berlin Heidelberg (2004)

[7] Kotis, K., Katasonov, A.: An IoT-Ontology for the Representation of Interconnected, Clustered and Aligned Smart Entities. Technical Report, VTT Technical Research Center, Finland VTT Technical Research Center, Finland (2012)

[8] Bassi, A., Bauer, M., Fiedler, M., Kramp, T., van Kranenburg, R., Lange, S., Meissner, S.: Enabling Things to Talk: Designing IoT Solutions with the IoT Architectural Reference Model. Springer, Berlin Heidelberg (2013)

[9] Tafazolli, R., Upstill, C., Aghvarni, H., Cooper, R., Dutton, W.: A Roadmap for Interdisciplinary Research on the Internet of Things. Technical Report, Swindon, GB, Technology Strategy Board (2012)

[10] Serbanati, A., Medaglia, C.M., Ceipidor, U.B.: Building Blocks of the Internet of Things: State of the Art and Beyond. Deploying RFID-Challenges, Solutions, and Open Issues, Dr. Cristina Turcu (Ed.), InTech (2011)

[11] Haller, S.: The Things in the Internet of Things. In: Internet of Things Conference, Tokyo, Japan (November 2010)