VANJA KORAĆ
Mathematical Institute SASA,
Kneza Mihaila 36/III,
Belgrade, Serbia,
e-mail: vanja@mi.sanu.ac.rs

MILAN TODOROVIĆ
Mathematical Institute SASA,
Kneza Mihaila 36/III, 11 000 Belgrade, Serbia,
e-mail: mtodorovic@mi.sanu.ac.rs

DRAGAN PRLJA
Institute for Comparative Law,
Terazije 41, Belgrade, Serbia,
e-mail: dprlja@yahoo.com

004.738.5:338.46
COBISS.SR-ID 264123404
Original research article

Received: January 25th 2018
Accepted: April 30th 2018

FEDERATED IDENTITY CONCEPT BETWEEN THE INSTITUTE OF ARCHAEOLOGY AND VIMINACIUM LOCALITIES

ABSTRACT

In this paper, the concept of a federated identity between the Institute of Archaeology and archaeological sites in Serbia is shown, based on the specific case Viminacium. In this manner, once the processing of a user's identification is performed by one of the identities, the need is eliminated to perform the same procedure for each site, since one can rely on the confidence that the primary subject is capable of providing a user's identity that can be trusted. As a result of such an approach, any user identified by the identity provider "Institute of Archaeology" shall automatically be recognised by service providers at any archaeological site in Serbia, in this particular case at the site Viminacium. In such a way, after a successful employee identification by the identity provider "Institute of Archaeology", all the Institute's employees would posses access to services (for example digital data bases) at the site of Viminacium.

KEYWORDS: FEDERATED IDENTITY, SECURE IDENTIFICATION, IDENTITY PROVIDER, SERVICE PROVIDER, SINGLE SIGN-ON.

The concept of a federated identity is based in law, in cases when there are business subjects establishing a legal relationship.[1] This is further upgraded with an informatics aspect, which gives extra security with the help of an informatics infrastructure. Once the process of user identification is conducted by one entity, there is no need for the second entity to perform the same procedure, since it can rely on the primary subject, being sure that it is capable of providing a user identity that can be trusted. Within such a relationship, two sides can be distinguished: the first one is the identity provider, while the other one is the service provider, offering a business service, but fully relying on the identity provider, since it rep-

1 The article is the result of the project: *Viminacium, Roman city and military camp – research of material and non- material culture of inhabitants by using the modern technologies of remote detection, geophysics, GIS, digitalization and 3D visualization (no 47018)*, funded by The Ministry of Education, Science and Technological Development of the Republic of Serbia.

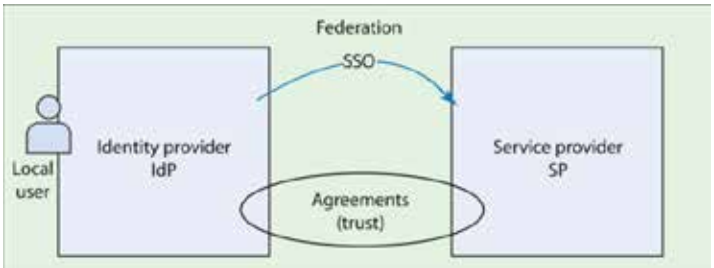


Fig. 1 The concept of federated identity
Source: https://www.ibm.com/developerworks/library/se-jitp/

resents the base for recognising a user's identity. In other words, IdP represents a business entity in charge of user registration and authentication that issues a certificate of established identity to other business entities. On the other hand, SP represents a business entity offering services to users (eg. access to business applications), but it does not establish their identity, since, as already mentioned, it relies on confirmations issued by IdP. ISAM 9[2] represents a concept of this kind that contains several functionalities. One of the functionalities of ISAM 9 is a single sign-on functionality (Fig. 1), intended to assist users that do not possess this benefit. Once they have been introduced to their basic information system of the business entity within which they operate, they are also capable of transparently accessing business system of the service provider without entering their user name or password. In other words, it is sufficient to know only the user name and password of their business system.

Besides the concept of "single sign-on", there is also the concept called "identity provisioning" that runs the life cycles of users' accounts on different systems (while employing, changing jobs etc…). It functions within the basic home organisation when, for example, dealing with employees: When new employees are registered in an organisation by making working contracts, they also need to receive user identities, actually accounts in different systems, either business applications, electronic mail system, data base access if it is an information environment, and so on. This is of special importance when users receive new positions within an organisation that require different access levels (Andronache and Nisipasiu 2011). This concept can be widened when there is a business identity one wants to cooperate with and, in such a case, it is referred to as federated identity provisioning. In such a case, the other business identity also needs to receive information about the user, enabling it to create user accounts for accessing its business applications. One here speaks about the provisioning of running user accounts to different systems, systems owned by a business partner, actually a service provider. Within the frames of standardised mechanisms, a situation can be recognised when such information is spread either via email or paper document or in more developed information structures, when users' identities are advertised to the service provider with ftp or some other mechanism. Provisioning can be more advanced, with IBM defined standards such as the so-called WS-Provisioning. Here, by entering a web service on a provider's page, one can securely create the user's identity, actually running such an account on its own page (Guruprasad and Rajesh 2012).

An even more upgraded mechanism is called "just-in-time provisioning". A user's account is here created by the SP at the moment of user's first access to this system (on the fly) (Ping Identity 2016). In other words, with this mechanism it is not necessary to transfer the entire user population from the identity provider to the environment of the service provider. The reason for this might be that not all of the users have a need to access such services placed on the service provider's page.

2 http://www-01.ibm.com/common/ssi/cgi-bin/ssialias?infotype=an&subtype=ca&appname=gpateam&supplier=897&letternum=ENUS215-191

Provisioning access, actually creating the user accounts can occur only after the first time a user addresses such a service on the service provider's page. Information about the user's identity is then embedded and provisioned as such on the service provider's page and can be used in two different ways. The received information can be used either only for the needs of executing business transactions or it can be used to create a user's identity in a local service provider's repository, further to be used to work on applications on the service provider's page. The method of provisioning a user's attributes (name, surname, email address, personal number and tax number) from the Idp provider to the service provider represents SAML as a part of the "single sign-on" mechanism. The test environment itself is based in the application of such a standard, supported by Oracle, IBM and Microsoft.

Although it is an old standard, accepted in 2005, it was implemented in different producers of application servers. This standard offers a possibility for inter-communication with different environments. The SAML standard defines the format of a message for exchanging confirmations of users' identities, and these are XML messages. These XML messages represent a valid standard for information exchange between different systems. Besides, SAML also defines protocols in the sense of mechanisms for message exchange aimed at specific functionality. For example, it defines protocols for sending users' authentication demands, it defines requirements for users' "log-ins" on "single sign-on" or "log-offs" on several systems and defines information exchange according to their value or their reference. The greatest benefit and specificity of this standard is its bindings, a mechanism for message exchange according to which the messages are transferred from one system to the other. Three initial mechanisms are the most interesting (Novičić and Mitić 2015) (IBM ISAM9 2015**)**:

- HTTP redirect (Browser redirect – no direct communication between IdP and SP)
- HTTP POST (Browser POST)
- HTTP Artifact (Browser artifact – transfers references, while SOAP transfers the real message)
- SOAP (Simple Object Access Protocol – direct communication between IdP and SP)

Within the first three mechanisms, the browser represents a medium for communicating between business partners, identity providers and service providers. That means that there is no need for any communication net between the information systems of identity providers and service providers and it is enough that the browser, actually the user's client, possesses connectivity to the identity providers and service providers. Such a mechanism can also be applied in the Internet environment, if it is the method of accessing identity providers and service providers and can be applied in huge infrastructure intranets of an opened or closed type. These three mechanisms rely on standard http protocols.

The first mechanism, the so-called browser redirect (HTTP redirect) possesses no communication between the IdP and SP. There is a possibility to establish communication when, within the URL itself, actually in the URL arguments, XML zipped information is forwarded encoded from base64. The only limitation is that the URL itself is limited, so a rather small quantity of information can be transferred.

The second mechanism, the so-called HTTP POST (Browser POST) is applied for sending information through a screen form, actually an html form, that is usually hidden within the http response by the service provider. The identity provider requires the user's authentication for the needs of "single sign-on". Such information is hidden within the http format form, after which javascript is activated by the windowslogon trigger and the form is submitted on the identity provider's page. As a response, the identity provider uses the same mechanism to return the response. Those are the XML format documents, carrying information about an authenticated user identity or some of its attributes useful to the service provider.

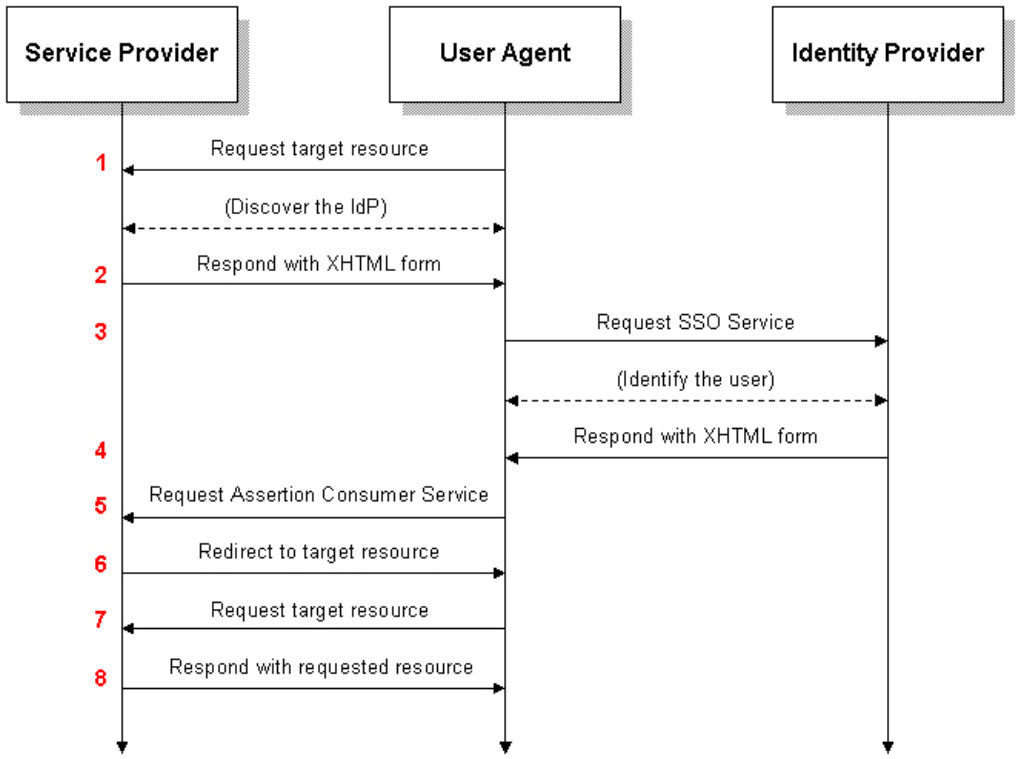


Fig. 2 Use of SAML in a web browser
Source: https://en.wikipedia.org/wiki/Security_Assertion_Markup_Language

The third mechanism, HTTP artifact information transfer, is done through references, while SOAP transfers the actual message. The artifact does not need to be entered into the http communication between browsers, identity providers and service providers in cases when there is no trust by the browser. In such cases, only the reference, actually an identifier of the service message, is transferred from one side to the other, while the systems communicate with each other via SOAP, basically communicating directly with the web services. The actual information is transferred through an alternative channel from one side to the other. It is a condition for the fourth mechanism to connect two systems through web services.

Within ISAM 9 there are profiles that include the most common applications of format standards, protocols and for binding (IBM ISAM9 2015**)**. Such application profiles in a tested environment are presented through a web browser with a web browser single sign-on implementation. In the backend, the ISAM9 infrastructure is performed. What does a standard profile look like and how do messages get exchanged? For example, when it is a web browser with a single sign-on profile and an http post mechanism, in our case, the user agent represents a browser attempting to address the service provider and demand access to a business application (see Fig.2).

This is the general case that is optional, since there can be several identity providers, so in that case it is necessary to decide which identity provider should receive the demand for user authentication. In a test environment, in which there is just a single service provider and a single identity provider, the system will respond with a hidden html form, not visible on the browser. Further on, it will be submitted and the XML formatted information, the authentication demand by the service provider, with the help of web browser user agent, will be forwarded as an http identity provider demand. There are two variants. Either the user has already been logged onto his system at his identity provider in his original environment, so there is

no need to perform authentication, or user authentication has not been performed yet and, in such a case, it would be necessary to make a screen form for the user's name and password in order to access a session. Only after this and based on the verified user's identity, a response confirming the identity will be issued to the browser in a XML formatted token, made using SAML 2 protocol. The web browser later forwards a response obtained in such a way (since it is now a communication medium between the identity provider and service provider) to the service provider's page and submission of this hidden form is automatically performed on the screen. If this XML is validated in the sense of a digital signature and in the sense of its structure and content, a session will be established on the service provider's system. The browser will be redirected to the application so that it can be used on the screen. Session sustainability is made through cookies.

There are three types of confirmations that can be presented within SAML 2 standards. The first one is when the user's authentication has been performed and in such a case, it is important that tokens contain the recognised user. When it comes to the method by which the user was authenticated, it should be noticed that certain systems can demand strong authentication forms, i.e. sometimes the user's name and password are not sufficient, but a smart card token or biometric authentication with fingerprint is also required. In other words, information is important as the method of authenticating users, but for some special purposes an extra step might be required in order to strengthen authentication in the sense of multi-factor authentication. The third, very important factor is the time at which authentication was performed, since it is necessary for these two systems to be chronologically synchronised, i.e. use the same time servers in order to keep the information about an authenticated user safe from misuse.

Apart from the information that the user has been authenticated, basically carrying information about the subject, actually the user's ID, it is also possible to provision the user's attributes, like name, surname or e-mail address, by placing them in the same SAML package and forwarding them to the service provider's page for the needs of performing applicative logic on another page. When it comes to working rights, if there are two or more applications on the service provider's page, it is possible to secure access to one, but not to all the applications for an individual user using the identity provider. Such an information exchange about whether the user is authorised to start an application can also be solved using the SAML protocol, after the identity provider and service provider have communicated with each other about issuing confirmation regarding permission (allowed, not allowed). The identity provider is the one that allows or does not allow the start o certain functions on the service provider's page for a specific user. This is usually not applied, since it belongs to specific applications in a business environment.

TOKEN PROCESSING OF SERVICE PROVIDER ISSUED BY IDENTITY PROVIDER

When, during to front end application access (the end user does not even have to know the exact link from the application to service provider, since it is rather complex), the user reaches the identity provider, it is obliged to issue confirmation of the user's identity. Confirmation in the form of a SAML response reaches the browser via XML and is then forwarded to the service provider (see Fig. 3) (IBM knowledge centre 2016).

As can be noticed in Fig. 3, within the infrastructure of the service provider there is a separated infrastructural part designed to perform the verification of digital signatures from XML, XML structures and to parse it. After it is verified, i.e. when the identity confirmation is adequate, the end result will be the formation of a session on the service provider's page. Such a session is usually

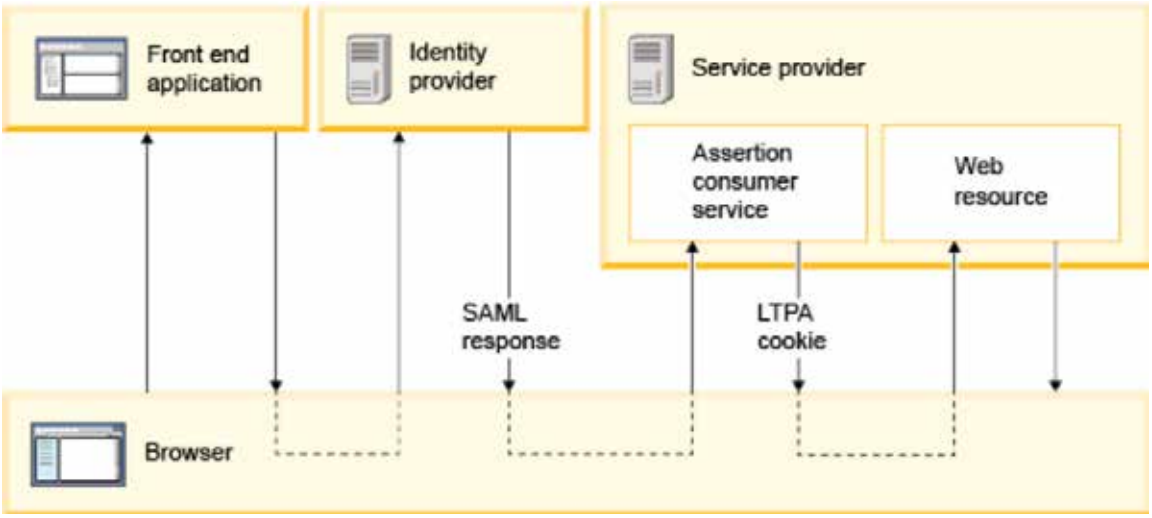


Fig. 3 Procedure of SAML SSO
Source:http://www.ibm.com/support/knowledgecenter/SSAW57_8.5.5/com.ibm.websphere.nd.doc/ae/cwbs_samlssosummary.html

implemented via cookies. In the IBM world, the LTPA (Lightweight Third-Party Authentication) cookie is a standard for maintaining sessions. The above mentioned infrastructure ensures that a user, once authenticated, can keep working with the web application on one (or possibly several) of the supported application servers owing to the fact that each of them trusts the LTPA cookie issued by the infrastructure on the service provider's page. It should also be mentioned that if a trust relationship is established between one or several applications on the service provider's page and the infrastructure, there is no need to change the applications themselves. Once a trust relationship is established within the infrastructure of the service provider, the applications follow the information about the authenticated user and his attributes and can keep working with him without the need to change anything in the application itself regarding login, authentication or authorisation. This represents the method of describing the content of this concept, the processing of a single sign-on message on the service provider's page.

JUST-IN-TIME PROVISIONING – PROCESSING SAML 2.0 TOKENS ON THE SERVICE PROVIDER'S PAGE

There are activities that need to be implemented either separately on each application server or delegated to the infrastructure in which one will be working for the needs of several backend applications on the service provider's page. These activities include: validation of a digital signature and the structure of SAML 2.0 tokens, parsing users' attributes, creation or alteration of local user accounts, establishing a local session and allowing access to a local business application according to the rules given to the user accounts.

An illustrated description of this procedure can be presented through the processes of the mentioned activities. In the first place there is a validation of a digital signature and the structure of the SAML 2.0 token, as well as parsing the user's attributes. When it comes to creating or changing a local user's account, these activities are performed to create a local session. Actually, if there is no user identity, it is possible to widen this process on the service provider's page by creating an identity within the user's register on the service provider's

page. This means that an operation of creating a user identity is performed and if it is recognised that the user's identity already exists, an update can be executed. In a life cycle, updating the user's identity must be foreseen, since users change their work places and gain more or less rights. This is why it is necessary to consider both the creation of the user's identity on the service provider's page and its changes within a life cycle. If this creation process is successful, the next activity is the establishment of a local session by the infrastructure, while the application itself can enable an undisturbed operation if the user possesses adequate membership to groups in the local register, the user's ascribed roles, which enable the starting of certain functions in this application on the service provider's page. It should also be mentioned that there are variations of the working procedure, depending on the individual corporation preferences.

ENABLING THE OPERATION OF THE SP APPLICATION ONLY ON FORWARDED USER IDENTITIES, WITHOUT SAVING THEM IN THE LOCAL REGISTER

In this case, for initiating an application on the service provider's page, it is not necessary to create the user's identity in a local repository on the service provider's page. In other words, according to the forwarded information containing the user's identity, name, surname, personal number and e-mail address, it is possible to make it a part of the token that reached the service provider from the identity provider. Based on this, all transactions can be performed in a business application. After the user has logged out, it is not necessary for all the information to remain within the register of the service provider's page. It is enough that in the transaction log of the business application details are contained that are related to the transaction and the user, which is traceable enough on the service provider's page. Connected to this, it is not necessary to retain the user's repository on the service provider's page in which the user's credentials would be noted, since it is sufficient to rely on what already exists within the infrastructure of the identity provider. This variation can help save privacy. This means that for performing a specific transaction, a SAML session must be established. Then, via SAML, provisioning of all the necessary user attributes is performed by the service provider for that specific transaction. After a user is logged out, all that was in the memory for this specific session object is deleted from the cache. The only trace that it was ever there remains in the transaction logs. This can be of importance for privacy protection on certain business systems.

TAKING OVER USER ATTRIBUTES FORWARDED THROUGH A SAML 2.0 TOKEN WITH AN INTERACTIVE SUPPLEMENT THROUGH A SCREEN FORM BEFORE REGISTERING A NEW USER

In cases when a business case requires the creation of a user's identity on the service provider's page, but where there is an insufficient number of attributes on the identity provider's page, it is possible for the existing attributes to be moved to the service provider's page and later on request to addition them through a screen form. For example, if there is no address on the identity provider's page, but on the provider's side there exists a service for sending email via an application, it is necessary to add this information for registering a user by e.g. adding the email address (all the information will be written in the user's registry on the service provider's page).

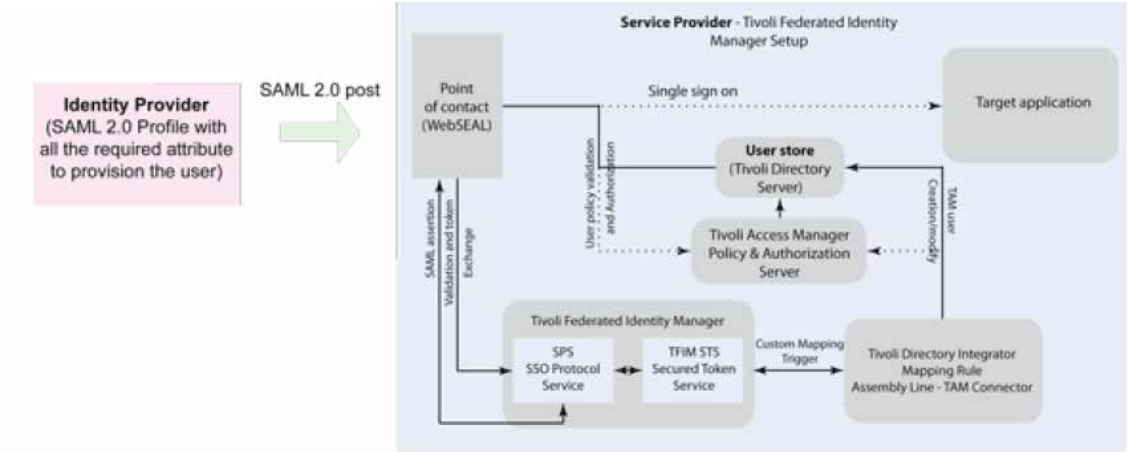


Fig. 4 Components of just-in time provisioning
Source: https://www.ibm.com/developerworks/library/se-jitp/

JUST-IN-TIME PROVISIONING SYSTEM AND ITS COMPONENTS ON THE SERVICE PROVIDER'S PAGE

It is necessary to parse the SAML token on the service provider's page, which carries the certificate of user's identity. Figure 4 shows that the target application can fully rely on the ISAM9 infrastructure regarding the validation of digital signatures, parsing, entering into local users' registry (LDAP or some other structure of users' database). The only thing users' applications need to do is trust the local ISAM9 infrastructure that the authenticated user is really the one that he claims to be over the token. The second way of provisioning attributes is to add attributes into the http demand, originally reaching the web browser, that were previously not there and then parse the SAML 2.0 token content. In such a way, the name, surname and other attributes are entered into the http heading and provisioned all the way to the application. If there is a need, it can be defined for the name and surname to appear in the page header, but it is also possible to use an e-mail address to perform automatic sending or to perform automatic SMS texting using a cell-phone number. Through an http request, such information can simply be entered into the http header, while with a simple code the application can extract it from the parser.

WEB SSO TESTING ENVIRONMENT

A presumption in the testing environment is that there are two fully independent entities: the Institute of Archaeology and the archaeological site Viminacium. Each entity possesses its own users' repository based on LDAP. Furthermore, each identity possesses its own security domain in which access permission is defined. Created users independently possess their passwords and are completely different in these two security domains, but they also possess different net domains. The presumed testing internet environment includes two domains, one of them being ai.ac.rs, while the other one is viminacium.rs. In order to allow the federated concept to be connected in both LDAPs, one must suppose that the same user named "User" is created, but with different attributes, not just according to their value, but also according to their description. For example, on one of the LDAPs, the mail attribute is created, while on the other LDAP, the telephone number attribute is created. Different rights are also defined, since different security domains are assumed. In one of them the user "User" will be a member of some groups, while on the other security domain, it will belong to other groups, since security domains are differently administrated on the identity provider and on the service provider. What is needed to be

shown in the testing environment is a web based single sign-on, on which the logged user from the identity provider's domain will transparently be logged onto the service provider's page without entering the password again, while another benefit can be seen in provisioning the missing attributes through the SAML2 token. Basically, in a different environment, the missing attributes will be provisioned and exposed in an application that is performed on the service provider's page. In other words, provisioning of the user's attributes is performed through a SAML 2.0 token from the identity provider's domain to the business application in a service provider's environment.

In order to secure this, the existence of business applications is also assumed in each environment that can show the user's data. For example:

Arheološki institut (http://miapp1.ai.ac.rs:8080/) application server

Viminacijum (http://sepapp1.viminacium.rs:8080/) application server

In this federated concept, applications are not approached directly, but over a reverse proxy (WebSEAL). The method of setting up the reverse proxy can be seen in the document *SafeNet Authentication Service: Integration Guide* (Gemalto 2016). This reverse proxy has the task to represent itself as the specific server that is used to perform the application. It receives an http request from the web browser and then initiates a new http request to the backend application. Basically, it tricks the backend application by representing itself as a direct client, while it also tricks the client by presenting itself as an application addressing the client. Owing to the fact that it now represents the interception point in the http communication between the client and the application, it can include additional functionalities, i.e. possibilities such as the user's authentication. That means that it alone will perform the user's authentication and not the backend application. It can also authorise users, for example a user can possess the right to access one, but not the other application. It is presumed that the firewall denies access to backend applications, so the reverse proxy represents the meeting point of the user and all the backend applications. This means that only through a firewall can one access the http request by the reverse proxy. Owing to this, it plays the role of both user authenticator and user authoriser. If one considers an Internet environment, it acts as a web application firewall. It takes over the protection of all of the backend applications in the event of malicious attacks. The advantage of this mechanism is that the backend applications do not need to possess implemented attack (threat) protection, since it is all delegated on a single web proxy that has integrated protection mechanisms. In the IBM infrastructure, it is a part of ISAM9 and this component is named WebSEAL. In this testing model, it is designed to initiate and end all the mentioned functions of verifying digital signatures, parsing etc. WebSEAL can parse a SAML 2 token and turn the information from it into the elements of an http heading. We will presume that we have created two applications, one on the identity provider's page in the *Institute of Archaeology (https://miapp1.ai.ac.rs:/app1)* and the other on the service provider's page at *Viminacium (https://sepseal.viminacium.rs/app2/).* Since each entity possesses configured access to the application through the reverse proxy (WebSEAL) that can read users' attributes from LDAP and a parsed SAML 2.0 token and forward them to the application through an HTTP heading, it means that the data received in the http heading is shown on screen. According to this, a processed SAML token entered into the http heading reaches the business application and will show it on screen. What should be mentioned is access via a WebSEAL request for the URL to be accessed will be targeted exactly as WebSEAL https://miapp1.ai.ac.rs:/app1 (in terminology, /app1 is called a junction). The WebSEAL junction represents a TCP/IP connection between the frontend WebSEAL server and backend server (IBM Tivoli Software 2016). The junction hides information about each http request that came to WebSEAL and is intend-

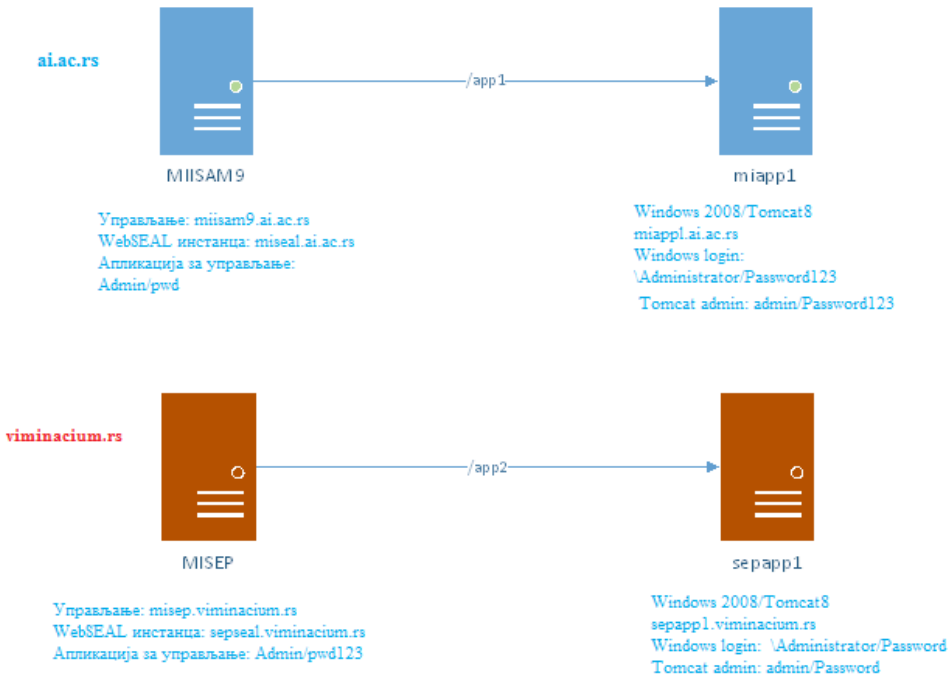


Fig. 5 Logical scheme of identity provider's and service provider's application servers

ed for the app1 application. Actually, it shall be initiated as URL, in this case targeting the appl application by WebSEAL. In another case, on the service provider's page, this junction, actually the logical name of the second application, app2, should internally indicate that each received http request will be turned into a new http request, targeting the backend server. Since WebSEAL is the "man-in-the-middle", it needs to receive information containing the name of the application server to which access is required and this is achieved with a junction, actually the logical name of the background server, e.g. /app1, /app2.

Figure 5[3] shows the logical scheme. Application servers of identity providers are shown in blue, while application servers of service providers are shown in orange (Novičić and Mitić 2015) :

As figure 5 indicates, there are no connections between these applications, actually this infrastructure, since all the communication between them goes over a web browser that can alternately access one server or the other, is the entry point of WebSEAL. This testing environment is made when, on the IdP page and on the service provider's page, adequate GUI ISAM9 wizards are initiated. They represent a series of screen forms in which specific configuration information needs to be entered. These steps define the partner relationship between the identity provider and the service provider. In addition, if attributes need to be provisioned, mapping of the user's attributes needs to be performed within the existing LDAP scheme of the identity provider to the SAML 2.0 attributes that represent the mechanism of their transfer to the service provider. In addition, digital trust is established by entering digital certificates on both sides, issued by the common CA bodies into trust root stores, in order to perform SAML 2 token digital signature validation that they exchanged. It should also be mentioned that the link connecting the identity provider with the service provider is rather complex, but it can be seen basically in the following scheme:

3 Dragan Novičić, Mita Mitić, IBM Security Access Manager 9.0 (ISAM9) - I*dentity Federation scenariji* MI SANU, presentation, SBS, December 2015.

https://miseal.ai.ac.rs/isam/sps/test/saml20/logininitial?RequestBinding=HTTPPost&NameIdFormat=Email&AllowCreate=true&PartnerId=https://sepseal.viminacium.rs/isam/sps/test/saml20&Target=https://sepseal.viminacium.rs/app2

In order to illustrate this, the scheme shows that within the link in a business environment of the identity provider, it targets the server on which ISAM9 is installed and where its WebSEAL component is. Then, ISAM represents a special functionality of a web reverse proxy that is able to generate SAML certificates and provision them to service provider's page. The service provider's address and its ISAM9 component can be seen, while functionality is hidden behind a false junction within the reverse proxy. Behind it there is an initiation of the functionality for creating SAML certificates, while on the service provider's page the validation functionality is hiding, parsing SAML2 certificates. When a request is received on the service provider's page, it needs to be told which backend application server we want to access. In our case that information is part of a complex link that shows in the text above: *Target=https://sepseal.viminacium.rs/app2*. There is actually a need to address the application on the service provider's page exactly through the app2 junction. In other words, with this link created on the IdP business application, the SP business application is accessed, while to the end user it will only represent a hyperlink to be clicked on.

DIFFERENCES OF LDAP ATTRIBUTES IN A TESTING ENVIRONMENT

Further on in this paper, the differences between a user's profile in LDAP on the identity provider's page and the service provider's page will be explained. Figure 6 shows the user's interface LDAP browser on the identity provider's page in which the defined user "User" can be seen, while within his user attributes there is also the email attribute, underlined in red. On the service provider's page in his LDAP, there is no attribute, but there is his cell-phone number as an attribute on his page. Figure 6[4] also shows that the repositories are not identical, but what they have in common is that there is the same user with the same user name on both sides. It should also be mentioned that within ISAM9 there is a LDAP that can be used as a user's repository, although this is not recommended in a production environment, since it is an OpenLdap. The existing user's repository of business environments will be used instead or another, more secure one, will be made. All this indicates that the same identity is created with different attributes in two different LDAPs.

DIFFERENCES OF LDAP ATTRIBUTS IN A TESTING ENVIRONMENT

Regarding authorisation, access rights are defined using memberships in groups. Figure 7[5] shows that within LDAP, a branch is defined with users' groups and that the identity "User" is a member of the group "group1" on the identity provider's page. We can assume that on the service provider's page there is a group called "group3" that, compared to the identity provider, contains completely differently regulated access to groups. Figure 7 shows that the identity "User" is a member of the group "group3". This illustration separates the user's attributes and group memberships, or operation rights in the different systems.

The result is as follows. The user will be applied to separate applications on the identity provider's page (Figure 8) and service provider's page (Figure 9) by using different passwords.

Since there is a link on the identity provider's

4 Dragan Novičić, Mita Mitić, IBM Security Access Manager 9.0 (ISAM9) - *Identity Federation scenariji* MI SANU, presentation, SBS, December 2015.

5 Dragan Novičić, Mita Mitić, IBM Security Access Manager 9.0 (ISAM9) - *Identity Federation scenariji* MI SANU, presentation, SBS, December 2015.

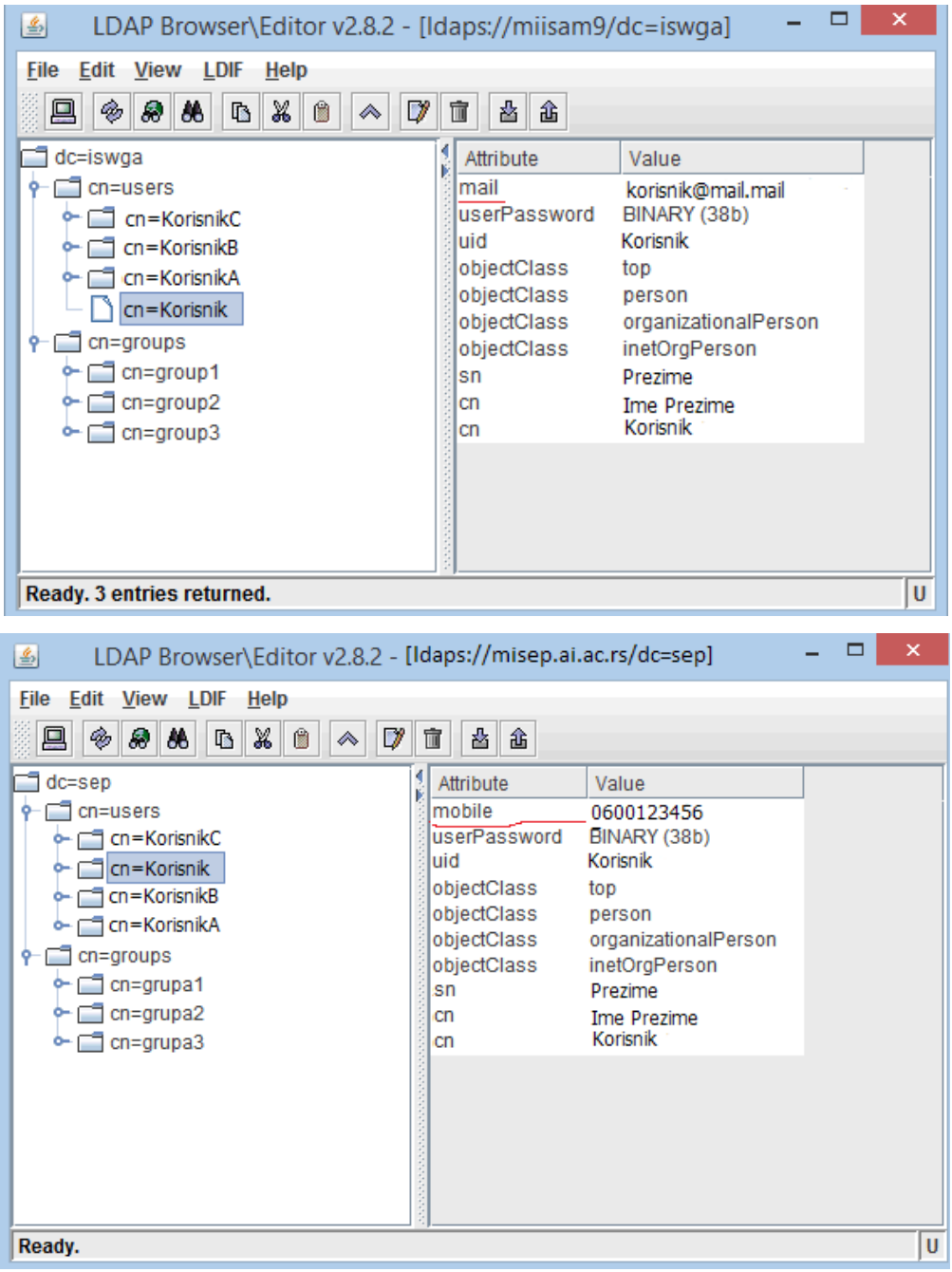


Fig. 6 User's interface LDAP browser on identity provider's page (left) and service provider's page (right)

page and when the user is logged onto the IdP application, by clicking on that link, he will automatically be transferred to the screen form on the service provider's page. The screen form in Figure 10 shows attributes that only exist on the identity provider's page (mail attribute, see Figure 10) will also be accessible on the service provider's page.

If the user is already logged into the identity provider's page and wants to access the service provider's page, the application will be transparent. However, if the user accessed an intranet web portal open to all users without the need for authentication and he then clicks on the link that can lead to the service provider's page in order to access their application, an authentication demand will pop up (since he has not yet been au-

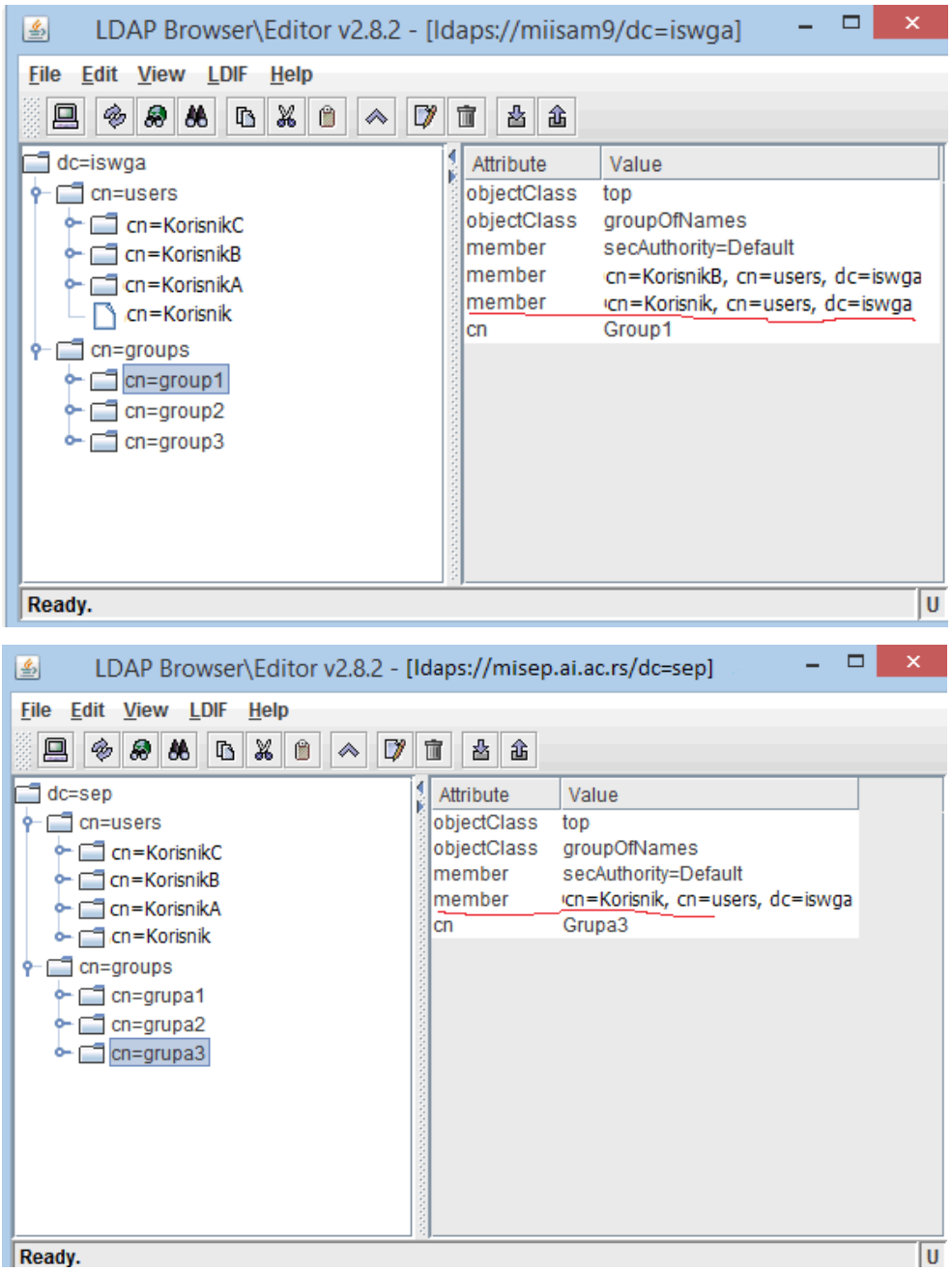
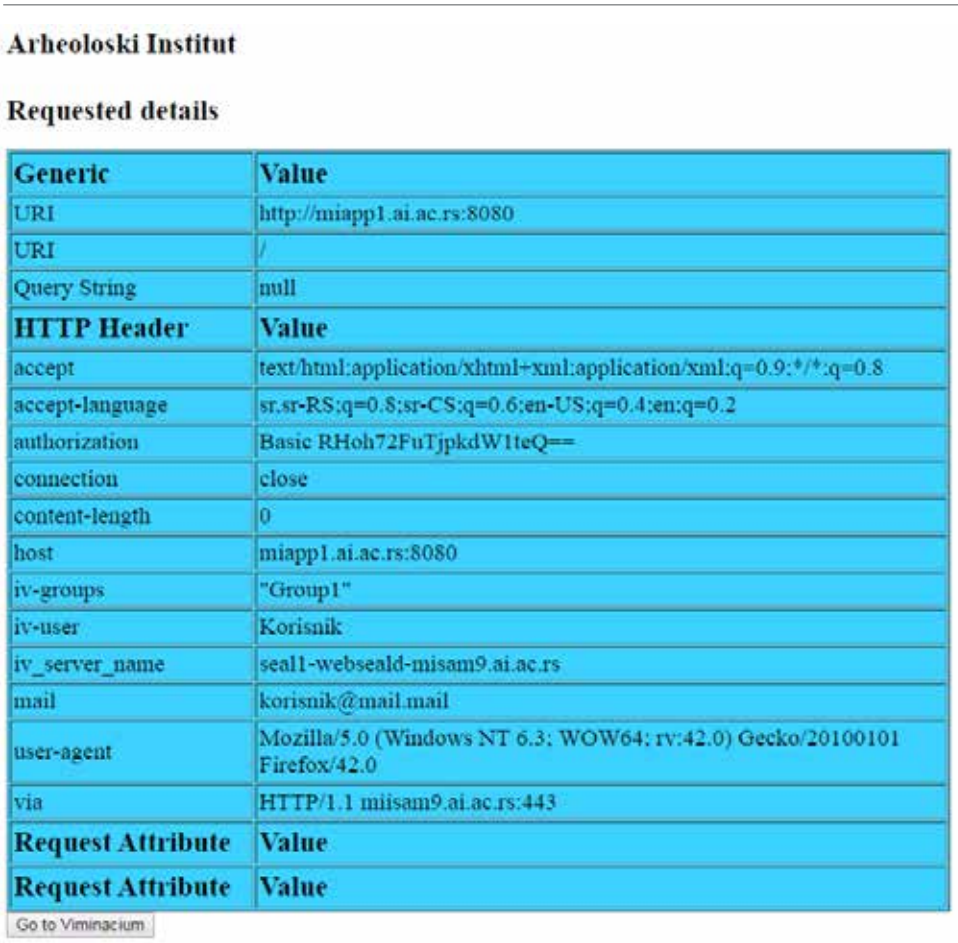


Fig. 7 Separating a user's attributes and group memberships

thenticated) on the identity provider's page (with the password for the identity provider). Only then will the application on the service provider's page be shown transparently.

In order to make this all function in a test environment, an IBM infrastructure has been created with two instalments of ISAM9 virtual appliance, one of them on the identity provider's page and the other on the service provider's page. ISAM9 is a modular appliance and it comes by default only with basic functionalities. In order to reach a federated environment, a licence is needed for a federation module, since it understands SAML 2 protocol. There is also an additional advanced access control module used for authentication with mobile web devices and for risk based authentication when access is performed over an insecure channel with a dynamic evaluation to determine



Arheoloski Institut

Requested details

Generic	Value
URI	http://miapp1.ai.ac.rs:8080
URI	/
Query String	null
HTTP Header	**Value**
accept	text/html;application/xhtml+xml;application/xml;q=0.9;*/*;q=0.8
accept-language	sr,sr-RS;q=0.8;sr-CS;q=0.6;en-US;q=0.4;en;q=0.2
authorization	Basic RHoh72FuTjpkdW1teQ==
connection	close
content-length	0
host	miapp1.ai.ac.rs:8080
iv-groups	"Group1"
iv-user	Korisnik
iv_server_name	seal1-webseald-misam9.ai.ac.rs
mail	korisnik@mail.mail
user-agent	Mozilla/5.0 (Windows NT 6.3; WOW64; rv:42.0) Gecko/20100101 Firefox/42.0
via	HTTP/1.1 miisam9.ai.ac.rs:443
Request Attribute	**Value**
Request Attribute	**Value**

Go to Viminacium

Fig. 8 User's application on identity provider's page.

whether it will allow access to a user or not (it is statically based on roles ascribed to it). In the production environment for the mentioned scenario, it is not necessary to put the ISAM9 appliance on the service provider's page, but in that case applications need to be made to the service provider side in the form of SAML 2 tokens, and also a change needs to be made in the configuration.

If there is a single-sign-on access through a web browser, communication is made to both systems via https. Then, they agree on which encryption will be applied, supported both by the web browser and by WebSEAL. This means that the browser can be re-configured in order to prevent connection to a weak https algorithm.

In addition, on the login page, it is possible to execute different methods of authentication. One is with the help of the user's name and password, while another is with the help of an external identification provider (EIP) that is able to define several types of authentication. This further implies that the already mentioned step-up authentication can be created, in which it is possible to define the provision of certain rights if the user accessed via a specific method. IBM within ISAM9 has a set of supported standards related to authentication mechanisms (double-factored, biometry and smart cards). Depending on the method of logging in, specific rights are provided. This represents a part of the WebSEAL configuration, since it receives the authentication demand and forwards it to the backend application.

Attribute mapping is performed within WebSEAL. There is a customising possibility when, instead of sending just the regular attributes, additional ones are sent, such as CN, SN and mail.

Viminacium

Requested details

Generic	Value
URI	http://sepapp1.viminacium.rs:8080
URI	/
Query String	null
HTTP Header	**Value**
accept	text/html;application/xhtml+xml;application/xml;q=0.9;*/*;q=0.8
accept-language	sr,sr-RS;q=0.8;sr-CS;q=0.6;en-US;q=0.4;en;q=0.2
cn	NOT_FOUND
connection	close
content-length	0
host	sepapp1.viminacium.rs:8080
iv-groups	"Groupa3"
iv-user	Korisnik
iv_server_name	seal1-webseald-misep.turorg.co.rs
mail	NOTFOUND
mobile	0600123456
user-agent	Mozilla/5.0 (Windows NT 6.3; WOW64; rv:42.0) Gecko/20100101 Firefox/42.0
via	HTTP/1.1 miisam9.ai.ac.rs:443
Request Attribute	**Value**
Request Attribute	**Value**

Figure 9. User's application on service provider's page.

These pieces of information are packed into SAML2 while on the other side what will be received is defined.

In addition, ISAM9 also contains a policy server in which polices can be made for certain applications set behind the mentioned junction.

The administration of the appliance itself is also possible, from the command line environment (with secure socket shell connection)

Securing the LDAP environment comes after configuring the runtime component of ISAM9. It is also possible to choose which type of LDAP will be used (external or embedded). After defining the LDAP type, ISAM9 makes its own specific suffix named **secAuthority=default**. It contains its specific attributes mapped to the users. After creating the new suffix or user, it maps them all onto its specific security LDAP for the needs of the access manager. These and the **secAuthority=default** could exist in their own local LDAP or in another LDAP, for example in an Active directory, Open Ldap or IBM directory server initiated on some other machine. The security suffix secAuthority can be set to be in the local LDAP and the ordinary suffix in some other directory. Regarding the administration of the local LDAP, there is a separate interface for the access manager that has a specific user who performs the administration

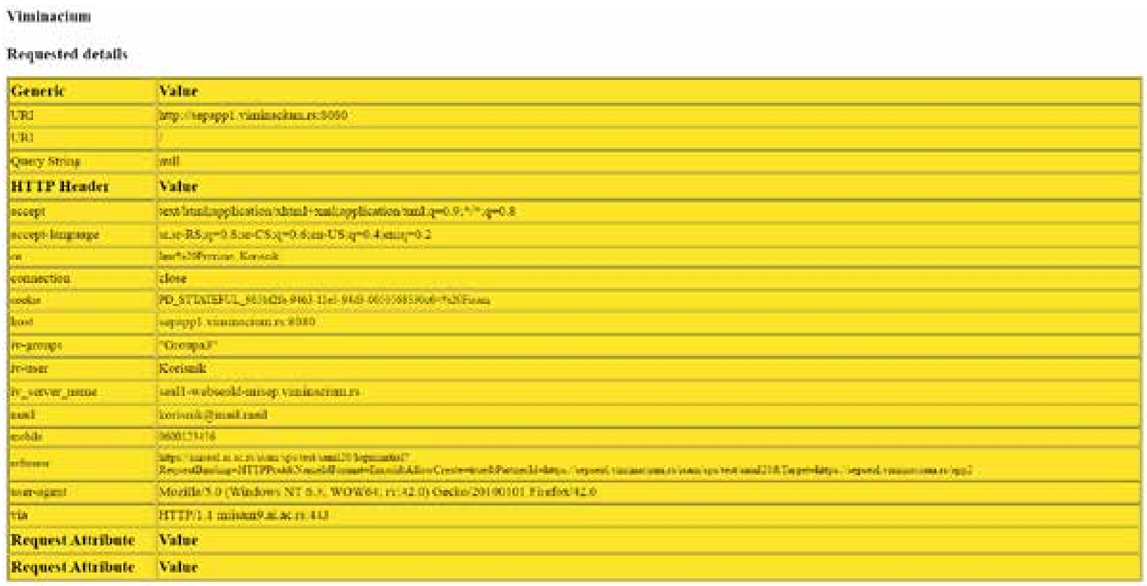


Figure 10. Screen form on the service provider's page

tasks. System users can only see each other in the secAuthority default**.**

The web space can additionally be protected using access control lists. For example, behind the reverse proxy there is the application app1. On the application app1, an ACL (access control list) can be added. The benefit gained is that all the authenticated users from ACL can have access to the app1 application, or it can also be set that access to the app1 application access is possible only from a specific ACL.

When it comes to protocols for creating federations, ISAM9 supports SAML2 and Openid, while TFIM has some more protocols for federation. If there is a need for some other federation, the ISAM9 licence includes a TFIM gateway. Such software actually represents an add-on that is installed onto the operating system, actually to the application server in order to broaden the ISAM9 functionality.

AUTHORISATION SETTING

WebSEAL itself secures the authorisation decision to be delegated by the application onto the reverse proxy server. As already mentioned, this proxy possesses in itself a definition of the protected web space. It is possible to define URL addresses that are subject to this protection. There is a root application that is placed on the specific URL address and then servlet calls and arguments are added onto the URL in order to start specific application functionality. If within such a defended space it is defined that according to a specific criterion (eg. joker signs or absolute routs etc...) something is allowed to pass through the web proxy, it will not be able to reach the application and be executed. By this alone, the decision is delegated from the application to the web proxy level as to whether the http request initiated by a client will reach the application, be initiated and return as a response, or if the proxy will return the response as "forbidden". Given the mentioned facts, it should be understood that the application is protected at the WebSEAL level and not within the application itself. As a security measure, it is also necessary to periodically perform vulnerability scanning and analysis of the server systems within an organisation in order to find and remove them in time. By detecting system vulnerability, recommendations are obtained for overcoming such security problems. (Korać, Prlja and Diligenski 2016).

CONCLUSION

The federated identity concept is most useful when there are business subjects with a legal relationship between each other. Onto this, an informatic aspect is imposed with the help of the informatic infrastructure. It secures that once the user's identification is performed on one of the identities, the same procedure is not repeated on another identity, since it can rely upon a trustworthy subject that can deliver the user's identity and that such an identity can be trusted. Regarding this, two sides can be distinguished, one of them being the identity provider, actually the one that secures the business procedure of establishing the user's identity, while the other is service provider, offering a business service, but fully relying upon the provider's identity as a base for recognising a user. In other words, the IdP represents the business entity in charge of user registration and authentication and issues confirmation of the established identity to other business entities. On the other hand, the SP represents the business entity that offers services to users (e.g. access to business applications), but it does not establish their identity, since it relies upon certificates issued by the IdP. Archaeological sites can be regarded as independent entities with their individual information systems. The services accessible to employees in the information systems and the services for end users that can be accessed from archaeological sites are wide ranging. Here, this refers to a digital database, virtual site visits, usage of video cameras for sightseeing, usage of cameras for video surveillance, streaming of events organised at some sites, and live transmission of concerts, operas and music events. Additionally, these services include observing archaeological excavations via video cameras, access to video conferences, souvenir sales related to a specific period (prehistory, Roman, Middle Ages, etc), a library database, exchange of library material, announcement of individual or group visits, ticket purchases and education programs. If the number of sites (Sirmium - Sremska Mitrovica, Singidunum - Beograd, Viminacium - Kostolac, Diana - Karataš, Felix Romuliana - Zaječar, Negotin - Šarkamen, Naissus - Niš, Iustiniana Prima - Caričin grad, Vinča, Lepenski Vir, Kale Krševica, Slatina near Paraćina) is added to these services, it becomes clear that the concept of federated identities would centralise and simplify identity exchange, at the same time offering secure access to servers after registration on the common identity provider, the Institute of Archaeology.

* * *

BIBLIOGRAPHY

Novičić, D. and Mitić, M. 2015
IBM Security Access Manager 9.0 (ISAM9) - *Identity Federation scenariji* MI SANU, prezentacija, SBS (Serbian Business Systems), December 2015.

Guruprasad, S. and Rajesh, B. 2012
Design and implement just-in-time provisioning with SAML 2.0, IBM Corporation, Developer Works, 2012, [e-book] https://www.ibm.com/developerworks/library/se-jitp/ [accessed February 12th, 2016]

Korać, V., Prlja D. and Diligenski, A. 2016
Digitalna forenzika, monografija, Institut za uporedno pravo, CNT, Arheološki institut, ISBN 978-86-87271-34-0, Beograd, 2016.

IBM Tivoli Software 2016
Understanding WebSEAL junctions, *[e-book],* https://publib.boulder.ibm.com/tividd/td/ITAME/SC32-1359-00/en_US/HTML/am51_webseal_guide16.htm, / [accessed August 25th, 2016]

Andronache, I. and Nisipasiu, C. 2011
Web Single Sign -On Implementation Using the SimpleSAMLphp Application, Journal of Mobile, Embedded and Distributed Systems, pp 21-29, vol. III, no. 1, 2011, *[e-book],* http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.301.6077&rep=rep1&type=pdf

Ping Identity 2016
Using Just-in-Time Provisioning, *[e-book]*, https://documentation.pingidentity.com/display/PF70/Using+Just-in-Time+Provisioning/ [accessed July 20th, 2016]

IBM knowledge center 2016
SAML usage scenario, *[e-book],* http://www.ibm.com/support/knowledgecenter/SSAW57_8.5.5/com.ibm.websphere.nd.iseries.doc/ae/cwbs_samlusagescenarios.html [accessed September 15th, 2016]

Gemalto 2016
SafeNet Authentication Service Integration Guide- Using SafeNet Authentication Service as an Identity Provider for IBM Security Access Manager (ISAM) for Web 9.0, *[e-book],* https://kb.safenet-inc.com/resources/sites/SAFENET/content/live/TECH_NOTES/2000/TE2687/en_US/007-013542-001_SAS_%20IntegrationGuide_ISAM_for_Web_SAML_RevA.pdf [accessed September 12th, 2016]

IBM ISAM9 2015
Security Access Manager 9, Federation Configuration topics, *[e-book],* http://www-01.ibm.com/support/docview.wss?uid=swg27046801&aid=8 [accessed December 28th, 2015]

REZIME

KONCEPT FEDERATIVNOG IDENTITETA IZMEĐU ARHEOLOŠKOG INSTITUTA I LOKALITETA U SRBIJI NA PRIMERU VIMINACIJUMA

KLJUČNE REČI: FEDERATIVNI IDENTITET, SIGURNA IDENTIFIKACIJA, PROVAJDER IDENTITETA, PROVAJDER USLUGA, SSO, ISAM9, TFIM, IDP, WEBSEAL.

U radu je prikazan koncept federativnog identiteta između Arheološkog instituta i arheoloških lokaliteta u Srbiji a na primeru Viminacijuma. Na taj način se obezbeđuje da, ukoliko je jednom sproveden postupak identifikacije korisnika od strane jednog identiteta, eliminiše se potreba da drugi identitet sprovodi tu istu proceduru za svaki lokalitet, već se oslanja na poverenje da prvi subjekt može da isporuči korisnički identitet kome se može verovati. Kao rezultat ovakvog pristupa identifikovani korisnik kod provajdera identiteta „Arheolški institut“ automatski će biti prepoznat kod provajdera usluga na bilo kom arheološkom lokalitetu u Srbiji, u konkretnom slučaju na primeru lokaliteta Viminacijum. Na taj način zaposleni u Arheološkom institutu bi imali omogućen pristup servisima (na primer bazi digitalne građe) na arheološkom lokalitetu Viminacijum nakon uspešne identifikacije zaposlenog od strane provajdera identiteta „Arheološki institut“.