Security-relevance of semantic patterns in cross-organisational business processes using WS-BPEL

K.P.Fischer[1,2,3], U.Bleimann[1], W.Fuhrmann[1] and S.M.Furnell[2,4]

[1] Aida Institute of Applied Informatics, University of Applied Sciences Darmstadt, Germany
[2] Network Research Group, University of Plymouth, Plymouth, United Kingdom
[3] Digamma Communications Consulting GmbH, Darmstadt, Germany
[4]School of Computer and Information Science, Edith Cowan University, Perth, Australia
e-mail: K.P.Fischer@digamma.de

Abstract

This paper gives an overview of the research project considering security aspects in the context of business process management. In particular, security issues arising when scripts written in the standardized scripting language WS-BPEL (formerly: BPEL4WS or BPEL for short) implementing cross-organisational business processes on top of Web services are deployed across security domain boundaries, are being investigated. It analyses the security-relevant semantics of this scripting language in order to facilitate checking for compliance with security policies effective at the domain of execution.

Keywords

Security Policy, Policy Enforcement, Cross-Organisational Business Process (CBP), Semantic Analysis, Web Services, Web Services Business Process Execution Language (WS-BPEL, BPEL)

1. Introduction

Web services are currently considered a broadly adopted approach for the realization of a service oriented architecture (SOA) used in service oriented computing (SOC) (Curbera *et al.,* 2003; Foster and Tuecke, 2005; Papazoglou and Georgakopoulos 2003). Web services, and the composition or orchestration of them, play a central role in current approaches to service oriented computing (Berardi *et al.,* 2003). Service orientation is also expected to play an important role in grid computing, where the provisioning of computing resources within a conceptual huge network of collaborating computers and devices can also be fostered by services (so called grid services in this context) (Tuecke *et al.,* 2003;).

In service oriented approaches using Web services, a layered architecture for composing new services from existing services or for defining and executing processes based on existing services has emerged (Medjahed *et al.,* 2003). The request for fast adaptation of enhanced services and processes to changing requirements as well as the request to avoid dependency on certain platforms (vendor lock-in) lead to the specification of platform independent, standardized process definition languages for the definition of enhanced Web services or business processes in the top layer of this architecture. For the definition of Web services, Web Services Description Language (WSDL) (Christensen *et al.,* 2001) has been established by the World Wide Web Consortium (W3C) as a single standard broadly accepted for the definition of Web services. However, for business process definition languages (BPDLs) several approaches to standardization have been taken by different vendor groups and standardization organisations, leading to a plurality of standards:

- Web Services Business Process Execution Language (WS-BPEL), formerly known as Business Process Execution Language for Web Services (BPEL4WS or BPEL for short) (Arkin *et al.,* 2004),
- Business Process Modelling Language (BPML) (Arkin, 2002),
- XML Process Definition Language (XPDL) (Workflow Management Coalition, 2002),
- Web Services Choreography Interface (WSCI) (Arkin *et al.,* 2002), and
- ebXML Business Process Specification Schema (Malu *et al.,* 2002).

Though the existence of several parallel standards aiming at the same goal detracts from the very purpose of standardization, the different standards at least have some obvious commonalities. Research comparing different BPDLs has shown that these languages are comparable with respect to their semantic expressiveness and are, at least to a certain extent, convertible to each other (cf. section 2). Given the fundamental similarity of the different languages used for business process specification, without loss of generality we will concentrate our research on one particular representative, namely WS-BPEL. Since WS-BPEL has been submitted to the Organization for the Advancement of Structured Information Standards (OASIS), supposed to be soon released as an OASIS standard, and is supported by prominent vendors like IBM, BEA, Microsoft, SAP, and Siebel, BPEL is expected to emerge as the dominant standard for business process definition (Wang *et al.,* 2004). For the remainder of this paper we will use BPEL as a short-hand for WS-BPEL.

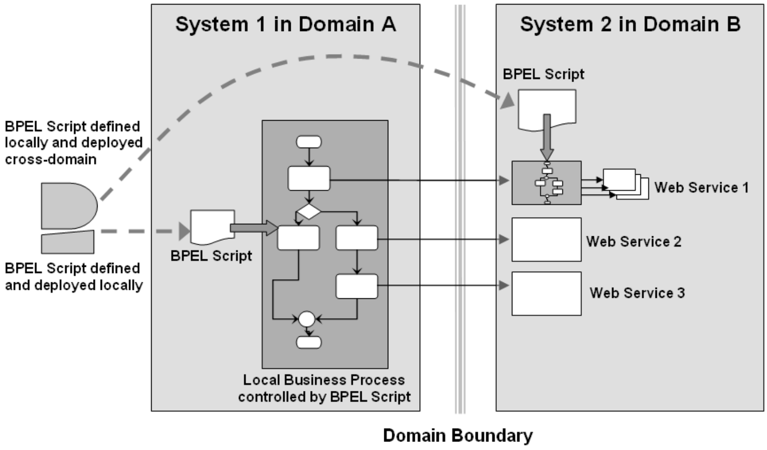
System 1 in Domain A
System 2 in Domain B
BPEL Script
BPEL Script defined locally and deployed cross-domain
BPEL Script
BPEL Script defined and deployed locally
Web Service 1
Web Service 2
Web Service 3
Local Business Process controlled by BPEL Script
Domain Boundary

Figure 1 - Distributed Development and Execution of Business Processes using BPEL

With the advent of Web services and business processes being specified in a standardized and platform-independent manner, BPDLs were considered an instrument for the definition of cross-organisational business processes (CBPs) (Lippe *et al.,* 2005), thereby supporting the concept of virtual enterprises (Coetzee and Eloff, 2003). An aspect of CBPs, that has not yet been addressed explicitly in research, is the distributed definition of a business process at one site and deployment and execution of this process at another site, for instance, being located in different organisations. Employing standardized BPDLs will make this approach feasible. However, use of this capability introduces new security issues that are not relevant in Web services as such.

Figure 1 illustrates an exemplary scenario for distributed development and execution of a BPEL script in two different domains A and B. The two domains are considered to belong to two different organisations. Each of the systems depicted in Figure 1 is capable of running BPEL-defined processes. Since a business process defined by a BPEL script may offer services to its environment, it can itself be considered a Web service. Therefore, in this example one of the Web services used by the business process in system 1 is realized as a business process controlled by a BPEL script. For this scenario it is assumed, that for whatever reasons this BPEL script is defined in domain A and deployed across the domain boundary to be executed in system 2 in domain B. Though this scenario would be technically feasible, given both systems are providing a BPEL-enabled platform, security aspects involved in this cross-domain scenario for defining and running a business process may prevent this scenario from being applied in a real-world cross-organisational environment. These security aspects will be discussed further in the following section.

2. Security Issues in Definition and Deployment of Cross-Organisational Business Processes

As security already is an important issue in distributed applications in general, this topic is also of significant importance for the application of BPDLs. Security of Web services is well studied and several approaches for access control to Web services exist (e.g. Abendroth and Jensen, 2003, Dimmock *et al.,* 2004). Role-based access control (RBAC) (Ferraiolo *et al.,* 2001) is the widely used concept for dealing with security aspects in this field.

While access control related aspects are predominant with Web services they are, of course, also an issue with BPDLs. In related work, Koshutanski and Massacci (2003) address access control issues of business processes defined by BPEL scripts, in particular the problem of providing the required evidence of possessing the proper access privileges at the right time to the right place during execution of a business process.

However, novel security aspects arise from the distributed definition and execution of cross-organisational business processes that have no correspondence in the context of Web services:

- Are the semantics of a remotely defined business process compatible with the security policy effective at the node where it is to be executed?
- Which classification, with respect to access control, is required for the Web service offered by the remotely defined business process in order to be compliant with the security policy in the domain where it will be executed?

While the second question again arises in the context of access control, albeit from a different point of view to the aspect that usual access control approaches address, the first point addresses a new view at access control and beyond, that had not needed to be considered in the context of Web services as it is not relevant with their basic incarnation.

In other related work, Mendling *et al.* (2004) present an approach to addressing the second aspect above. By extracting RBAC models from BPEL scripts, and converting BPEL code in a format suitable for a particular RBAC software component, they provide an automated link of access control requirements into business processes defined by the BPEL scripts.

Security aspects in Web services concern questions like: What kind of privileges are required in order to be allowed to invoke a particular Web service? In the cross-organisational deployment scenario of Figure 1, the view to security is taken from an opposite direction, aiming to questions like: What functionality is allowed to be provided by a remotely defined business process with respect to the security policy effective in the domain of execution? The answer to this question may in most cases depend on the intended use of the Web service provided by this business process. To keep it simple, we assume, that

a) the domain where the BPEL script is specified and from where the scripts is sent cross-domain to the system where it will be executed, is identical with the domain invoking the new Web service provided by the business process, for instance domain A in the scenario of Figure 1;
b) all potential invokers of this new Web service from outside the domain running it are supposed to be residing in the domain where the BPEL script was specified, for instance also domain A; and
c) with respect to access control and potential other security aspects relevant in the relation between both domains, all potential external invokers are provided the same set of privileges.

Given this preconditions, the answer to the above question as to the allowable functionality of the business process is related to the set of privileges owned by the invokers. In terms of RBAC (Ferraiolo *et al.,* 2001), due to precondition c) all invokers are associated with the same role. Hence, the answer is related to this role, that means, in the above scenario, it depends on the role associated to invokers in domain A with respect to domain B. At this point, it becomes obvious that both security issues identified above with respect to the scenario of Figure 1 are closely related. They may be considered to be complementary to each other, since the first issue is taking the view from inside to outside, while the second one is taking the view from outside to inside.

The research project will concentrate on the first issue in the list above. It started with a literature review of current research on standardized business process definition languages and their comparability (Aalst *et al.,* 2002, Shapiro, 2002, Wohed *et al.,* 2002) and convertibility to each other (Fischer and Wenzel, 2004). Furthermore, study has been carried out to investigate security relevant semantics of business process definition languages, in particular BPEL, as will be presented in section 3. Study has also been dedicated to the formalisation of security policies aiming to facilitate analysis of business processes with respect to compliance to the restrictions imposed by them (Fischer *et al.,* 2005). In addition, a concept for a security infrastructure coping with issues raised particularly by cross-organisational or cross-domain deployment of business processes specified using BPDLs has been developed and published (Fischer *et al.,* 2005). This security infrastructure and the formalisation of security policies will be further developed as appropriate during the research project.

The project will also investigate to which extent the security issues raised by cross-organisational deployment of business processes and the solution proposed for them may be transferred to the field of grid/utility computing (Foster and Tuecke, 2005). After the definition of requirements for a proof of concept, an evaluation prototype will be developed. Upon completion of the implementation, simulation and evaluation will be carried out using this prototype to investigate the applicability to varying usage scenarios as well as the feasibility of automated security assessment processing. At the end of the project, the concept will be updated in the light of the insights from the simulation and evaluation performed.

In this paper, we will consider in detail the security-relevant semantics of BPEL that need special attention in the security assessment of business processes when used in BPEL scripts defining them.

3. Analysis of Security-Relevant Semantic Patterns of BPEL

The analysis of the semantics of code written in programming languages is a well-known difficulty (Cousot, 1999). Therefore, the need to analyse the semantics of a BPEL script with respect to security-relevant semantics will make this approach of cross-domain definition and execution impractical unless this analysis can be provided automatically, at least to a large extent.

Fortunately, the nature of BPEL (as well as of other business process languages) accommodates this analysis. This is further supported by the fact that no thorough analysis of each and every particular aspect of the semantics will be required. Instead, only a direct search for features potentially violating the security policy of the target domain will be sufficient. In order to be able to perform the analysis this way, the security-relevant semantic pattern of BPEL as a scripting language has to be analysed. This will be done in this section.

In order to answer the question whether any given BPEL script is compliant to the security policies effective at the system running the business process defined by it, it is important to investigate which semantics described in a BPEL script could be detrimental with respect to security (*i.e.*, have the potential to compromise the policies). Therefore, the language constructs of BPEL and their implied semantics have to be examined as to which extent they might be in conflict with security restrictions.

The analysis of the security-relevant semantics of BPEL makes use of the fact, that BPEL like other business process definition languages offers little or no means for defining data processing or computational tasks as part of the language itself. For these purposes, BPEL scripts have to invoke Web services or must import constructs from other XML standards such as XPath (Berglund *et al.,* 2004). Furthermore, security aspects such as authentication, provision of secure communication channels, non-repudiation are not to be considered in this context, since they usually are catered for by the platform running BPEL scripts and the language does not provide any means related to these security aspects.

Thus, the analysis will concentrate on the business or workflow logic, that can be expressed in BPEL. While a detailed description of BPEL can be found in its specification (Arkin *et al.,* 2004), a comprehensive analysis of the semantics of BPEL was conducted by Wohed *et al.,* (2002) based on a former version of the BPEL specification. An overview of the language and a comprehensive example is given by Leymann and Roller (2004). In BPEL, two types of processes may be modelled: executable and abstract processes. As abstract processes are not executable by their definition, they are not in the scope of our analysis. Executable processes specify workflow logic in terms of activities. The prevalent semantics expressed in BPEL is the exchange of messages with one or several partners, that can be thought of a invoking Web services provided by partners or being invoked as a Web service by partners. In a definition part, BPEL script define the potential links to external partners by references to WSDL definitions (Christensen *et al.,* 2001) of the Web services involved.

The activities expressing the semantics of a business process may be either primitive or complex. BPEL provides the following primitive activities:

- **`invoke`** invocation of a Web service
- **`receive`** waiting for a message to arrive
- **`reply`** sending a reply to a message received
- **`assign`** assignment of values between two different locations
- **`wait`** waiting for a specified amount of time
- **`throw`** indication of exceptions such as failures during execution
- **`exit`** termination of a process instance (note: was **`terminate`** in previous versions of BPEL)
- **`empty`** no operation

The structured activities provided by BPEL are:

- **`sequence`** definition of a fixed execution order
- **`flow`** parallel execution of activities

- `switch` branching between several alternate activities depending on conditions
- `while` iterative execution (*i.e.*, looping)
- `pick` waiting simultaneously for several events to occur and proceeding depending on the event that actually occurs; typically, on event waiting for is a timeout event

In order to avoid compromising of security policies, analysis of access control and information flow control as the mechanisms of choice for this purpose (Dobson, 1994) has to be conducted. Access control to the Web service offered by a business process under consideration is the concern of a complementary security issue not addressed in this paper as stated above (*cf.* Mendling *et al.,* 2004). Hence, the analysis addressed in this section aims to examine whether information or resources accessed and the flow of information from inside to outside the domain and vice versa are consistent with the limitations of the security policy.

Possible violations of the policy are:

- making information or use of resources available outside the domain beyond the restrictions imposed by the policy, for instance, reading information from a data base and sending it to an external partner;
- bringing information from outside into an internal data storage that is not allowed to be written to from external sources; and
- using functionality or resources that are not allowed to be used, for instance, writing into data storage or exercising a system control function.

As the language constructs are not security-relevant as such, they have to be considered in conjunction with access to information or resources. Since in BPEL scripts access to information or resources may only be gained via Web services, the language constructs will be investigated in conjunction with different types of Web services. Given a particular set of rules in security policies and a particular set of privileges (*i.e.*, a particular role), the following cases, presented as three groups A through C, will be distinguished:

A) invocation restrictions implied by (lack of) knowledge of semantics of Web services invoked:
   1) Web services having well-known semantics within this domain. Internal Web services, *i.e.*, those defined and executed in the same domain that runs the business process, are expected to belong to this group.
   2) Web services not having well-known semantics, *i.e.*, its semantics is only known to a certain extent (as far as is required for invoking them). External Web service, *i.e.*, Web services residing outside the domain that runs the business process, are most likely to belong to these group. Therefore, there may exist some uncertainty in judgement of security risks related to their invocation

B) invocation restrictions with respect to the location of the invoker:
   1) Web service accessible independent from the location of the invoker

2) Web services that are allowed to be accessed only from inside domain B, but not from inside domain A
3) Web services that are not allowed to be accessed, neither from inside domain B, nor from inside domain A

C) invocation restriction with respect to (parts of) resources or information:
1) Web service with unrestricted access to all parts of resources or information offered
2) Web service with restricted read access: some information made accessible are not allowed to be carried outside domain B, *i.e.*, parameters returned by Web service are partially read protected
3) Web service with restricted write access: some of the input parameters of the Web service are not allowed to be used
4) Web service with restricted write access: some of the input parameters of the Web service may only be used with particular values, while others may be used without restrictions
5) Web service with restricted write access: for some of the input parameters of the Web service only values from particular sources may be used, for instance, only values returned by a particular Web service

Invoking Web services belonging to these classes in combination with the semantics provided by BPEL will be investigated as semantic patterns in order to determine their relevance with respect to security policies, in particular access control and information flow control.

It is noted, that the following considerations all refer to a given set of permissions or restrictions derived from security policies related to an external user (human or system) accessing resources or information in a particular security domain (*i.e.*, relative to a given role in terms of RBAC). The functionality of business processes brought in from domain A into domain B for being executed there and used by a user in domain A has to be consistent with this given role.

It is further noted, that the analysis of the security-relevant semantic patterns assumes, that the restrictions derived from security policies shall be as strict as required to avoid any violation of security policies, but also as weak as possible to allow as much functionality as possible in a business process within the limits imposed by security policies.

To reduce complexity, it may be assumed, that for Web services in case A1, it is known which cases of group B and C apply. For Web services in case A2, it is expected to be known which cases of group B apply, but only C1 of group C is relevant in this case, as cases C2 through C5 would require knowledge with respect to the semantics, that may be not available in domain B.

Obviously, Web services with unrestricted access permission as well as Web services with global access restriction (*i.e.*, cases B1&C1 and B3 above) do not pose any particular challenge for analysis. In these cases, any further distinction between cases of group A and group C above is not relevant. The reason for this is, that their allowed or forbidden use in a BPEL script, respectively, may be detected by examining the definition part in a straightforward manner. No Web service with global access restriction (B3) must occur in the definition part, or at least, if such Web services should occur, they must not be used in any communication performed in the business process.

If only Web services with unrestricted accessibility occur (B1&C1), the business process could also be executed at the domain where it is specified. The only difference in having such a business process executed in a different domain is the fact, that computational and communicational load involved is moved to this other domain. With respect to security, this is only relevant as bearing the potential for making exhaustive use of computational or communicational resources of this other domain. When driven to an extreme, this could cause a sort of denial of service attack in this domain. Since such exceptional behaviour may easily be controlled by the run-time environment executing the BPEL script, this is not considered constituting a security threat that needs particular examination before running a BPEL script. However, detecting such behaviour by analysing the BPEL script prior to execution is also feasible. For the sake of volume of this paper, this aspect will be not addressed here any further.

If at least one of the Web services used in a business process belongs to a case other than C1 in group C, then the information flow from and to Web services belonging to cases B1&C1 require further attention as will be discussed below. However, this is considered belonging to the risks due to Web services of cases C2 through C5, but not due to the Web services of cases B1&C1.

For case B2, Web services belonging to case A2 may sensibly only belong to case C1 (cf. above discussion). In contrast, if they belong to case A1, it is supposed, that from the security policy, it is decidable which case of group C applies to each of them. If B2 and C1 applies, Web services may be invoked without any restriction from within domain B, independent of cases A1 or A2. Thus, their invocation does not imply any security risk, but information flow to and from these Web services will need further attention, if at least one Web service belonging to a case other than C1 of group C is involved in the business process, as was already the case above with Web services belonging to cases B1 and C1.

Therefore, only group C needs to be considered in detail in conjunction with the semantics provided by the language construct. The results of the analysis is depicted in Table 1. Consistent with the foregoing discussion, case C1 does not involve any security risks with any of the activities defined in BPEL. Also many other combinations of language constructs in presence of Web services of cases C2 through C5 in the same BPEL script do not cause any harm.

Language Construct	**Case C1**	**Case C2**	**Case C3**	**Case C4**	**Case C5**
`invoke`	no harm	no harm, IFA required (see note 1)	writing to restricted input parameter	using forbidden values, IFA required (see note 2)	using forbidden values, IFA required (see note 3)
`receive`	no harm	no harm, IFA required (see note 1)	no harm	no harm	no harm
`reply`	no harm	no harm	writing to restricted input parameter	using forbidden values, IFA required (see note 2)	using forbidden values, IFA required (see note 3)
`assign`	no harm	no harm, to be considered in IFA	no harm	no harm, to be considered in IFA	no harm, to be considered in IFA
`wait`	no harm	duration dependent on restricted value	no harm	no harm	no harm
`throw`	no harm	condition thrown dependent on restricted value	no harm	no harm	no harm
`exit`	no harm	condition for termination dependent on restricted value	no harm	no harm	no harm
`empty`	no harm	no harm	no harm	no harm	no harm
`sequence`	no harm	no harm	no harm	no harm	no harm
`flow`	no harm	no harm	no harm	no harm	no harm
`switch`	no harm	switch dependent on restricted value	no harm	no harm	no harm
`while`	no harm	loop control dependent on restricted value	no harm	no harm	no harm
`pick`	no harm	timeout dependent on restricted value; for values read: IFA required (see note 1)	no harm	no harm	no harm

Table 1: Potential Violation Analysis for Internal Web Services

Note 1: Information flow analysis (IFA) is required as to how restricted information read is used during further processing

Note 2: IFA is required as to where information comes from, that is going to be written to a restricted input parameter in order to determine if the values are consistent with the restrictions applying to the particular input parameter.

Note 3: IFA is required as to where information comes from, that is going to be written to restricted input parameter in order to determine it comes from a source allowed for this input parameter

As indicated in Table 1, some activities require special attention with respect to information flow. Analysis of information flow is required, if a Web service belonging to case C2 is used in the one of the activities **`invoke`** (with respect to the inbound parameters), **`receive`** or the **`on message`** part of **`pick`** to determine if the restricted information returned by them is kept inside the business process and is not send outside via one of the activities **`invoke`** (with respect to the outbound parameters) or **`reply`**.

In C3, only **`invoke`** (with respect to the outbound parameters) and **`reply`** need special attention to check that the outbound parameters of the particular Web service will not be used. Cases C4 and C5 are similar, since with **`invoke`** (with respect to the outbound parameters) and **`reply`** information flow is required to determine if the restricted use of values is obeyed.

Analysis of information flow will embrace **`assign`** activities to observe the movement of information within the business process. If processing such as calculation or string manipulation is being performed using language constructs imported from XPath (Berglund *et al.,* 2004) occurs, it has to be analysed that no restricted information is involved, or at least, that results from this processing is not used in a manner violating the security policies. Since allowing processing on restricted information could cause obfuscation of information flow, thereby complicating the analysis of information flow, such processing should be considered violating the security policy independent of the further use of the results.

As special cases, use of restricted information gained from Web services in case C2 in the activities **`wait`** (duration), **`throw`** (exception thrown), **`exit`** (condition for termination), of and **`switch`** (definition of cases), **`while`** (loop control), and **`pick`** (timeout interval) has to be considered in the analysis of information flow. If any of the terms indicated in parenthesis is defined dependent on such restricted information this could be used to circumvent the restrictions implied by security policy. For instance, if the restricted information *I* (in case C2) is used to control the amount of cycles of looping in a **`while`** activity, this could be exploited to circumvent the restriction on *I.* Providing some external observable behaviour such as sending a message to an external Web service within the loop body would enable an external observer to count the numbers of messages observed. From this count the value of *I* could be revealed to the external observer, thereby violating the security policy that restricted this information from being disclosed outside the domain.

4. Conclusions and Further Work

In this paper we have presented an analysis of security-relevant semantics patterns of business processes that are defined using BPEL scripts externally from the security domain where they are to be executed. The security risks involved by applying particular constructs of BPEL in conjunction with various types of restrictions on the use of Web services implied by security policies have been considered. Having determined the security-relevance of the different semantic patterns allows for specifying security policies in terms of such patterns (*i.e.*, language constructs in conjunction with particular Web services). In (Fischer *et al.,* 2005), we have defined a formalism for the definition of security policies based on security-relevant semantics of business processes and have described an infrastructure that supports the analysis of distributed developed and executed cross-organisational business processes. It is expected that coping with security issues arising from this way of applying standardized BPDLs such as BPEL will foster the acceptance of cross-organisational developing business processes, that has already been made technically feasible by these standards, in practical applications.

Further work will be addressed to the definition of an XML schema in order to provide a formal way of expressing restrictions on BPEL scripts and to serve as basis for the construction of a research prototype.

In addition, the applicability of approach to grid computing where grid processes are being defined using BPEL will be investigated.

5. References

Aalst, W.M.P. v.d., Dumas, M., ter Hofsted, A.H.M., and Wohed, P. (2002) “Pattern Based Analysis of BPML (and WSCI)” *Technical report FIT-TR-2002-05*, Queensland University of Technology, May 2002, http://sky.fit. qut.edu.au/~dumas/bpml_report.pdf, last accessed 2004-12-28

Abendroth, J. and Jensen, C.D. (2003) “Partial Outsourcing: A New Paradigm for Access Control” In *Proceedings of the 8th ACM Symposium on Access Control Models and Technologies, SACMAT'03*, pages 134–141, June 2003, Como, Italy

Arkin, A. (2002) “Business Process Modeling Language”, *BPMI.org*, November 2002, http://www.bpmi.org/ bpml-spec.htm, last accessed 2005-02-25

Arkin, A., Askary, S., Fordin, S., Jekeli, W., Kawaguchi, K., Orchard, D., Pogliani, S., Riemer, K., Struble, S., Takacsi-Nagy, P., Trickovic, I., and Zimek, S. (2002) “Web Service Choreography Interface (WSCI) 1.0”, *W3C,* 8 August 2002, http://www. w3.org/TR/2002/NOTE-wsci-20020808, last accessed 2004-12-28

Arkin, A., Bloch, B., Curbera, F., Goland, Y., Kartha, N., Liu, C.K., Thatte, S., and Yendluri, P. (2004) “Web Services Business Process Execution Language Version 2.0”, *OASIS*, December 2004, http://www.oasis-open. org/committees/download.php/10347/wsbpel-specification-draft-120204.htm, last accessed 2004-12-28

Berardi, D., De Rosa, F., De Santis, L., and Mecella, M. (2003) “Finite State Automata as Conceptual Model for E-Services”, In *Proceedings of the 7th World Conference on Integrated Design and Process Technology, IDPT-2003,* June 2003

Berglund, A., Boag, S., Chamberlin, D., Fernández, M.F., Kay, M., Robie, J., and Siméon, J. (2004) “XML Path Language (XPath) 2.0”, *W3C*, 29 October 2004, http://www.w3.org/TR/xpath20, last accessed 2004-12-28

Christensen, E., Curbera, F. , Meredith, G., Weerawarana, S. (2001). “Web Services Description Language (WSDL) 1.1”, World Wide Web Consortium, March 2001. http://www.w3.org/TR/2001/NOTE-wsdl-20010315, last accessed 2005-09-24.

Coetzee, M. and Eloff, J.H.P. (2003), “Virtual Enterprise Access Control Requirements”, In *Proceedings of the 2003 annual research conference of the South African institute of computer scientists and information technologists on Enablement through technology,* pages 285–294, 2003

Cousot, P. (1999) “Directions for research in approximate system analysis”, *ACM Computing Surveys (CSUR),* Volume 31, Issue 3es , September 1999

Curbera, F., Khalaf, R., Mukhi, N., Tai, S., Weerawarana, S. (2003), “The Next Step in Web Services”*, Communications of the ACM,* 46(10):29-34, October 2003

Dimmock, N., Belokosztolszki, A., Eyers, D., Bacon, J., and Moody, K. (2004) “Using Trust and Risk in Role-Based Access Control Policies”, In *Proceedings of the 9th ACM Symposium on Access Control Models and Technologies, SACMAT'04*, pages 156–162, June 2004, Yorktown Heights, New York, USA

Dobson, J. (1994), "Messages, Communications, Information Security and Value", In *Proceedings of the 1994 workshop on New security paradigms*, pages 10–19, Little Compton, RI, USA, August 1994

Ferraiolo, D., Sandhu, R., Gavrila, S., Kuhn, R., Chandramouli, R. (2001). “Proposed NIST standard for role-based access control”*, ACM Transactions on Information and System Security (TISSEC)* 4(3): 224–274, 2001

Fischer, K.P., Bleimann, U., Fuhrmann, W., Furnell, S. (2005), “A Security Infrastructure for Cross-Domain Deployment of Script-Based Business Processes in SOC Environments”, In *Proceedings of the 5th International Network Conference, INC’2005,* pages 207–216, Samos Island, Greece, July 2005

Fischer, O. and Wenzel, B. (2004) “Prozessorientierte Dienstleistungsunterstützung: Workflowbasierte Komposition unternehmensübergreifender Geschäftsprozesse”, University of Hamburg, http://vsis-www.informatik.uni-hamburg.de/getDoc.php/thesis/177/DA-Wenzel-Fischer-final.pdf, last accessed 2004-12-28

Foster, I. and Tuecke, S. (2005), “Describing the elephant: the different faces of IT as service”, *Enterprise distributed computing, Queue,* 3(6): 26-29 July/August 2005

Koshutanski, H. and Massacci, F. (2003) “An Access Control Framework for Business Processes for Web Services”, In *Proc. of the 2003 ACM Workshop on XML Security*. pages 15–24, October 31, 2003, Fairfax VA, USA

Leymann, F. and Roller, D. (2004), "Modelling Business Process with BPEL4WS", In *Proceedings of the 1st Workshop on XML Interchange Formats for Business Process Management (XML4BPM'2004)*, pages 7–24, March 2004, Marburg, Germany,

Lippe, S., Greiner, U., Barros A. (2005). "Survey on State of the Art to Facilitate Modelling of Cross-Organisational Business Processes", In *Proceedings of the 2nd Workshop on XML Interchange Formats for Business Process Management (XML4BPM'2005)*, pages 7–22, March 2005, Karlsruhe, Germany, http://wi.wu-wien.ac.at/ ~mendling/XML4BPM2005/ xml4bpm-2005-proceedings.pdf, last accessed 2005-09-23

Malu, P., Dubray, J.J., Lonjon, A., Buchinski, E., Chan, A., Mukkamala, H., and Smiley, D. (2002) "ebXML Business Process Specification Schema, Version 1.05", *UN/CEFACT and OASIS*,. http://xml.coverpages.org/ ebBPSSv105-Draft.pdf, last accessed 2005-02-25

Medjahed, B., Benatallah, B., Bouguettayaet, A., Ngu, A.H.H., and Elmagarmid, A.K. (2003) "Business-to-business interactions: issues and enabling technologies", *VLDB Journal* (2003) 12: 59-85, April 2003

Mendling, J., Strembeck, M., Stermsek, G., and Neumann, G. (2004) "An Approach to Extract RBAC Models from BPEL4WS Processes", In *Proc. of the Thirteenth IEEE International Workshops on Enabling Technologies: Infrastructures for Collaborative Enterprises (WET ICE 2004)*, pages 81–86, Modena, Italy, June 2004

Papazoglou, M.P. and Georgakopoulos, D. (2003) "Service-Oriented Computing", In *Communications of ACM*, 46(10):25–28, October 2003

Tuecke, S., Czajkowski, K., Foster, I., Frey, J., Graham, S., Kesselman, C., Maquire, T., Sandholm, T., Snelling, D., and Vanderbilt, P. (2003) "Open Grid Services Infrastructure (OGSI) Version 1.0", *Global Grid Forum, GGF*, June 2003, http://www.ggf.org/documents/GWD-R/GFD-R.015.pdf, last accessed: 2004-12-28

Wang, H., Huang, J.Z., Qu, Y., Xie, J. (2004). "Web services: Problems and Future Directions", *Journal of Web Semantics,* 1(3): 309-320, April 2004

Wohed, P., van der Aalst, W., Dumas, M., and ter Hofstede, A. (2002). "Pattern-Based Analysis of BPEL4WS", *Technical report, FIT-TR-2002-04*, Queensland University of Technology, Brisbane, 2002, http://tmitwww.tm.tue. nl/research/patterns/download/qut_bpel_rep.pdf, last accessed 2004-12-28

Workflow Management Coalition (2002) "Workflow Process Definition Interface - XML Process Definition Language, Version 1.0", October 2002, http://www.wfmc.org/standards/docs/TC-1025_10_xpdl_102502.pdf, last accessed 2004-12-28