International Journal of Cyber-Security and Digital Forensics (IJCSDF) 7(3): 275-282
The Society of Digital Information and Wireless Communications (SDIWC), 2018 ISSN: 2305-001

Comparison of Attribute Based Access Control (ABAC) Model and Rule Based Access (RBAC) to Digital Evidence Storage (DES)

Moh Fadly Panende[1], Yudi Prayudi[2], Imam Riadi[3]
[1,2]Department of Informatics, Universitas Islam Indonesia
[3]Department of Information System, Universitas Ahmad Dahlan
[1]16917111@students.uii.ac.id, [2] prayudi@uii.ac.id, [3]imam.riadi@is.uad.ac.id

ABSTRACT

Digital Evidence Storage (DES) should have been established not only using simple authentication and authorization methods, namely authentication and authorization processes of username and password only, but also had to use more complex authentication and authorization processes by implementing rule policy as the determiner in request access. RBAC was an access control that worked based on user role, while ABAC was an access control model which its work principles was based on the attribute. Meanwhile, XACML is a programming language that specifies RBAC and ABAC policies using XML format. This research was aimed to make comparison toward two access control models which was acknowledged to be suitable for being implemented on DES. The implementation of ABAC is to increase DES security level. The final result of this research was the creation of a better access control model on DES system according to RBAC and ABAC models comparison.

KEYWORDS

Access Control Model, ABAC, RBAC, DES, XACML, XML

1 BACKGROUND

Digital evidence storage (DES) was a business model from parts that connected directly with digital evidence and information storage of digital evidence [1]. This concept was first introduced by [2] in his research related to digital evidence cabinets in which the research explained that digital evidence storage (DES) was a system made to a chain of custody (CoC) handling from every [3] digital evidence that was acquired. This concept was established on three approaches, namely: digital evidence management frameworks, digital evidence bag, and access control [4].

The access control model that existed in the DES is currently only a process of authentication and user authorization only using username and password. No other parameters are used except authentication and authorization. This, of course, will not be following the security aspects that exist in the DES.

As its principle, the access control was a mechanism to limit operation or action toward computer system for legitimate user only [5], access control on digital evidence or its resource, so far the issue handling of access control application on digital evidence was done by [6] Suggested concept was mechanism application of partial and full supervision to depict different rights and functions between investigator who directly handled digital evidence and other law enforcement that controlled the evidence use. Although there was other researcher about hierarchical access control from [7] however in that research, the application was toward the Cloud SAS environment using rule based access control model (RBAC). So, the continual research of access control on digital forensic environment had no other comparison yet. In fact, according to [8] access control was center of computer security. In the context of digital evidence integrity and credibility, a concept of access control was the most important factor to be concerned on.

In this research, there would be a comparison of two access control models which would be used as a solution on DES access control namely rule-based access control (RBAC) and attribute-based access control (ABAC). The implementation will be using Extensible Access Control Mark-Up Language (XACML) which was the standard of

OASIS used as a programming language for specifying RBAC and ABAC policy using XML format. The question of this research is to determine which access control that most suitable and the most proper to be used in digital evidence storage (DES). This research used functional testing method toward two access control models

2. LITERATURE STUDY

Access control on software was security mechanism that could give every entity has a status valid permit to do a specific action [9].
RBAC was used to analyze source and determine the user with permit policy in a software system. Besides, the research was done by [10] that analyzed the shortcoming of traditional RBAC and combined it with SOA characteristics, tried to solved authentication and authorization adjustment for a role by team and service as a turning point to guarantee safety access according to SOA system.

Research by [11] has discussed how to build data access control for cloud storage service using RBAC and attribute-based access control (ABAC) that was easy to use and manage. This research illustrates how to provide policies and facilitate flexible access control for data access service in cloud storage.

Research by [12] has given a solution to a control system problem of ABAC cross domain together with security domain as an attribute with a subject, object, authority, environment attribute as basic access to access decision making. Meanwhile, research which was done by [13] discussed an attribute real-time-based access control model that reflected their requirement of criticism application of two model namely Central Guard System and Physical Cyber of a medical system.

The next research was done by [14] that focused on the re-writing request that accepted inappropriate response by reducing necessary resource; they made a new model by utilizing framework XACML 3.0 to find out the most proper policy for every response request in four aspects namely subject, environment, and resource action for re-writing request.

Another research has been done by [15] that proposes a framework for Internet-based forensic logs that aims to assist in the investigation process to reveal DoS attacks. The framework in this study consists of several steps, among others: logging into the text file and database as well as identifying an attack based on the packet header length.

According to [16] there are four predefined attributes namely: subject, resource, action, and environment. However, user attribute type could also be applied for a specific application. XACML supported various kinds of data types, name types and path expressions for attributes, e.g., string, integer, internet-based names, regular expression, and XPath. In this attribute use, data type was more mainly specified than domain.

3 METHODOLOGY

The initial step of this research is identifying problem research to analyze the access control problems in DES. The next step is making literature study to align theories that related to research and do an implementation of the access control policy for the DES. The final step is doing the comparison method by using functional testing method for testing both types of access controls.

Figure 1. Research Methodology

4. IMPLEMENTATION AND RESULT

4.1 Policy Statement

The policy statement built should be based on the needs that must exist on the access control design and the needs of the system that will implement the access control. Based on the analysis of the DES system then the following table is a reference of the policy statement to be applied to the DES system.

Table 1. Policy Statement

Subject	Resource	Actions	Environment
First Responder	Upload Digital Evidence	Upload	IP Address Mac Address Time Access
	Create Cabinet	Create	

	Create Rack	Create	
	Create Bag	Create	
	Input Data Case Coc	Input	
Investigator	Download Digital Evidence	Download	IP Address Mac Address Time Access
	Complete The Data Coc	Complete Data	
Officer	Delete Digital Evidence	Delete	IP Address Mac Address Time Access
	Change Password User	Change Password	
	Validate Digital Evidence	Validate	
	Validate Case Status	Validate	
	Download Form Coc	Download	
	Change Code Signature	Change Code	
	Validate Data Coc	Validate	
Layer	Download Form Coc	Download	IP Address Mac Address Time Access

Table 1 explains that in a policy statement for DES consists of 4 pieces of subject that is the user positions are first responder, investigator, officer, and lawyer. Then consists of 15 types of resources as objects, 9 types of actions, and 3 types of environment which is the environmental conditions when the request is made.

The first subject is the first responder, the subject has the right to perform the resource operation, upload digital evidence, create cabinet, create rack, create bag, and input data case for chain of custody, as well as upload, create and input actions. The environment used for this first subject activity is ip address, mac address, and access time. The second subject is the investigator, this subject has only 2 access rights on the resource that is downloading digital evidence and complete the data of chain of custody with the actions of download and complete data and its environment is IP address, mac address, and access time. The third subject is the officer, the subject has access rights on the resource, delete digital evidence, change user password, validate digital evidence, validate status ces, download chain form custody, change code signature, validate data chain of custody. Actions are owned by this subject is delete, change password, validate, download, change code. Environment attached to this subject is the IP address, mac address, and time access. The fourth subject is the lawyer, the subject has only one resource permission that is downloading the chain of custody form, and the download action, and the environment attached is IP address, mac address and access time.

The policy statement described in table 1 will be part of the components that will be included into the rule and will be a logical expression to satisfy every request made by the user.

4.2 POLICY RBAC OF DES

Referring to the research undertaken by [17] in the RBAC process the access permissions are only related to the roles and users that are members of the role so that the user obtains the role permission on the system. Here is an overview of the LPBD RBAC model.

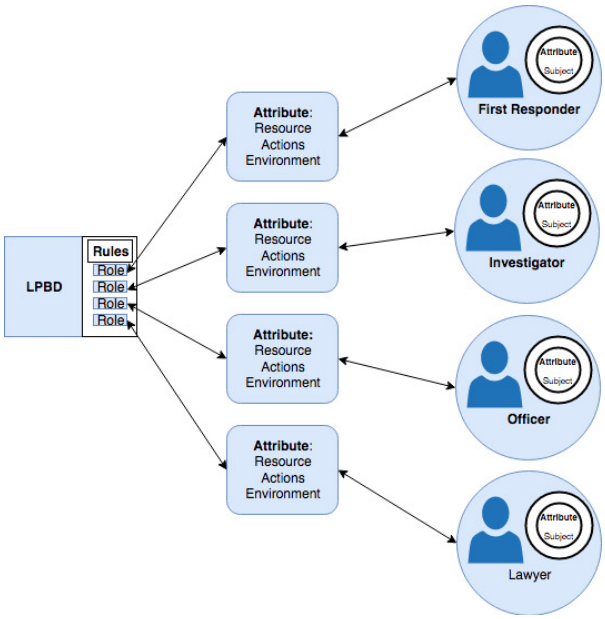

Figure 2. RBAC DES Model

Figure 2 explains that granting permissions on each user's access is based on the user's role. For example, user first responder will only be granted permissions when requesting when username and

password are input in accordance with the role has been entered.

The RBAC Policy design for DES was created using the UMU XACML Editor Version 1.3.2, tools. Here is the RBAC policy for DES

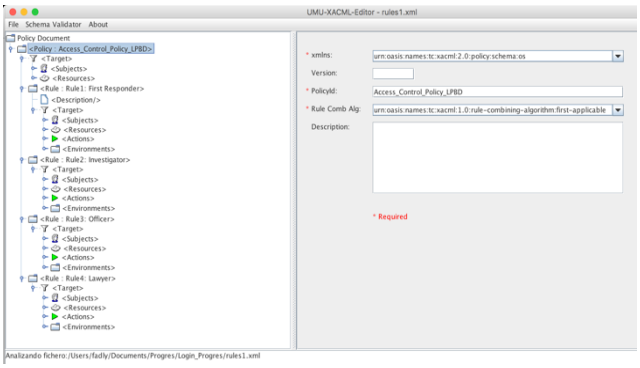

Figure 3. XACML Policy RBAC DES

Figure 3 explains that the policy that has been created using the rule based access control has 1 policy and 4 rule with each rule contains every policy rule given to each user. As has been explained by [14] that permit terminology is used in rules that are allowed to access and deny for rules that are not allowed to access.

In RBAC LPBD design this rule 1 contains subject: first responder, resource: upload, digital evidence, create cabinet, create rack, create bag, input data case of chain of custody, actions: upload, create, input, environment: IP address, mac address and time access. Rule 2, contains subject: investigator, resource: download digital evidence, complete the data of chain of custody, actions: download, complete data.

The next is rule 3 with subject is officer, resource: delete digital evidence, change user password, validate digital evidence, validate case status, download chain of custody form, change code signature, and validate data of chain of custody. Actions: delete, change password, validate, download, change code. Environment: IP address, mac address, and time access. And the last rule 4 with subject is lawyer, resource: download form coc, action: download, environment: IP address, mac address, and time access.

Figure 4. Output XACML Policy RBAC

The functional test on RBAC LPBD is done using one sample user first responder along with component elements of resources, actions, and environment used to test access control functionality. Figure 5 shows the results of functional tests that have been performed.

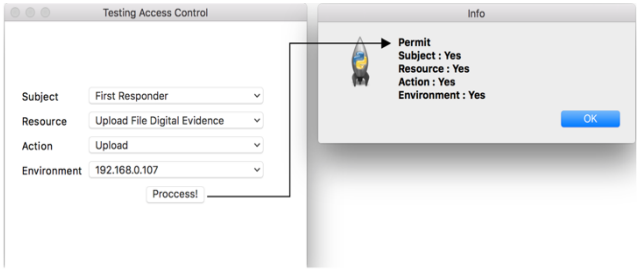

Figure 5. RBAC Functional Test

Figure 5 explains that the test results using user first responder and resource: upload digital evidence, then actions: upload, and environment ip address: 192.168.0.107 got permit result. This result is obtained because the role of subject, resource, actions and environment are included components that are in the appropriate rule.

4.3 POLICY ABAC DESC

Referring to research conducted by [19], the basic idea of ABAC is not to give permission as the output of a direct relationship between the subject and the object, but underlying the granting of the permission through the attribute of both. This is due to the ABAC system authorization element defined in the terminology attribute. The attribute itself is a characteristic of the entity defined before by those who have authority for it. The illustration is as shown in the figure 6.

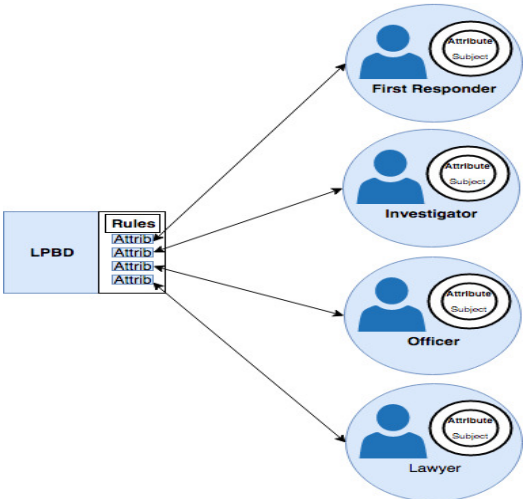

Figure 6. ABAC DES Model

Figure 6 explains that permission granting each user access is viewed based on the attribute attached to the user. For example, the user investigator will only be granted permissions when making a request if the username and password entered in accordance with the attribute has been pinned on the user. The ABAC policy of DES was created using tools UMU XACML Editor. Figure 7 shown the captured of ABAC policy for DES.

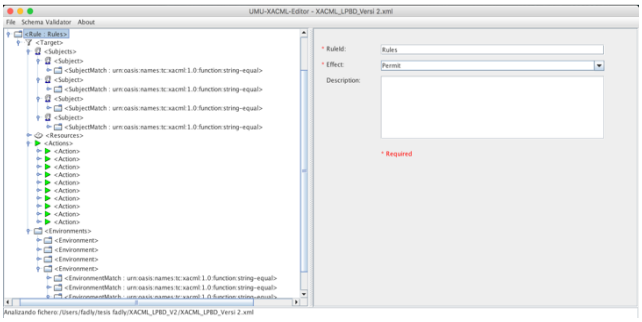

Figure 7. XACML Policy ABAC DES

Based on Figure 7, it can be explained that the policy on ABAC DES consists of 1 policy that contains the subject and resource attribute and 1 rule that contains the attribute of subject, resource, actions, and environment. This is in accordance with the basic concept of ABAC as an access control model that operates on the basis of attributes. Furthermore, based on the explanation of [5] there are 4 aspects of attribute in ABAC namely:

1. Subject is a human or non-human user (e.g. device or software component) that requesting access. Meanwhile, access requests can use individual attributes of the subject or combination to indicate a unique identity.
2. Resource is something of a protected target such as devices, files, records, tables, processes, programs, and networks.
3. Actions are executions of a function when requesting a subject against a resource.
4. Environment is a characteristic of operational or situational such as current time, current temperature, and IP Address.

Figure 8 is the output of XACML policy design using UMU XACML Editor Version 1.3.2.

```
- <Policy PolicyId="Access_Control_LPBD" RuleCombiningAlgId="urn:oasis:names:tc:xacml:1.0:rule-combining-algorithm:first-applicable">
  - <Target>
    + <Subjects></Subjects>
    + <Resources></Resources>
    </Target>
  - <Rule Effect="Permit" RuleId="Rules">
    - <Target>
      + <Subjects></Subjects>
      + <Resources></Resources>
      + <Actions></Actions>
      + <Environments></Environments>
      </Target>
    </Rule>
  </Policy>
```

Figure 8. Output XACML Policy ABAC

Figure 8 explains that the Policy Id, named access_control_policy, uses a rule combining algorithm first applicable. this policy only uses single algorithm with 1 policy effect that is permit. The deny condition will automatically occur when the access request made cannot meet the component attribute as a policy rule that has been created.

Functional tests performed on ABAC LPBD conducted using a sample user that is the investigator and other elements that become attributes of resources, actions, and environment to test the extent to which access control is running in accordance with the policy that has been designed. Figure 9 shows the results of functional tests that have been performed

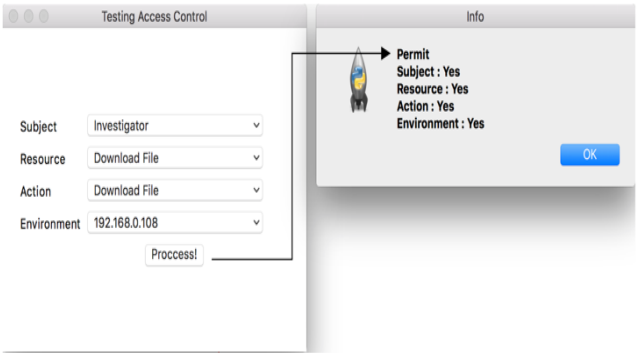

Figure 9 ABAC Functional Test

Figure 9 explains that the result of the test using subject investigator with resource: download file digital evidence, actions: download file, and environment ip address: 192.168.0.108 get permit result. This result is obtained because the resources, actions and environment included is an attribute that is pinned on the subject investigator. In this case each request will generate permit when the attribute is an attribute that attached to the subject.

4.5 Result

Based on the problems presented in the background of this research, there are problems in access control for pre-made DES systems that only implement authentication and authorization mechanisms of users only, and no other parameters that support more complex authentication and authorization processes. Therefore, the solution offered through this research is the use of two access control models namely RBAC and ABAC.

Based on the access control test results described in Sections 4 that RBAC and ABAC have been tested using specially testing tools to test access control performance, the results obtained that both access control models can run and function properly.

Table 2. Access Control Testing Result

Subject	Resource	Actions	Environment	Output
First Responder	Upload File Digital Evidence	Upload	Ip Address: 192.168.0.107	Permit
Investigator	Download File Digital Evidence	Download File	Ip Address: 192.168.0.108	Permit

Based on the data in the table, the most accessible access control model used on DES is the ABAC model, one of which is the more flexible ABAC characteristic. As mentioned by [20] that ABAC will be used more in terms of flexibility in terms of applying attributes to users. While in other studies conducted by [21] it also mentions that ABAC has a number of better features than models in the previous generation, one of the features is that ABAC allows grant access control through a combination of a number of attribute authorization elements such as: subject, resources, actions, and environments into an access control decision.

5 RELATED WORKS & DISCUSSION

From the test results, it can be seen that applying access control to DES will provide more complex options for managing the relationship between subject, resource, action and environment. It is not as simple as implementing authorization and authentication using a username and password only.

Research related to the solution of an access control model and performing a comparison between the RBAC and ABAC models applied to certain cases, was previously performed by [11]. The research proposed an RBAC role-based access control design that is compatible for cloud storage services and provides an easy-to-use and manageable ABAC mechanism. RBAC and ABAC have their own characteristics so for the needs of cloud storage, then introduced a combined solution through attribute based encryption (ABE).

Other related research is done by [22] they mentioned that the limitations of RBAC are now shifting towards the ABAC model to increase flexibility by using attributes outside of roles and groups. They propose an ABAC extension with the user attribute of the OpenStack Access Control (OSAC) model and demonstrate its implementation with the benefits of a policy machine (PM).

Medical record, personal health record, health care system is included in the category of secure critical system which is then used as research object by [23]. In his research xx conducted a series of activities of development and application of RBAC as one of the access control model it uses. Although it does not provide a framework as a solution, [23] argues that access control in the healthcare system must be dynamic and adaptable to support unpredictable patient activity. The same issue on access control issues to electronic health records is also discussed by [24] through the RBAC and ABAC approaches. In this case xx then develops a permission model named tests confidentiality model (TCM). The model specifications developed are built and tested using

the B-Methods tool, a tool to build and test formal method specifications. Another study on the control of the use of data in the field of health is also carried out by [25], in this case a schema solution called Ciphertext-Policy Attribute Based Sign-cryption (CP-ABSC) has been proposed. This solution is the use of digital signatures and encryption to ensure the fulfillment of the nature of confidentiality, authenticity, unforgeability, anonymity and collusion resistance from personal health record usage.

It appears that most of the ABAC implementation research is in the health field system. This is because the subject, resource in the health sector including the very sensitive so that the need handling access to a fairly complex resource, not only limited to the application of user authorization and authentication. Access control research in the field of health in principle has similarities with access control to digital evidence, unfortunately the issue access control on digital evidence has not been a major study in the field of digital forensics or digital evidence.

6. CONCLUSION & FUTURE WORKS

Based on the explanation of the comparison of RBAC and Attribute Based Access Control (ABAC) models in the Digital Evidence Storage (DES) described earlier, it can be concluded that both access control models of RBAC and ABAC are applicable for DES. Both access control model are the solutions of such as limited authorization and user authentication that has been applied to the DES system before. The functional tests that have been carried out using special tools designed to test RBAC and ABAC show that the access control model that is suitable to be applied to DES is ABAC model.

Subsequent research in the development of access control model on DES can be done by considering the following factors: the two access control models are not equipped with XACML design schema test; therefore, it is necessary to do research related to existing XACML schema. In addition to testing the schema it is also necessary to validate the XACML design that has been created. The next research can also accommodate the input from the design of access control in the health field, whether the various access controls that have been applied in health field can also be applied or adapted for the DES environment.

REFERENCES

1. K. Widatama, "Konsep Lemari Penyimpanan Bukti Digital Menggunakan Struktur Bahasa XML," *Semin. Nas. Inform. dan Apl. ke-3 dengan tema "Digital Evid. Comput. Crime,"* p. 23, 2017.
2. Y. Prayudi, A. Ashari, and T. K. Priyambodo, "Digital Evidence Cabinets : A Proposed Frameworks for Handling Digital Chain of Custody," *Int. J. Comput. Appl.*, vol. 109, no. 9, pp. 30–36, 2014.
3. I. Riadi and R. Umar, "Identification Of Digital Evidence On Android ' s," vol. 15, no. 5, pp. 3–8, 2017.
4. M. P. Aji, I. Riadi, and A. Lutfhi, "The digital forensic analysis of snapchat application using XML records," *J. Theor. Appl. Inf. Technol.*, vol. 95, no. 19, pp. 4992–5002, 2017.
5. R. Sandhu, "Security Models : Past , Present and Future," no. August. Institute for Cyber Security, UTSA USA, San Antonio, TX, USA, pp. 1–28, 2010.
6. C. Hsu and Y. Lin, "A Digital Evidence Protection Method with Hierarchical Access Control Mechanisms," in *IEEE International Carnahan Conference on Security Technology (ICCST)*, 2011, pp. 1–9.
7. D. Li, C. Liu, B. Liu, and DUMMY, "H-RBAC: A Hierarchical Access Control Model for SaaS Systems," *Int. J. Mod. Educ. Comput. Sci.*, vol. 3, no. 5, pp. 47–53, 2011.
8. W. Stallings and L. Brown, *Computer Security : Principles and Practice*, 3rd Editio. USA: Pearson Education International, 2015.
9. T. H. Pham, N. T. Truong, and V. H. Nguyen, "Analyzing RBAC security policy of implementation using AST," *KSE 2009 - 1st Int. Conf. Knowl. Syst. Eng.*, pp. 215–219, 2009.
10. Z. Qu and N. Meng, "Design and implementation of the RBAC-SOA model," *Int. Conf. Signal Process. Proceedings, ICSP*, pp. 2948–2951, 2008.
11. Y. Zhu, D. Huang, C.-J. Hu, and X. Wang, "From RBAC to ABAC: Constructing Flexible Data Access Control for Cloud Storage Services," *IEEE Trans. Serv. Comput.*, vol. 8, no. 4, pp. 601–616, 2015.
12. N. Dan and C. Yuan, "Attribute Based Access Control (ABAC) -based cross-domain access control in service-oriented architecture (SOA)," pp. 1405–1408, 2012.
13. M. Burmester, "T-ABAC : An Attribute-Based Access Control Model for Real-Time Availability in Highly Dynamic Systems," pp. 143–148, 2013.
14. X. Son Ha, T. Luong Khiem, and T. K. Dang, "Rew – XAC : An approach to rewriting request for elastic ABAC enforcement with dynamic policies," *Int. Conf. Advenced Comput. Appl.*, pp. 25–31, 2016.
15. I. Riadi, J. Eko, A. Ashari, and S. -, "Internet

Forensics Framework Based-on Clustering," *Int. J. Adv. Comput. Sci. Appl.*, vol. 4, no. 12, pp. 115–123, 2013.

16. A. Kannan and A. A. Abd El-Aziz, "A comprehensive presentation to XACML," *Third Int. Conf. Comput. Intell. Inf. Technol. (CIIT 2013)*, pp. 155–161, 2013.
17. X. Jin, R. Krishnan, and R. Sandhu, "A role-based administration model for attributes," *Proc. First Int. Work. Secur. Resilient Archit. Syst. - SRAS '12*, pp. 7–12, 2012.
18. P. Gupta, S. D. Stoller, and Z. Xu, "Abductive Analysis of Administrative Policies in Rule-Based Access Control," *IEEE Trans. Dependable Secur. Comput.*, vol. 11, no. 5, pp. 412–424, 2014.
19. T. Priebe, W. Dobmeier, C. Schläger, and N. Kamprath, "Supporting attribute-based access control in authorization and authentication infrastructures with ontologies," *J. Softw.*, vol. 2, no. 1, pp. 27–38, 2007.
20. V. C. Hu, D. R. Kuhn, and D. F. Ferraiolo, "Attribute-based access control," *Computer (Long. Beach. Calif).*, vol. 48, no. 2, pp. 85–88, 2015.
21. D. Xu and Y. Zhang, "Specification and analysis of attribute-based access control policies: An overview," *Proc. - 8th Int. Conf. Softw. Secur. Reliab. - Companion, SERE-C 2014*, pp. 41–49, 2014.
22. S. Bhatt, F. Patwa, and R. Sandhu, "An attribute-based access control extension for OpenStack and its enforcement utilizing the policy machine," *Proc. - 2016 IEEE 2nd Int. Conf. Collab. Internet Comput. IEEE CIC 2016*, pp. 37–45, 2017.
23. L. Røstad, *Access Control in Healthcare Information Systems*, no. June. 2009.
24. Jim Longstaff, "Cyber Security and Privacy: Trust in the Digital World and Cyber Security and Privacy EU Forum 2013 Brussels, Belgium, April 18-19, 2013 Revised Selected Papers," *Commun. Comput. Inf. Sci.*, vol. 182 CCIS, pp. 127–137, 2013.
25. J. Liu, X. Huanga, and J. K. Liuc, "Enhanced secure sharing of personal health records in cloud computing," *Futur. Gener. Comput. Syst.*, vol. 52, pp. 67–76, 2015.