CADENUS SECURITY CONSIDERATIONS

Gašper Lavrenčič[1], Borka Jerman-Blažič[2], Aleksej Jerman Blažič[3]
[1]SETCCE, [2]University of Ljubljana, [3]Institut Jožef Stefan, Ljubljana, Slovenia

Abstract: This paper deals with the development of security service provision required for end-user services in Premium IP networks that are being developed and tested within CADENUS project funded under an EU initiative. The initiative is focused towards setting up and experimenting with mechanisms that enable dynamic user-services creation, configuration and provisioning with in-built QoS. The authentication, authorization, data integrity and non-repudiation services are provided through exchange of information and documents between the mediators communicating with end user, the network resource manager and the service manager acting in the middleware part of a Premium IP network. The exchange is using the technology specified in the Extensible Markup Language (XML) messaging with the additions taken from the emerging Electronic Business XML (ebXML) model. The paper presents the XML/ebXML specified security mechanisms, which are in-built in the CADENUS architecture.

Key words: XML, ebXML, digital signature, Access Mediator, Service Mediator, Resource Mediator, Collaboration Protocol Profile/Agreement, Messaging Services

1. INTRODUCTION

The current Internet is based on the best-effort model, which does not provide any traffic segregation or differentiation within the network. Guaranteed delivery or high-priority delivery provides a much-needed alternative to the best-effort networks of today. In order for networks to continue to grow, to satisfy new service needs, and to expand to serve real-time applications, networks must provide mechanisms that ensure delivery within sensible bounds or offer preferential treatment to certain traffic. In recent years, much industrial and academic effort has been devoted to the

The original version of this chapter was revised: The copyright line was incorrect. This has been corrected. The Erratum to this chapter is available at DOI: 10.1007/978-0-387-35612-9_23

B. Jerman-Blaži et al. (eds.), *Advanced Communications and Multimedia Security*

definition of schemes and architectures that provide guarantees to data communication carried over packet-switched networks. These efforts have achieved several interesting results: the Integrated Services (IntServ) model [10], the Differentiated Services (DiffServ) model [11], Multi-Protocol Label Switching (MPLS) [12], etc. by the IETF [13] community, as well as proposals for the efficient integration of IP and ATM are among the most prominent examples. However, in the case of the IETF, much of the work is yet to be completed in terms of real specifications and implementation in running networks. In this document, Premium IP networks refer to these new IETF network technologies known as the next generation of IP networks.

The project "Configuration and Provisioning of End-User Services in Premium IP networks" or in short CADENUS is an R&D project from the 5th Framework Program funded by the European Commission. The main focus of the project is integration of existing research in Premium IP networks and end-user services with QoS guarantees. Current DiffServ developments within the IETF have presented an architecture and framework to enable Premium IP network services to be configured and provisioned. These proposals do not, however, describe concrete realisations for networks providing Premium IP using Differentiated Services. In addition, the Premium IP packet stream differentiation is the only aspect of the end-user service offering. Many other problems that deal with the provision of these services on network level are not sufficiently elaborated. In this context a service provider dealing with traffic streams that require multiple QoS traffic streams has to address the bearer level service as well as higher-level service alike. The CADENUS project proposes an integrated solution for the configuration and provisioning of end-user services with QoS guarantees in Premium IP and is exploring the possibility and benefits of bringing classes of end-user services into a unified framework.

The CADENUS framework is a set of core functionality at the user - provider interface in Premium IP. The service creation and configuration is provided in a dynamic way by linking user-related service components (authentication, authorization, registration, subscription, etc.) to network related service components (QoS control, resources management, accounting, call control, etc.). This paper discusses the security solutions proposed and implemented within CADENUS framework that links relevant end-user request and network related service components.

2. SHORT OVERVIEW OF THE CADENUS ARCHITECTURE

The architecture of the integrated solution for provision of end-user services in Premium IP networks is based on elements that support service creation and service configuration in QoS aware IP networks. Closely related to this activity is the management of those resources on the underlying networks that are reserved on registration/subscription, and those that are used and maybe subsequently modified – when the service is invoked/configured. Associated with the reservation and usage of resources is the automated production and presentation of the corresponding SLAs (Service Level Agreement) to the user, and SLA-SLS (Service Level Specification) translation. CADENUS architecture (see Figure 1) introduces 3 components that are necessary to supervise the dynamic service creation configuration process: Access Mediator (AM), Service Mediator (SM) and Resource Mediator (RM).

The architectural platform is distributed among various mediators for its flexibility and guaranty of QoS. Each mediator is designed for its administration environment to ensure transparency of the services offered (e.g. backbone providers, internet service providers).

AM is a device into which users input their requests to the system. Whilst optional, the AM adds value for the user, in terms of presenting a wider selection of services, ensuring the low service cost, and offering harmonised interface. In the CADENUS model AM presents the current service offer to the user. The source of services is stored in the Service Directory database (SDDB). The AM maintains some of the elements allowing access control for the end user in order to assist and ease the service selection process. The usage of the service involves two business processes: registration of the user to the service and invocation of the service at the moment it is used (change of parameters of the same services during a session is considered as new invocation). AM may form associations with one or more Service Mediators to which requests are issued. SM is responsible for finding and in some cases, building from individual elements – the requested service, through a process that involves information request from the Resource Mediators and later selection of the appropriate Resource Mediator. Resource Mediators are responsible for selection of the appropriate network capabilities from the available options in the underlying network. The service creation and service configuration is achieved from components of all architectural elements (AM, SM, RM) and is presented to the user in the form of SLA.

The Service Mediator (SM) supervises the incorporation of new services in the SDDB and the management of the physical access to these services via the underlying network, using the Resource Mediator(s) - RM. AM prepares

a SLA, containing data about a service chosen by a user (e.g VoD, Best Quality), passes it to SM, that transforms data from SLA to service technical specification document –SLS, (e.g Best Quality=2 Mbit/s) and passes it to the RM.

The SM has an important role as this is the place where created services are attached, and from where the impacts of service reconfigurations are communicated to the network resource management. A task of the SM is also to make information about new service offerings available to AM. SM is also responsible for refreshing and up dating the system responsible for management of all services with new rules (if they appear) and configurations of the devices affecting the network resource management functionality.

Reservation of the network capacities is done with communication between the SM and RM. This communication is generic and is independent of the used network technology. The reservation of the network capacities is specified in the SLSs. In the CADENUS architecture the RM is responsible for the end-to-end view of the network QoS. This is done through communication of the RM with the appropriate underlying network management systems. A network provider wishing to offer its resources must support an interface capable of handling an SLS, from its network management system to one or more RMs. In order for the RM to maintain and up date the end-to-end network view of the current QoS availability, it may use a set of policy that is agreed with the underlying network management systems.

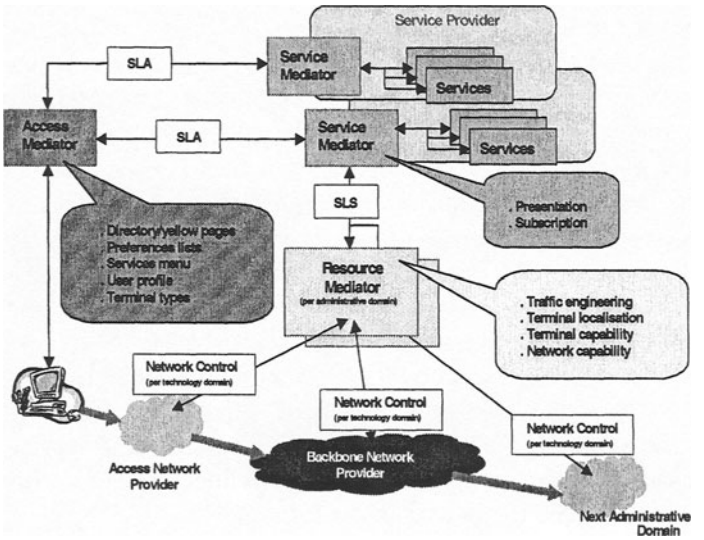

Figure 1. Basic CADENUS architecture

Communication and coordination within the architectural platform is based on XML messaging technique. Extendable Markup Language was chosen to provide transparency and interoperability of the overall architecture. Requests, demands and responses are processed using dynamic XML document generation. Due to the specific request regarding the security provision and the constrains that appeared in the deployment of the platform several new mechanisms have been proposed. Electronic business XML specification was chosen for production of the specification of the overall CADENUS architecture.

3. SECURITY CONSIDERATIONS

The security considerations in the CADENUS architectural model include all aspects of communications e.g. between a client and the platform, between the elements of the platform and between the platform and the underlying network.

Access Mediator acts as an interface between a client and service providers. AM, for example, can select the cheapest offer of a movie available within the offers of more service providers, it can notify the user immediately when a new movie becomes available that matches the stored user profile. In accessing the platform through AM *authentication* is required on the user request level. When authenticated, the user requirements are then captured, and the AM can send the acquired information to the SM who then employs the RM(s) to map the request and to provide the selected service into the physical network. A Trusted Third Party that is a part of a value-added service provider can be involved in authentication. A task of the AM is also to register the users and keep information about their access control parameters.

After the service selection has been agreed with all parties, then a SLA needs to be "signed" between the AM and the SM(s) – in order to provide *data integrity* and *non-repudiation* of the requested and offered service. AM is responsible for the preparation of the SLA. The SM then transforms the SLA into a SLS and passes it to RM(s) enabling the service to be put in place. The provision of integrity and non-repudiation is also provided with digital signature on the SLS. The communication between AM and SDDB also requires data integrity protection as well as authentication of the communicating parties. Here, again the Trusted Third Party services for key management purposes are used.

The RM is associated with the underlying network and its capabilities that provide QoS, but the communication between the SM and the RM is

generic (i.e. independent of the technology employed by the underlying network). RM collects the technical data (e.g. the necessary bandwidth for provision of user-selected service) related to the services from the SM via SLSs, and passes them to the network dependent Network Controller (NC). NC is an entity that contains the underlying DiffServ and/or MPLS network configuration architecture data.

3.1 Selected technology

The CADENUS platform is built on mechanisms for process co-ordination and as a consequence the ebXML specification seemed to be natural tool for specification of these processes and their interactions.

EbXML has its roots in the requirements of business environment for enterprises that dynamically conduct business with each other in an open market environment. The major tasks specified within XML technology are:

- Enabling discovering of products and services in an open on-line market offer.
- Enabling shared business processes and associated document exchanges.
- Setting up contact points and exchange of information for relevant businesses.
- Enable contractual terms for chosen business processes and associated information.

EbXML is being designed to meet the needs for information exchange in business environment in an automated way and as such ensures three basic concepts to be met:

- Infrastructure that ensures data communication interoperability,
- Semantics framework that ensures commercial interoperability and
- Mechanisms that allow enterprises to find each other in the open market on line, agree to become trading partners and conduct business with each other.

The following features of the ebXML specification were relevant in the CADENUS specifications relevant to the security considerations:

- Messaging services.
- Registry and Repository system.
- Collaborative Partner Agreements.

The ebXML Messaging Service [4] specification defines a set of services and protocols that enable electronic applications to exchange data via ebXML messages in the Messaging service definition part of ebXML. The specification allows well-established cryptographic techniques to be used for implementation of strong security mechanisms in the message exchanging process, e.g. HTTP over Secure Socket Layer (HTTPS) can be used for

provision of confidentiality, data integrity and authenticity, as well as digital signatures can be applied to individual messages for provision of authenticity, message integrity authorisation and non-repudiation.

The Registry & Repository part of ebXML specification provides storage rules for all published ebXML documents. It enables storage of company/user profiles and trading partner specifications. The Registry & Repository provides many key functions. For the user it stores company profiles and trading partner specifications. It provides mechanisms for the trading partners to find each other's company business profile.

The Collaborative Partner Agreement specification [3] defines the technical parameters of the Collaborative Protocol Profile (CPP) and the Collaboration Protocol Agreements (CPA). CPP is an XML document that contains technical specifications of the communicating partner and its roles in the business processes. The CPA is an XML document representing technical interception of two (or more) CPPs.

The CADENUS defined SDDB is implemented as a Registry & Repository where the business process specification documents for each type of the service (VPN, VoD, VoIP) are stored. The Service Mediator that is in the Service Provider's domain creates different business process specification documents [2], e.g. one for each service type (VoD, VPN or VoIP), according to the ebXML Process Specification Schema. According to the created documents the Service Mediator creates the Collaboration Protocol Profile ebXML document(s) defining its collaboration role in business process, and stores it in the ebXML Registry & Repository. The Access Mediator creates (in the same way the Service Mediator does) Collaboration Protocol Profile ebXML document according to the Business Process Specification Document(s) in ebXML Registry & Repository. At the service invocation time a user claims his service requirements to the Access Mediator. The Access Mediator browses the Registry & Repository and downloads the Service Mediator Collaboration Protocol Profile, and makes user-understandable information about the service. If a user agrees on the terms the Collaboration Protocol Agreement ebXML document representing the contents intersection of both CPPs is created and installed in the Access Mediator. A communication channel for ebXML messages (SLAs) is established between the Access Mediator and the Service Mediator. The Service Mediator defines the technical data, packs them into SLSs, and passes to the Resource Mediator. The message exchange between the Service Mediator and the Resource Mediator is via XML documents, but itself doesn't belong to the ebXML infrastructure.

3.2 Implementation

ebXML messages used for communications over the Internet are exposed to several third party attacks. From security point of view the most exposed parts of the ebXML architecture are messages exchanged between an AM and the Registry & Repository (CPP, optionally CPA), SLAs exchanged between AM and SM, as well as SLSs between SM and RM.

The disclosure of these messages is not very critical and the confidentiality provision is not necessary. The security concern here is to disable unauthorized parties to alter the message contents and thus message integrity and authentication must be provided as well as reliable delivery. ebXML Messaging Service provides elements for reliable message delivery via implementation of the protocol for message sending repetition on the sender's side. When a message arrives to the recipient an acknowledgement message is created and sent back to the party whose message is acknowledged. In addition to that, the message is digitally signed, providing non-repudiation.

The data integrity of the Service Mediator CPP(s), the Access Mediator CPP and the created CPA within Access Mediator referencing to the business process specification document(s) within the ebXML Registry & Repository is provided by use of XML signatures [6]. This is done in **ProcessSpecification** element described in CPPs and a CPA.

The signature generation steps follow the W3C/IETF recommendations:

3.3 Create a **ds:SignedInfo** element with **ds:SignatureMethod**, **ds:CanonicalizationMethod** and **ds:Reference** elements for the SOAP Header and any required payload objects.

3.4 **Canonicalize** and then calculate the **ds:SignatureValue** over **ds:SignedInfo** based on algorithms specified in **ds:SignedInfo**.

3.5 Construct the **ds:Signature** element that includes the **ds:SignedInfo**, **ds:KeyInfo**, and **ds:SignatureValue** elements.

Example of a digitally signed ebXML SOAP Message can be seen on Figure 2

```
<?xml version="1.0" encoding="utf-8"?>
<SOAP-ENV:Envelope
  xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:eb="http://www.ebxml.org/namespaces/messageHeader"
  xmlns:xlink="http://www.w3.org/1999/xlink">
  <SOAP-ENV:Header>
    <eb:MessageHeader eb:id="…" eb:version="1.0">
    ...
    </eb:MessageHeader>
    <eb:TraceHeaderList eb:id="…" eb:version="1.0">
      <eb:TraceHeader>
        ...
      </eb:TraceHeader>
    </eb:TraceHeaderList>
    <ds:Signature xmlns:ds="http://www.w3.org/2000/09/xmldsig#">
      <ds:SignedInfo>
        <ds:CanonicalizationMethod
Algorithm="http://www.w3.org/TR/2000/CR-xml-c14n-20001026"/>
        <ds:SignatureMethod
Algorithm="http://www.w3.org/2000/09/xmldsig#dsa-sha1"/>
        <ds:Reference URI="">
              <Transforms>
                 <Transform Algorithm="http://www.w3.org/TR/1999/REC-
xpath-19991116">
                    <XPath
xmlns:dsig="http://www.w3.org/2000/09/xmldsig#">
                          not(ancestor-or-self::eb:TraceHeaderList or
                          ancestor-or-self::eb:Via)
                    </XPath>
                 </Transform>
              </Transforms>
          <ds:DigestMethod
Algorithm="http://www.w3.org/2000/09/xmldsig#dsa-sha1"/>
          <ds:DigestValue>...</ds:DigestValue>
        </ds:Reference>
        <ds:Reference URI="cid://blahblahblah/">
          <ds:DigestMethod
Algorithm="http://www.w3.org/2000/09/xmldsig#dsa-sha1"/>
          <ds:DigestValue>...</ds:DigestValue>
        </ds:Reference>
      </ds:SignedInfo>
      <ds:SignatureValue>...</ds:SignatureValue>
       <ds:KeyInfo>...</ds:KeyInfo>
    </ds:Signature>
  </SOAP-ENV:Header>

  <SOAP-ENV:Body>
    <eb:Manifest eb:id="Mani01" eb:version="1.0">
      <eb:Reference xlink:href="cid://blahblahblah"
        xlink:role="http://ebxml.org/gci/invoice">
        <eb:Schema eb:version="1.0"
eb:location="http://ebxml.org/gci/busdocs/invoice.dtd"/>
      </eb:Reference>
    </eb:Manifest>
  </SOAP-ENV:Body>
```

Figure 2. Digitally signed ebXML SOAP Message

The key authentication is to be provided by CAs (Certification Authorities). Additionally, XML Key Management Services, specified by W3C, can be used to delegate some key management functions to a Trusted Service and thus reduce the complexity on the entity's side. The decision which solution will be used depends, on the processing capabilities of the Mediators on one hand, and the Trusted service's server on the other, and the composition of ebXML messages itself. For example, several signatures in a document may use a key verified in an X.509 certificate chain appearing remotely; each signature's **ds:KeyInfo** element can reference this chain using a single Uniform Resource Locator (URL), instead of including the entire chain of X509Certificate elements. The entity that is verifying the signature can then retrieve the whole chain and validate the key, or delegate this task to its Trusted Service.

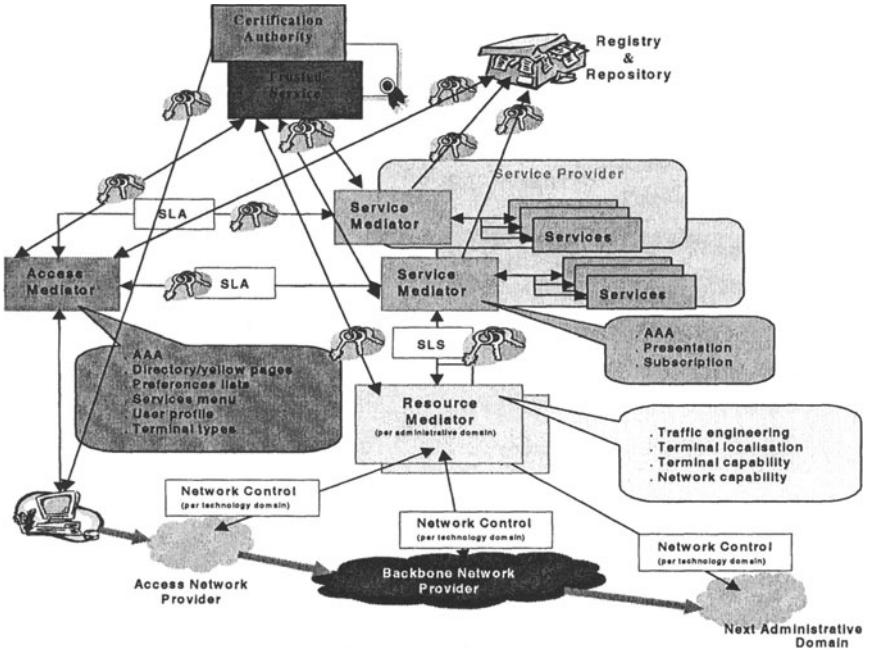

Figure 3. CADENUS secure ebXML/XML architecture

Figure 3 presents the CADEUNS ebXML/XML secure architecture, where keys present secure communication links for message exchange.

4 CONCLUSION

Next generation IP networks are gaining more and more proselytes. Its appeal is due to the given opportunity of a standard and consistent way for network configuration, independently of the underlying architecture and QoS provisioning model assumptions. While this technology is powerful and alluring, it's also generally untested and unproven. The IST project from the 5th Framework Program of EU, CADENUS – Configuration and Provisioning of End User Services in Premium IP Networks is developing an architecture that aims to test and validate the policy and business based approach in a real DiffServ network. The architecture being developed is multi-layered and is based on Internet technologies such as: HyperText Transfer Protocol (HTTP), Transfer Control Protocol/Internet Protocol (TCP/IP), XML as and ebXML. As all these architecture elements are exposed to unauthorised third party attacks, security services are required that provide authentication, authorisation, integrity and non-repudiation of the communications between co-operating parties within the proposed architectural model. Implementation and testing of various specifications for provision of security is part of the work that is on going. In the article the proposed architecture for dynamic creation, configuration and provisioning QoS services to end user with in-built security provision is presented. The security features are based on secure XML/ebXML message transfer and corresponding Key Management functionality. The secure access is not presented as it is supposed that the end user accesses the Access Mediator through secure SSL connection where user proves his authenticity with an X.509 certificate. The model is still under testing in real environment.

REFERENCES

[1] UN/CEFAT and OASIS ebXML Technical Architecture Team: "ebXML Technical Architecture SpecificationV1.04", February 2001, http://www.ebxml.org/specs/ebTA_print.pdf

[2] UN/CEFAT and OASIS ebXML Business Process Team: "ebXML Business Process Specification Schema V1.01", May 2001, http://www.ebxml.org/specs/ebBPSS_print.pdf

[3] UN/CEFAT and OASIS ebXML Trading Partner Team: "Collaboration-Protocol Profile and Agreement Specification V1.0", May 2001, http://www.ebxml.org/specs/ebCPP_print.pdf

[4] UN/CEFAT and OASIS ebXML Transport, Routing & Packaging Team: "ebXML Message Service Specification V1.0", May 2001, http://www.ebxml.org/specs/ebMS_print.pdf

[5] UN/CEFAT and OASIS ebXML Technical Architecture Security Team: "Technical Architecture Risk Assessment V1.0", May 2001,

http://www.ebxml.org/specs/secRISC_print.pdf
[6] IETF/W3C: "XML-Signature Syntax and Processing", W3C Recommendation 12 February 2002, http://www.w3.org/TR/xmldsig-core/
[7] W3C: "XML Key Management Specification (XKMS 2.0)", W3C Working Draft 18 March 2002, http://www.w3.org/TR/xkms2/
[8] Eric Rescorla: "SSL and TLS Designing and Building Secure Systems", ISBN 0-201-61598-3, October 2000
[9] CADENUS project home page: http://www.cadenus.org
[10] IETF Integrated Services home page: http://www.ietf.org/html.charters/diffserv-charter.html
[11] IETF Differentiated Services home page: http://www.ietf.org/html.charters/diffserv-charter.html
[12] IETF MPLS home page: http://www.ietf.org/html.charters/mpls-charter.html
[13] IETF home page: http://www.ietf.org/home.html