# LDAPRoam: A Generic Solution for Both Web-Based and Non-Web-Based Federate Access

Qi Feng, Wei Peng

# LDAPRoam:A Generic Solution For Both Web-Based And Non-Web-Based Federate Access

Qi Feng[1][0000-0001-8595-9533] and ✉Wei Peng[1][0000-0001-7183-3374]

1 East China Normal University, 3663 N. Zhongshan Rd, Shanghai,China
{qfeng,wpeng}@admin.ecnu.edu.cn

**Abstract.** Identity federation technology has been widely used in recent years. But the solution for federate access is totally different between the Web-Based and non-Web-Based scenarios. Furthermore, it is highly limited for lack of support from non-Web-Based scenarios now. This paper proposes a generic federate access solution based on LDAP roaming, which can provide reliable identity roaming in any internet service. To service providers, our solution is transparent and looks like a LDAP. The paper first presents the difficulties in realizing LDAP roaming and discusses offers solutions to the implementation of LDAP roaming. Then it evaluates the easy integration and usability of LDAP roaming. Finally it compares the Generic Solution with the existing federal access solution.

**Keywords:** Identity Federation, Non-Web-Based, LDAP, SAML, eduroam.

## 1 Introduction

A consensus of resource sharing based on identity federation has been gradually reached[1]. However, the existing solutions for federate access, such as SAML[2] based on Web-based, cannot be applied under non-Web-Based scenarios. Although SAML can use the ECP mode to support applications on Non Browser, it is still limited for reliance on session and working on HTTP. The same is true of the case of the AAA-based identity federation, for example, eduroam[3], which does not depend on web, still asks for EAP to send authentication and accounting messages. Due to the lack of attributes, AAA-Based identity federation only appears in eduroam, but is hardly applicable under non-Web-Based scenarios, such as the console access(e.g., via SSH[4]) common in HPE.

Besides, the user experience of the two federate accesses is completely different. In SAML, users need to select their own home IdP on discovery service and input the username and password to finish the access, while in eduroam they should take user@domain as the username. In fact, the federations based on the two federate accesses are also different. In China, SAML Federation——CARSI contains 77 IdP members, almost comes from school libraries, while eduroam contains 235 IdP members, almost all of whom come from the network center or information technology center. This causes bad user experience and is not what we have expected either.

In this paper, we introduce a generic solution suitable for both Web-Based and Non Web-Based federate accesses. We put the point on the coupling degree of federate access and service provider. The lower degree, the smaller differences for experience. If the authentication solution is completely transparent to the service provider, the latter cannot perceive the existence of federation authentication for there is no difference between Web-Based and Non-Web-Based service and the authentication experience will be consistent.

Taking eduroma as a model, users can add @domain as a suffix after the username, i.e. username@example.org, to roam to their home organization LDAP[5] and finish the LDAP authentication process, which is transparent to the service provider. The roaming LDAP federation can be regarded as a virtual LDAP which is shown in Fig.1. The service provider does not need to participate in the details of the federate access; it only needs to support the LDAP protocol. In view of the extensive support of LDAP protocol, this solution will be very friendly to the service provider and will be easy to integrate.

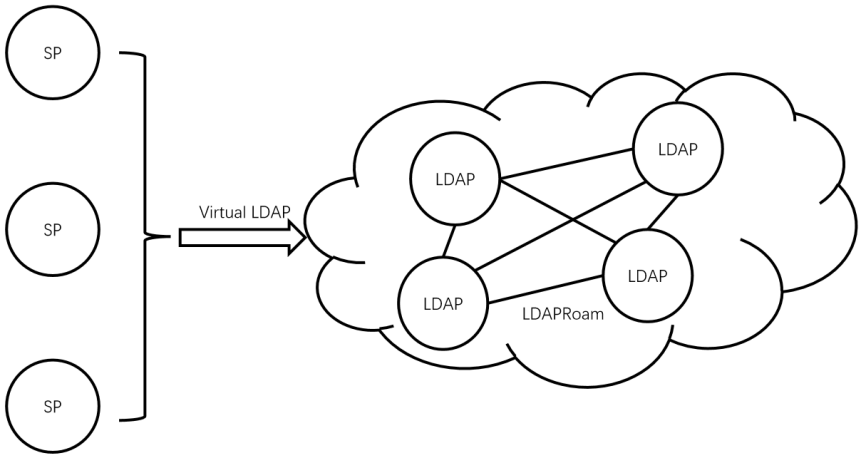


**Fig. 1.** LDAPRoam As A Virtual LDAP

It looks similar to the radius proxy structure of eduroam. Considering the particularity protocol of LDAP, the coincident attribute in the federation, and privacy protection of user information, we choose a net structure similar to SAML federation instead of the tree structure of eduroam federation. We also use the asymmetric encryption technology to form the trust relationship. It allows encrypting passwords by using the public key, which can help to prevent service providers from obtaining any plaintext of the password. We construct an independent service, LDAPRoam, as a proxy for the actual LDAP at the back, which can forward authentication requests of roaming.

The contributions of this paper are:

- The concept of LDAP roaming. This Identity federation is transparent to the service provider and supports identity roaming for internet services anywhere.

- The architecture of LDAP roaming. This solves the difficulties of LDAP protocol application in federate access.
- Evaluated LDAPRoam from the perspective of usability and easy integration, and compared it with the existing federate access solutions.

## 2 Related Works

There are many Web-Based identity federation technologies, such as SAML, OAuth[6], and OpenID[7]. Most of the time, they all rely on browsers. In Shibboleth project (SAML-Based), ECP[8] (Enhanced Client or Proxy) is proposed to work in a non-browser environment. However, due to the ECP needs supported by client modification, there is little "real world" support other than Shibboleth.

In some specific Non-Web-Based scenarios, such as roaming of wireless net-works, the AAA-Based technology is an option. Eduroam is a case that has been widely used in educational and scientific research institutions. Users can roam the authentication back to their own organization when accessing the eduroam network. Eduroam uses hierarchical Radius architecture. The roaming authentication request are protected by EAP methods, such as PEAP/EAP-MS-CHAPv2. Although the accouting request can also carry attributes[9], the standard of Radius attributes is designed for network accounting, which cannot meet the authorization requirements in several general scenarios. So, the federate access solution coming from AAA-Based cannot be applied in scenarios other than network authentication.

ABFAB[10] is the outcome of a project named Moonshot. The project is created to serve the programs built on Non-Web-Based services. The solution of this project is to extend eduroam to support SAML assertion. So ABFAB requires that application clients must support GSS-API[11]. Although many protocols are already supporting GSS-API (e.g., via ssh, nfs, ftp), the application client is still asked for modifications that intrude too deep, in order to turn on the function. The goal of Moonshot is to push all these changes into a standard and require updating at the client-end, but this is obviously unrealistic in the short run. This means that there is little possibility for ABFAB to be implemented at present.

FedKERB[12] and ABFAB have similar structures. FedKERB adds a KDC component in order to support Kerberos, which makes it difficult to change things on the client. Therefore, this solution is also very hard to be promoted.

Jens Köhler's[13] work is similar to ours. They also proposed an LDAP-based solution to keep it transparent to the service provider. However, in its solution, the attribute is acquired through SAML ECP, which also requires that the IdP and application client must support ECP first. Therefore, there are still difficulties in popularizing and implementing this solution.

Our solution does not require any modification on the application client, and all identity privacy and passwords can be well protected through external plugins, thus providing the maximum possibility for promote solution.

## 3 Challenges

Because the authentication and authorization mode of LDAP is very different from Radius, there are many challenges in LDAP-based roaming.

1. It must allow users to input plaintext passwords indirectly. It also cannot make extra modifications on application clients.
2. A trust relationship must be established between LDAP when providing roaming services, to avoid any possible hijacking in the process.
3. LDAP authentication is usually divided into two steps: Search and Bind. This may cause the loss of roaming domain name information in the second Bind step.
4. Since Bind must occur after Search, there is an unauthorized Search behavior. If there is no restriction, it may result in a leakage of user information on Search step.
5. There may be different attribute categories among different LDAP nodes, which need to be standardized using some methods.

We will describe in detail how to overcome these challenges in Section 4.

## 4 Solution

As mentioned in Section 3, the first challenge is the security of transmission. In the radius proxy structure of eduroam, there are intermediate forwarding nodes. Though passwords are well protected by client-side encryption, attribute information (e.g., username) can still be obtained by intermediate nodes, into which additional attributes can even be inserted. In fact, we have no choice but to believe that the intermediate forwarding node can be trusted. We also need to prevent possible cheating from middleman, which requires encryption. Since eduroam architecture itself does not support encryption, it needs another way to assist, such as connecting the radius nodes through the GRE tunnel.

Therefore, we draw on the experience from the SAML federation structure, a mesh point-to-point interconnection structure. The whole federation maintains a main metadata, which contains the basic information and certificate information of each LDAPRoam node.

The certificate is issued by LDAPRoam through sending the private key. Fig.2 shows the differences between LDAPRoam and eduroam architectures.

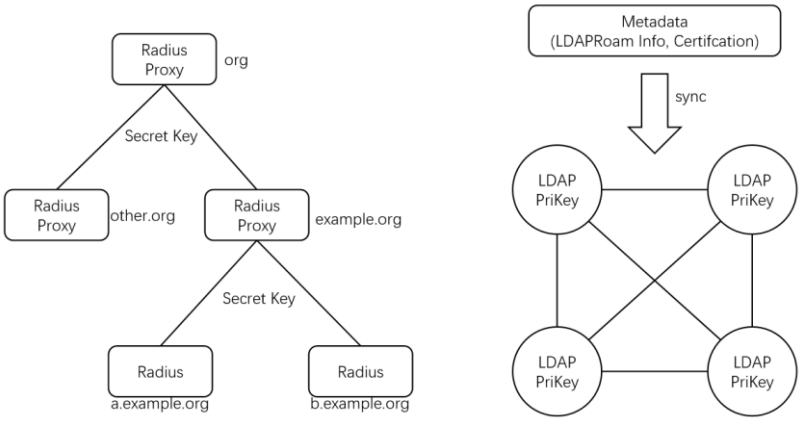


**Fig. 2.** Different structures with eduroam and LDAPRoam

Through the asymmetric encryption system of the certificate and private key within the mesh point-to-point structure, the security of the whole transmission is well guaranteed. Firstly, there is no intermediate nodes in the point-to-point structure and the data transmission path has been minimized. Secondly, the private key-certificate system based on asymmetric encryption can not only prevent the request message from being hijacked by the middleman through verifying the certificate, but also ensure that the request message comes from the trusted initiator by verifying the signature.

The second challenge is password protection. The password must be directly input from the service provider's client because there are no invasive changes taken by the service provider. When we are roaming in eduroam, we can directly type the password on the trusted operating systems (OS), because the OS usually uses mschapv2 encryption method to ensure that the password is securely encrypted at the beginning. But we cannot give the same trust to these third-party service provider clients. But in our mesh point-to-point structure, this problem is very easy to solve. We only need to encrypt the password through the public key of the other node. Then this password can only be decrypted after roaming to the destination. This process does not require any modification of the service provider's client. In fact, even if modified, it cannot be trusted as well. Using an external plugin is helpful to encrypt the password forms and keep user experience unchanged. The automatic filling of plugin forms has been verified by many password managers (e.g., via 1Password).

The third challenge is the authentication mode of LDAP. Unlike the AAA mode, which sends the user name and challenge message directly, LDAP uses DN(distinguished name) in Bind operation, while the username is an LDAP attribute, such as uid or sAMAccountName. Therefore, for an LDAP application, the standard practice is usually divided into two steps:

- First, search the DN of items by taking that username as the query condition of the attribute filter

- Second, bind with the DN and the password to verify whether the authentication is successful

Fig.3 shows the difference on authentication between AAA and LDAP

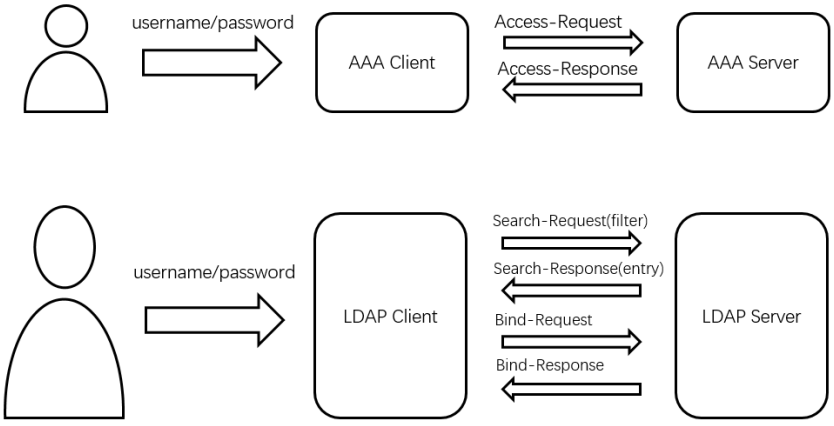


**Fig. 3.** Different Authentication Process with AAA and LDAP

In the second step of LDAP authentication, the DN information no longer contains the roaming domain name of the user. In order to forward the Bind-Request correctly to the appropriate node, we need maintain not only the correspondence between DN and LDAPRoam service but between domain and LDAPRoam service. For example, the roaming domain name of alice@a.example.org is example.org, and the DN record in LDAP of this account is cn=alice, dc=a, dc=example, dc=org. Then we need to map dc=a, dc=example, dc=org and a.example.org to the same LDAPRoam node. These information will be published by LDAPRoam to Metadata for other nodes to query.

The fourth challenge is the unauthorized LDAP Search. Due to the two-step nature of LDAP authentication, Bind must occur after Search. We cannot evade unauthorized LDAP Search behaviors. However, it is obviously inappropriate to return unauthorized user attributes to roaming LDAP. Our solution is to introduce the concept of authorization validity into LDAPRoam, and generate a cache record with valid authorization for users to have a successful Bind. The roaming party is allowed to query the user attribute within the validity period of authorization; otherwise only DN will be returned. This may lead to some abnormality of LDAP Client which does not meet the standards, but it has no influence on clients implemented according to the standards.

The fifth challenge is the standardization of attributes. The attribute standard among different LDAP may be completely different. Although there are a series of RFC standards for LDAP attributes, the understanding of the attribute fields may still be inconsistent. At least between OpenLDAP and Active Directory, the default field for the username is totally different. OpenLDAP usually uses uid, while Active Directory uses sAMAccountName. Moreover, since the details of LDAPRoam are transparent to the application, service providers have no way to adapt this attribute relationship to specific

LDAPRoam nodes. In SAML federation, IdP can map attributes into specific attribute names and oid strings when it queries them, which makes the attribute exchange within SAML follow the fixed standards. LDAPRoam also uses the same method of mapping attribute relationships by LDAPRoam nodes to shield attribute differences among different LDAP. Same as SAML2, LDAPRoam nodes protect the identity privacy of users and reduce unnecessary provision by releasing attributes when roaming in various nodes.

Fig.4 shows the overall architecture of LDAPRoam. The LDAPRoam node provides an LDAP-style interface for application clients, which makes authentication roaming transparent to the service provider. LDAPRoam uses RESTful API interface to simplify roaming message processing between each other. All LDAPRoam nodes synchronize Metadata information at a regular time to obtain basic information and certificate information of each node in the federation. When LDAPRoam queries the attributes from Backend-LDAP, the authoritative data source, it will make an attribute-map according to the attribute standard of the federation.

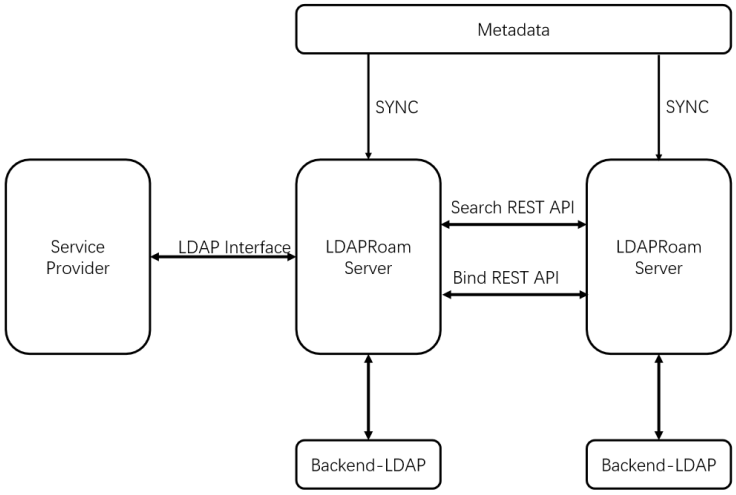


**Fig. 4.** LDAPRoam structure

Table1 shows the LDAPRoam field information provided in metadata.

**Table 1.** Metadata Field For LDAPRoam.

| Field | Value | Comment |
|---|---|---|
| **domain_name** | ldap.b.example.org | server domain name |
| **served_domain** | [“b.example.org”] | served domains |
| **base_dn** | “dc=b,dc=example,dc=org | The basedn |
| **certification** | “nLmIuZXhhbXB….” | The certification |
| **bind_endpoint** | "https://ldap.b.example.org/api/v1/bind" | Bind API endpoint |
| **search_endpoint** | "https://ldap.b.example.org/api/v1/search" | Search API endpoint |

## 5 Evaluations

Whether the solution can be effectively promoted depends on the simplicity of deployment. Our solution does not require any changes from the service provider. For identity providers, they only need to deploy a LDAPRoam service.

For service provider, we have two modes, full trust mode and limit trust mode.

- In the Full trust mode, it is safe to users to directly enter the username and password to service providers that are usually government agencies. A typical application scenario is that a fully trusted service provider encapsulates the interface again, such as oAuth2, in order to support other service providers to interface. This mode can well solve the federate access requirements in some cookie limited scenarios. For example, the embedded browser in Alipay cannot be used normally in SAML2, because the cookies are completely disabled. But the LDAPRoam solution encapsulated by oAuth2 can solve this problem well.
- In Limit trust mode, in order to protect our password security. We must first encrypt the password through a plugin or tool, and then submit the encrypted password to the client of the service provider.

We compared the differences between LDAPRoam and other federation access solution, and listed the differences in several dimensions, mainly based on the seven requirement and the level of support provided by Alejandro Pérez-Méndez[14]. Table2 lists the comparison with current mainstream solutions and Table3 lists the comparison with other experimental solution.

**Table 2.** Comparison Between Mainstream Federate Access solutions.

| Topic | LDAPRoam | Eduroam | SAML2(shibboleth) |
|---|---|---|---|
| **R1 – Authentication in the IdP** | Roam back to IdP | EAP and AAA proxy back to IdP | Redirect to IdP Portal |
| **R2 – High level authorization** | LDAP attribute search | Limit support with radius attribute | SAML2 assertion from the IdP |
| **R3 – Data transport security** | Asymmetrical encryption, point to point | EAP Tunnel, with intermediate node | Asymmetrical encryption, point to point |
| **R4 – Single Sign On** | Not Support | Not Support | Support |
| **R5 – Re-use of instructures and standards** | Based on LDAP ,TLS and RSA | Based on EAP and AAA | Based on SAML2 |
| **R6 – Usability** | username/password | username/password | username/password |
| **R7 – Identity Privacy** | Attribute map control | EAP Tunnel and pseudonyms | Attribute map control |
| **Web-Based** | Support | Not Support | Support |
| **Non Web-Based APP(Full trusted)** | Send password directly | Password encrypt, such as mschapv2 | Must modify the client |

| | | | |
|---|---|---|---|
| **Non Web-Based APP(limit trusted)** | Send password encrypt | Password encrypt, such as mschapv2 | Must modify the client |

**Table 3.** Comparison Between Experimental Federate Access solutions.

| Topic | LDAPRoam | ABFAB | FedKERB |
|---|---|---|---|
| **R1 – Authentication in the IdP** | Roam back to IdP | EAP, AAA and GSS-API | EAP, AAA and GSS-API |
| **R2 – High level authorization** | LDAP attribute search | SAML2 assertion from the IdP | SAML2 assertion from the IdP |
| **R3 – Data transport security** | Asymmetrical encryption, point to point | EAP Tunnel and RadSec | EAP Tunnel and RadSec |
| **R4 – Single Sign On** | Not Support | Not Support | Based on Kerberos |
| **R5 – Re-use of instructures and standards** | Based on LDAP ,TLS and RSA | Based on EAP, AAA and GSS-API | Based on EAP,AAA,GSS-API and Kerberos. |
| **R6 – Usability** | username/password | username/password | username/password |
| **R7 – Identity Privacy** | Attribute map control | EAP Tunnel and pseudonyms | EAP Tunnel and pseudonyms |
| **Web-Based** | Support | Support | Support |
| **Non Web-Based APP(Full trusted)** | Send password directly | Must modify the client | Must modify the client |
| **Non Web-Based APP(limit trusted)** | Send password encrypt | Must modify the client | Must modify the client |

LDAPRoam can provide good support in all other dimensions except SSO. Compared with the current mainstream solutions, LDAPRoam combines the advantages of eduroam and SAML2, and can support both web-based and Non Web-Based applications with the same user experience. Compared with other experimental solutions, LDAPRoam does not require service provider clients to make any modifications, which is very helpful for practical application and promotion.

Taking HPC as an example, it is usually necessary to provide federation access on web-sites and give support for users' console access (i.e. ssh). As an HPC service with a long history, it is obviously unrealistic to require all users to upgrade their SSH clients in order to provide federate access. Now the solution of LDAPRoam can be operated and implemented easily without damaging the users' privacy interests.

## 6 Conclusion

In this paper, we introduce the concept of LDAPRoam, which combines the advantages of SAML2 alliance and eduroam alliance. We discuss many challenges that the model faces, and give solutions by designing and adapting the particularity of the LDAP protocol. After evaluation, the roaming solution of LDAPRoam is highly suitable for de-

ployment and promotion. We have already implemented the experimental roaming verification between East China Normal University and the Information Center of Shanghai Municipal Education Commission through LADPRoam. It is planned to carry out the promotion step by step. In a wide range of situations, the application still needs further observation and verification.

## References


1. Torres, J., Nogueira, M., Pujolle, G.: A Survey on Identity Management for the Future Network, IEEE Communications Surveys & Tutorials, vol. 15, no. 2, pp. 787–802(August 2013).
2. Cantor, S., Kemp, J., Philpott, R., Eve, M.: Assertions and Protocols for the OASIS Security Assertion Markup Language (SAML) v2.0, OASIS Standard(March 2005).
3. Wierenga, K., et al.: Deliverable DJ5.1.4: Inter-NREN Roaming Architecture. Description and Development Items, GN2 JRA5. GEANT2(September 2006).
4. Ylonen, T., Lonvick, C.: The Secure Shell (SSH) protocol architecture, IETF RFC 4251(January 2006).
5. Sermersheim, J.: Lightweight Directory Access Protocol (LDAP): The Protocol, IETF RFC 4511(June 2006).
6. Hardt, D.: The OAuth 2.0 Authorization Framework, IETF RFC 6749(October 2012).
7. OpenID Connect Core 1.0 incorporating errata set 1, https://openid.net/specs/openid-connect-core-1_0.html, last accessed 2014/11/08.
8. ECP-Shibboleth Concepts, https://wiki.shibboleth.net/confluence/display/CONCEPT/ECP, last accessed 2016/04/05.
9. Rigney, C.: RADIUS Accounting, IETF RFC 2866(June 2000).
10. Application Bridging for Federated Access Beyond Web (ABFAB) IETF Working Group, https://datatracker.ietf.org/wg/abfab/charter/, last accessed 2016/09/30.
11. Linn, J.: Generic Security Service Application Program Interface Version 2, Update 1, IETF RFC 2743(January 2000).
12. Pereniguez, F., Marin-Lopez, R., Kambourakis, G., et al.: PrivaKERB: A user privacy framework for Kerberos, Computers & Security, vol. 30, no. 6/7, pp. 446–463(September 2011).
13. Köhler J., Simon M., Nussbaumer M., Hartenstein H.: Federating HPC Access via SAML: Towards a Plug-and-Play Solution. In: Kunkel J.M., Ludwig T., Meuer H.W. (eds) Supercomputing. ISC 2013. Lecture Notes in Computer Science, vol 7905. Springer, Berlin, Heidelberg(2013).
14. Perez-Mendez, A., Pereniguez-Garcia, F., Marin-Lopez, R., et al.: Identity Federations Beyond the Web: A Survey, IEEE Communications Surveys & Tutorials, vol. 16, no. 4, pp. 2125-2141(May 2014)