Proc. 14th Int. Conf. on Global Research and Education, Inter-Academia 2015
JJAP Conf. Proc. **4** (2016) 011603

Towards Advanced Networking and M-services with Enhanced Information Security and Integrated Support for Big Data Analytics

Patrick C. K. Hung[1], Kamen Kanev[1,2], Yasuto Shirai[2], Katsuhiko Yuura[2], Masakatsu Nishigaki[2], Hiroshi Mineno[2], and Valerie Wilkinson[2]

[1] *University of Ontario Institute of Technology, Canada*
[2] *Shizuoka University, Japan*

E-mail: patrick.hung@uoit.ca

(Received September 25, 2015)

The maturing mobile communications infrastructure leads to a wider adoption of mobile services (m-services) both by the community and the industry. Consequently, effective and timely support of m-services computing security and privacy models become very important to the commercial, financial and logistics sectors worldwide. Within the general scope of business computing our work puts the research focus on advanced networking and m-services with enhanced information security and integrated support for big data analytics and knowledge creation. The pervasive nature of m-services has given rise to an emergent, data-focused economy with unprecedented research opportunities for big data analytics that we address in our work.

1. Introduction

With recent advances in mobile technologies and infrastructure, there is an increasing demand for ubiquitous and pervasive access to advanced networked services. Nowadays each and every mobile device is in fact running a range of such services that support different functionalities and accommodate various user needs. These services, generally known as mobile services (m-services), extend support from Web browsers on personal computers to mobile devices, such as smart phones and tablet computers, over the Internet to cloud computing and big data analytics. Pervasive computing and infrastructures increase the need for sharing and coordinating the use of m-services for different business processes in a loosely coupled execution environment. In principle, an m-service refers to an autonomous unit of application that provides either some e-business functionality or information to a service consumer at anytime and anywhere through wireless and Web technologies. M-services can thus be seen as the integration of wireless and Web services technologies in the context of services computing.

A Web service is a software system that supports interoperable application-to-application interaction over the Web. Web services are fundamentally based on a set of eXtensible Markup Language (XML) standards, such as Web Services Description Language (WSDL), Simple Object Access Protocol (SOAP), and Universal Description, Discovery and Integration (UDDI). Each service makes its functionality available through a well-defined or standardized XML-format Application Programming Interface (API). The result of this approach is the Service-Oriented Architecture (SOA), essentially a type of distributed system architecture as defined by W3C [1]. One of the major goals of Web services is to facilitate their composition to form more complex services such as a workflow. To this purpose, many emerging languages, such as Web Service Business Process Execution Language (WSBPEL) and Business Process Model Language (BPML), have been proposed to coordinate Web services into a workflow. A workflow is a computer supported business process and business processes play key roles in enabling business application integration and collaboration across multiple organizations. To stay competitive, the enterprise infrastructure must

provide capability for dynamic discovery of service providers and enable federated security mechanisms, solution monitoring and management for supporting business process.
Big data is the term for a collection of large and complex datasets from different sources that is difficult to process using traditional data management and processing applications. Many businesses nowadays are increasingly interested in utilizing big data technologies and conducting big data analytics for supporting their business intelligence. Big data analytics is a process of examining diverse, large-scale data sets to uncover patterns, unknown correlations and other useful information. Big-data analytics employ software tools from advanced analytics disciplines such as data mining and predictive analytics, e.g. Hadoop, MapReduce, and others. In this context the security and privacy enforcement model for m-services computing becomes an important and challenging topic in the research area of services computing that we explore.

2. Related Work

There are XML languages proposed for describing security assertions in Web services, but not in m-services. These XML languages restrict access to Web services to authorized parties only and protect the integrity and confidentiality of messages exchanged in a loosely coupled execution environment. Referring to the SOA, the research area of Web services security is also challenging as it involves many disciplines, from authentication/encryption to access control management/security policies. In the recent Web services research, there is increasing demand for and discussions about privacy technologies for supporting different business applications. For example, WS-Policy describes the business policies to be enforced with intermediaries and endpoints. The business policies contain certain requirements such as security tokens, supported encryption algorithms and privacy rules. The WS-Policy is represented by a policy expression, that is, an XML Infoset representation of one or more policy statements. The WS-Policy includes a set of general messaging-related assertions defined in WS-PolicyAssertions and a set of security policy assertions related to supporting the WS-Security specification defined in WS-SecurityPolicy. However, the current WS-Policy specification does not discuss the privacy rules in detail. Even though WS-Privacy is mentioned in terms of describing a model for defining subject privacy preferences and organizational privacy practice statements, WS-Privacy has not been developed yet.
Next, the Enterprise Privacy Authorization Language (EPAL) technical specification is used to formalize privacy authorization for actual enforcement within an intra- or inter- enterprise for business-to-business privacy control. EPAL services themselves are exchanging privacy policies and making privacy authorization decisions. In particular, EPAL concentrates on the privacy authorization by abstracting data models and user-authentication from all deployment details. On the other hand, the eXtensible Access Control Markup Language (XACML) is a general- purpose access control policy language used to describe policy and access control decision request/response [2]. The XACML has been widely deployed and there are several implementations of XACML in various programming languages available. Although XACML is designed to support both centralized and decentralized policy management in comparison to EPAL, neither EPAL nor XACML framework can handle the privacy enforcement in a mobile environment. Recently, the Internet Engineering Task Force (IETF) proposed a privacy threat model for mobile and multi-homed nodes, however their research focuses are on the threats and possible attacks against privacy in the Media Access Control (MAC) and Internet Protocol (IP) layers, not in the application layer. This indicates the missing component of integrating privacy-enhancing technologies into security mechanisms.
The immense popularity of mobile devices can be explained by their personal, portable, and pervasive nature. These characteristics create a unique platform for services, particularly those based on context data. Mobile services can be context-aware, gathering context information from the mobile device, and providing relevant personalized services based on the context. To gather context information, a context-aware service can either listen for events sent by a context provider, or query

the context provider. Gu et al. [3] propose a middleware for building context-aware mobile services, using a Service Locating Service to allow entities to locate different context providers. However, this model does not consider privacy preferences of the user.

When software is running on any device, the application will need to communicate with other services whether they are internal or external (over a network). The Device Profile for Web Services (DPWS) [4] follows the SOA framework for automatic device and service discovery for networked embedded devices. DPWS offers a standardized device representation of services on a network and this allows for access to a set of built-in services such as secure accessing of metadata and exchange services by utilizing WS protocols. In other words, DPWS defines a minimal set of implementation constraints to enable secure Web service messaging, discovery, description, control, and events on resource-constrained endpoints [4]. The specification permits the definition of services for mobile devices considering the peer-to-peer direct communication between them that combine several devices as SOA. DPWS allows sending secure messages to and from services, dynamically discovering a service, describing a service, subscribing to, and receiving events from a service.

In Web Services terms, a profile is a set of guidelines for how to use Web services technologies for a given purpose or application. Web services standards allow implementers to choose from a variety of message representations, text encodings, transport protocols, and other options, some of which are not interoperable. By constraining these decisions, profiles ensure that conforming implementations will work well together. DPWS is a profile developed by Microsoft and others for communication with and among networked devices and peripherals. The DPWS library for the .NET Micro Framework is not a full Web services implementation but a lightweight subset with only the functionality needed to support DPWS on a device [5]. DPWS was built on the foundation of existing web services (WS) and as such uses many common specifications such as XML, SOAP, WS-*, WSDL and Message Transmission Optimization Mechanism (MTOM). DPWS defines two main types of services that are run by devices: hosting services, and hosted services [5]. Devices can be DPWS clients (invoking hosted services on devices), servers (providing hosting services), or both. DPWS for the .NET Micro Framework supports devices in either role or both simultaneously. Hosting services allow other devices to use, subscribe, and obtain metadata of the given services. DPWS defines the extensions required for using services in mobile devices, taking in account their specific constraints. A DPWS enabled device has access to provided functionality such as: the discovery of other, utilizing WSDL to describe a Web service, service subscription, and secure sending of messages, given that the other device also utilizes DPWS.

The Web Services for Devices (WS4D) [6] framework is an extension of DPWS to bring SOA and Web services technology to industrial automation, home entertainment, automotive systems and telecommunication systems. There have been ongoing initiatives to connect internet technologies and web services to resource-constrained devices in ad-hoc networks while conserving interoperability. WS4D provides technologies for easy setup and management of network-connected devices in distributed embedded systems [7]. Araujo and Siqueira [8] used WS4D to implement a DPWS Device Service Bus (DSB), establishing a Device Tunnel to deal with virtual devices and services. Pohlsen et al. [9] present a plug-and-play architecture for connecting medical devices through DPWS, using WS-Discovery protocol. Unlike traditional Web service architectures, the authors propose using a WS-Discovery proxy server rather than a UDDI server, to better meet the requirements of resource constrained devices. Further, the work uses SOAP-over-UDP (User Datagram Protocol) for multicast messaging, as included in DPWS. El Kaed et al. [10] present an implementation to interoperably connect Universal Plug and Play (UPnP) and DPWS smart home devices such as a TV, printer, and light bulb. DPWS does not support fine-grained security requirements, direct authentication between devices without a third party, and does not propose a comprehensive authorization concept. All of these works present the foundation technologies for this research work. To the best of our knowledge, there is no unified framework for enforcement for location privacy in mobile services for toy computing.

3. M-services Security and Big Data

M-services architectures are built on an insecure, unmonitored and shared environment, which is open to events such as security threats. As is the case in many other applications, the location based information processed in m-services might be sensitive so it is important to protect it from security threats such as disclosure to unauthorized parties. The research challenge is much bigger than many other disciplines with these five properties:

(1) Mobility: M-services should only be limited by the range, which is set by the logic of the business process.
(2) Peer-to-Peer: M-services have to interact and communicate directly, without using a central server, with each other.
(3) Collocation: All logical interactions between m-services may have to result in a physical interaction between location-based users.
(4) Collaboration: Collocated m-services need to be willing to collaborate.
(5) Transitory Community: M-services/users may join and withdraw from the interactions at any time, making it an ever changing map.

Further the pervasive nature of m-services has given rise to an emergent, data-focused economy stemming from many aspects of business process. The richness and vastness of these data are creating unprecedented research opportunities for big data analytics. Big data is the term for a collection of large and complex datasets from different sources that is difficult to process using traditional data management and processing applications such as in m-services environment. In these datasets, some information must be kept private and/or secret from others. On the other hand, some data has to be released for acquainting information or big data analytical services. Privacy is described by the ability to have control over the collection, storage, access, communication, manipulation and disposition of data. Some refer to privacy as the right for individuals to determine for themselves when, how, and to what extent information about them is communicated to others. Many countries including Japan set out legislations for how organizations may collect, use, or disclose personal information in the course of commercial activities. Failing to comply with these legislations, in their respective countries, may lead to civil and/or criminal penalties. In addition to that, organizations may suffer loss of reputation and goodwill when the non-compliance of legislation is publicized. The framework should provide user with the complete personalization and control facilities. Users can configure the functionalities and notification mechanism according to their own preferences. This is achieved through a refined abstract model for policy enforcement derived from the one defined by the IETF and incorporating the following components:

- Policy Decision Points (PDP): The point where policy decisions are made;
- Policy Enforcement Points (PEP): The point where the policy decisions are actually enforced;
- Resources: Something of value in a network infrastructure to which rules or policy criteria are first applied, before access is granted. This can be referred as the objects in the privacy access control model;
- Policies: The combination of rules and services where rules define the criteria for resource access and usage. This can be referred to as the privacy rules in the enforcement model.

In general, the PEP is implemented at the service requester side while the PDP is implemented at the service provider side. The application and the PEP represent a subject that makes a request to access an object (resource). The enforcement model can use this abstract model as the base of technical framework. However, this abstract model does not consider any privacy entities at all and this model is not designed for supporting m-services. Referring to [1], there are five fundamental requirements (AC020) for enabling privacy protection for the consumer (player) of a web service across multiple domains and services as summarized in Table I.

Table I. Fundamental requirements for privacy protection across multiple domains and services.

Requirement ID	Requirement description
AR020.1	The WSA must enable privacy policy statements to be expressed about Web services.
AR020.2	Advertised web service privacy policies must be expressed in the Platform for Privacy Preferences Project (P3P).
AR020.3	The WSA must enable a consumer to access a web service's advertised privacy policy statement.
AR020.5	The WSA must enable delegation and propagation of privacy policy.
AR020.6	Web services must not be precluded from supporting interactions where one or more parties of the interaction are anonymous.

For the development of such framework on the illustrative scenario, the SOA for Devices (SOA4D) and WS for Devices (WS4D) implementation of DPWS will be used. Referring to services computing there is increasing demand for, and discussions about privacy technologies for supporting m-services. In our research, WS-Policy is used to describe the privacy policies for location based information to be enforced with intermediaries and endpoints. The WS-Policy is represented by a policy expression, that is, an XML Infoset representation of one or more policy statements. The WS-Policy includes a set of general messaging-related assertions defined in WS-PolicyAssertions and a set of policy assertions related to supporting the WS-Security specification defined in WS-SecurityPolicy. However, the current WS-Policy specification does not discuss the privacy rules in detail and needs thus further refinement. One of the objectives of our current work, therefore, is to address this issue in more detail.

All this, as well as additional security enhancements, have to be properly integrated in big data technologies that propel the growing business computing needs of rapidly developing organizations. Enhanced security Big Data analytics is addressed through investigation and security audits of a variety of software tools from advanced analytics disciplines such as data mining, predictive analytics, and machine learning that are relevant to business computing m-service. This research work is especially interested in the management of m-services supported business processes and their seamless integration with business-to-business or backend processes in the BYOD paradigm with big data. In this respect we support three different categories of views namely user interface, process, and data views. User interface views allow different presentations of inputs and outputs. Data views summarize data over limited bandwidth and map data form different sources into unified formats for convenient processing. A process view is a structurally correct subset of a process definition derived from that of an original process in an enterprise. Process views support service adaptation, enabling designers to systematically elicit requirements for more concise versions or modified procedures by considering process and platform capabilities. Process views also serve as the key mechanism for integrating user interface views and data views. These views are customized to cater for different technologies, protocols, platforms, situations and individual customers' preferences. The adaptation of process views for the provision of m-services can be driven by a mechanism that matches the capabilities required by process activities against the features supported by the Bring Your Own Device (BYOD) paradigm with big data support. It is likely that some process activities exercise certain security and privacy requirements, such as the need for authentication and authorization. As mentioned, one of the important issues in the discovery process is for m-services providers and requestors to negotiate and find an integrative solution that is optimal for both sides. Thus, a more sophisticated model with negotiation features is required for this emerging scenario where a policy negotiation point (PNP) is introduced between the PEP and PDP in the privacy access control policy management architecture. In order to build a negotiation model for m-services, we take advantage of the following two new components into the abstract model for policy enforcement defined by the IETF:

(1) Policy Administration point (PAP): The point at which policies are created, modified and

stored. Policy is the combination of rules and services where rules define the criteria for resource access and usage.

(2) Policy Information Point (PIP): The point that acts as a source of attribute values. Attribute is the characteristic of a subject, resource, action or environment that may be referenced in a predicate or target.

4. Conclusions

Our work is especially interested in the management of m-services supported business processes and their seamless integration with business-to-business or backend processes in the BYOD paradigm with big data. The work focuses on a theoretical model with a technical framework for enforcing and managing a security policy into services requestor at the mobile devices from the services provider perspective to support business processes in the context of m-services computing from two major perspectives: communication and business policy. However, despite the advancements in this research topic, complicated technical issues (e.g., security and privacy enforcement) and organizational challenges (e.g., business process integration and management, negotiation and agreement) remain to be solved in m-services environments. Our work in progress is anticipated to generate further valuable research outcome including student theses, models, protocols, implementations and evaluations, warranting subsequent submissions to international conferences and journals.

Acknowledgment

This work was supported in part by funding for a Cooperative Research Project at Research Institute of Electronics, Shizuoka University and KAKENHI Grant 25560109.

References

[1] Web Services Architecture Requirements, W3C Working Draft, 14 November 2002. URL: http://www.w3.org/TR/2002/WD-wsa-reqs-20021114, Retrieved September 2015.

[2] OASIS eXtensible Access Control Markup Language (XACML) Version 1.0, OASIS Standard, 18 February 2003. URL: https://www.oasis-open.org/committees/download.php/2406/oasis-xacml-1.0.pdf, Retrieved September 2015.

[3] T. Gu, H. K. Pung, D. Q. Zhang, A Middleware for Building Context-Aware Mobile Services, In Proceedings of the 59th IEEE Vehicular Technology Conference, May 17-19, 2004, pp.2656-2660. DOI: 10.1109/VETECS.2004.1391402.

[4] OASIS Devices Profile for Web Services (DPSW) Version 1.1, OASIS Standard, July 1, 2009. URL: http://docs.oasis-open.org/ws-dd/dpws/1.1/os/wsdd-dpws-1.1-spec-os.pdf, Retrieved September 2015.

[5] Microsoft, Introducing DPSW, URL: https://msdn.microsoft.com/en-us/library/dd170125.aspx, Retrieved September 2015.

[6] R. Want, An Introduction to RFID Technology, IEEE Pervasive Computing, Vol. 5, No. 1, 2006.

[7] WS4D: Web Services for Devices, URL: http://ws4d.e-technik.uni-rostock.de/about/, Retrieved September 2015.

[8] G. M. Araujo, F. Siqueira, The Device Service Bus: A Solution for Embedded Device Integration through Web Services, in Proceedings of the 2009 ACM Symposium on Applied Computing SAC'09, Waikiki Beach, Honolulu, Hawaii, USA, March 9-12, 2009, pp.185-189.

[9] S. Pohlsen, S. Schlichting, M. Strahle, F. Franz, C. Werner, A Concept for a Medical Device Plug-and-Play Architecture based on Web Services, ACM SIGBED Review - Special Issue on the 2nd Joint Workshop on High Confidence Medical Devices, Software, and Systems (HCMDSS) and Medical Device Plug-and-Play (MD PnP) Interoperability, Vol. 6, No. 2, Article No.6, July 2009. DOI: 10.1145/1859823.1859829.

[10] C. El Kaed, Y. Denneulin, F. G. Ottogalli, Dynamic Service Adaptation for Plug and Play Device Interoperability, in Proceedings of the 7th Int. Conf. on Network and Services Management (CNSM), Paris, France, October 24-28, 2011, pp. 1-9.