GUISET LogOn: Design and Implementation of GUISET-driven Authorization Framework.

Obeten O. Ekabua
Dept. of Computer Science and Info. Systems,
University of Venda, Private Bag X5050,
Thohoyandou 0950, South Africa
Email:obeten.ekabua@univen.ac.za

Matthew O. Adigun
Department of Computer Science,
University of Zululand, Private Bag X1001,
KwaDlangezwa 3886, South Africa
Email: matthewo@pan.unizul.ac.za

***Abstract* - Authorization is an important part of GRID security systems with each GRID domain having its own policies that may change dynamically. Authorization ensures that resources can be accessed only by parties who have the appropriate privileges. Many authorization frameworks exist, but these are not applicable to our GUISET (GRID-based Utility Infrastructure for SMME enabled Enterprise Technology) domain. Therefore in this research as reported, we have developed and implemented a GUISET-driven framework, as a security gatekeeper, that satisfies access and privacy requirements for service requesters and providers in GUISET environment.**

Keywords - Authorization; Authentication; GUISET; GRID; Security; Service.

I. INTRODUCTION

As the days goes by, Web Services are being echoed as a solution to the next generation enterprise integration. A large scale service-oriented computing environment like GRID consists of many computers, storage systems and other devices distributed over heterogeneous wide area networks, but presents unique security problems that are not addressed by traditional client-server/distributed computing environments. While providing basic security requirements like authentication, authorization, confidentiality and integrity, the security infrastructure for GRID and Web Services must also be able to support more advanced security features like dynamic delegation of access rights [1].

Using personal sensitive information so as to gain access to a resource in a distributed environment raises an interesting paradox. Firstly, in order to make the services and resources accessible to legitimate users, the authorization infrastructure requires the users' attributes. Secondly, the users may not be ready to disclose their attributes to a remote service provider without determining exactly who the provider is and how personal attributes will be used [2].

GRID refers to the collection of distributed (networked) computers that are geographically dispersed, pooling resources together in such a way that users may utilize processing, storage, software and data resource from any of interconnected computers, leading to greater resource sharing and higher utilization ratio. Therefore, GRID may be viewed as virtual organisations (VO) [3]. Thus, a GRID system is a VO comprising several independent autonomous domains. The security of the GRID system should provide the same protection that conventional systems provide, including establishing the identity of users or services (authentication), protecting communications (encryption/decryption), determining who is allowed to perform what actions (authorization), and recording the important operations processed by the systems (auditing) [4] . GRID can give VO members direct access to each other's computers, programs, files,

data, and networks. Therefore, the sharing of resource in VO must be controlled, secure, flexible, and usually time-limited. The need to support the integration and management
of resources within VO introduces challenging security issues in GRID environment [4].

Authorization is an important part of GRID system, in which every domain may have its own policy and may change its policy dynamically. Hence, the authorization mechanism of GRID computing platform needs to support multiple security policies and need to have flexibility to support dynamism in security policies [4, 5].

Security is becoming increasingly significant when integrating services across multiple VOs. The different resources in a GRID have different access policies, including how they authorize users in accessing those resources or services [5]. Many authorization mechanisms exist including role-based authorization, rule-based authorization, and identity-based authorization. But these authorization mechanisms alone cannot satisfy the access requirements of distributed services as the access depends on many other factors like privacy requirements of the requester, authentication requirements of the service, trust relationship with the requester, authorization and management policies among participating parties, etc. [1].

Authorization ensures that resources can be accessed only by parties who have the appropriate privileges. This makes the resource gatekeeper to require that some level of trust be established before sensitive information can be released. Service requesters are required to submit sufficient authorization credentials before access will be granted. Wherever people are involved in the exchange of digital information, such as credentials, privacy becomes an issue of some concern [2]. Authorization in a distributed environment should be determined as a result of evaluating the request of an authenticated user against various policies like privacy policy, trust policy, authorization policy. Many authorization mechanisms for large scale distributed systems like GRID and Web

 ISBN: 978-1-61208-106-9

ignore one of the components from privacy, trust and policy [1].

The significance of small, micro and medium-sized enterprises SMME in stimulating economic growth, generating economic growth, generating employment, creating social cohesion, as well as regional and local development in Africa is almost undisputed. Throughout the continent SMME promotion is priority in the policy agenda of most African countries as its contribution to poverty alleviation and economic development is globally recognized [3].

The GRID-based Utility Infrastructure for SMME enabled Technology (GUISET) is an infrastructure used to solve barriers experienced by SMME in Africa due to the un-affordability of the technology to run them. The concept of GUISET is based on the idea that there is indeed a technology that is affordable for these SMME to use, through the utility approach to service delivery [3]. The GUISET infrastructure allows these SMME to share information by helping them market and sell their products without spending much on the technology. The infrastructure allows these SMME to subscribe for only needed services that are available in the GUISET GRID and that can help in the marketing and selling of the products online. This again raises the issue of security concerns since during an online transaction or payment, sensitive data is being processed, and this then creates a need for such data to be properly secured [3].

II. REQUIREMENT ANALYSIS

In this section, we mainly focus on the analysis of requirements that leads to the design of the authorization framework and the model of authorization as applicable in the GUISET environment. We also present a typical usage scenario and some important security requirements as well as other vital design considerations for the GUISET infrastructure.

The analysis is based on the requirement analysis for GUISET authorization framework and on the collection of all relevant and critical information pertaining to the framework.

a) **Requirement name: Resource Request**
Description: This feature allows the client to make a request for the service that he/she want to access.
Justification: The framework should evaluate client request against the policies before granting GUISET services.

b) **Requirement name: Grant/Deny Access**
Description: The framework should ensure that every GUISET resource is secured from being access by unauthorized client by using policies.
Justification: The framework should allow authorized client to have access to the GUISET services that are ready available.

c) **Requirement name: Decision**
Description: This feature allows the framework to make decision based on the client request for the services.
Justification: The framework should make sure that client must conform to service policy in order to have to have access to the service.

III. DESIGN METHOD

A. USE CASE DIAGRAM

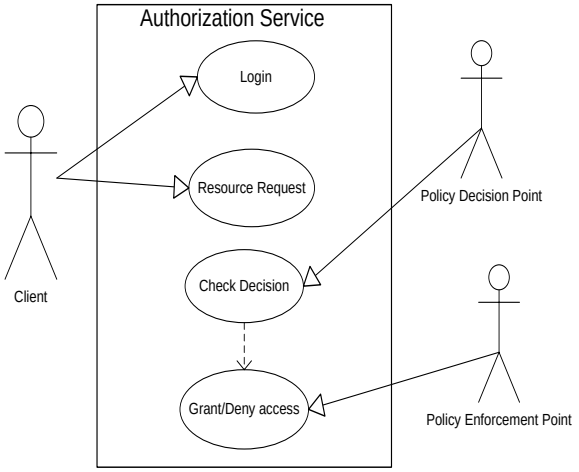

Figure1: Use Case Diagram

B. Use Case Descriptions

Use case descriptions are as follows

i) **Login Details**

TABLE I: LOGIN DETAILS

Use Case Name	Login.
Participating Actor	Initiated by Client communicating with the system.
Entry Condition	The client login to access information on the system.
Flow of Events	Client enters username and password to access information on the system.
Exit Condition	Opens a new window if the username and password is correct.

ii) Resource Request

TABLE II. RESOURCE REQUEST

Use Case Name	Resource Request
Participating Actor	Initiated by Client.
Entry Condition	Client makes a resource request to the system.
Flow of Events	The controller of a resource or service-a Policy Enforcement Point (PEP) - checks whether the client Authorized to have access to the resource or with the service with the Policy Decision Point (PDP).
	The PDP provide authorization service to the PEP
Exit Condition	PEP check decision provide by PDP to either grant/deny client access.

 ISBN: 978-1-61208-106-9

iii) Check Decision

TABLE III. CHECK DECISION

Use Case Name	Check Decision.
Participating Actor	Initiated by PDP.
Entry Condition	PDP check the message that was pass by PEP based on the client request
Flow of Event	The PDP provide authorization service to the PEP.
Exit Condition	The PEP check decision if client is authorized to have access to the resource.

iv) Deny/Grant Access

TABLE IV. DENY/GRANT ACCESS

Use Case Name	Deny/Grant Access
Participating Actor	Initiated by PEP.
Entry Condition	PEP check the decision that was made by PDP based on the client request to the service or resource.
Flow of Event	PEP check whether the client is authorized to have access to the service or resource with the PDP
Exit Condition	PEP check permission of client, if client is authorized then PEP grant access, if client is not authorized PEP deny access.

C. GUISET Authorization Scenarios

GUISET Authorization scenario examples are:

a) GUISET Scenario Name: Accessing new products
Participating Actor: Khulani Ngwenya
Flow of Events: Khulani as GUISET client submit his credential to the BBC Company marking a request for accessing new products as a client to the company.
: The BBC Company checks whether Khulani is an authorized registered client to the company who can access the new products that are available.
: If Khulani have been granted access and proven to be an authorized registered client in the company then Khulani can access new product that are in the company.

b) GUISET Scenario Name : Search for product
Participating Actor: Khulani Ngwenya
Flow of Events: Khulani as client of GUISET Infrastructure he wants to search for new product in BBC Company of his choice.
: Khulani as an authorized client in the GUISET Infrastructure he can now search for the product of his choice as an authorized client.

IV. GUISET ARCHITECTURE

The architecture is 3 tiers consisting of multi-modal interfaces, middleware layer and the GRID infrastructure layer. Each tier consists of its relevant components. In the GUISET architecture, we pay more attention on the GRID Infrastructure Layer as this is where services or resources reside and therefore there is a great need to ensure that only authorized Client has access to these services [4].

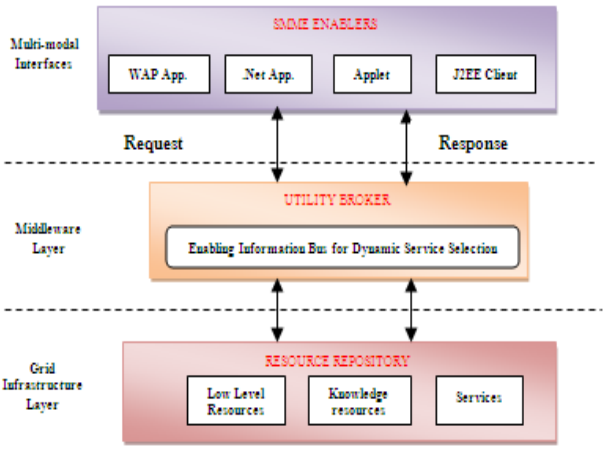

Figure 2. GUISET ARCHITECTURE

V. MESSAGE BROKER COMPONENT

This message broker component is the intermediary in the access path between the Subject and the web service operation or the resource being requested, it manages all the interactions related to authorization and access control by intercepting all SOAP messages when requests to services are being made [6].

This component not only controls access to web service requests but also all interactions related to the collection of authorization information. The figure below depicts how messages are intercepted before access to a resource is granted [6].

The UML sequence diagram, Figure 4 shows the sequence related to the message broker component, it shows the interaction of the various sub-components of the message broker for a basic operation [6].

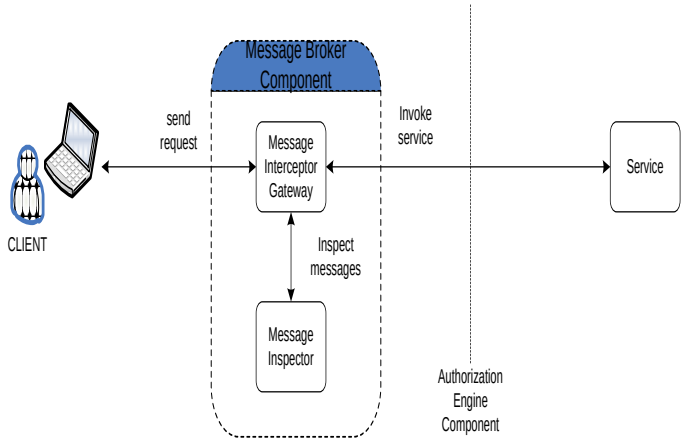

Figure 3. Message Broker component

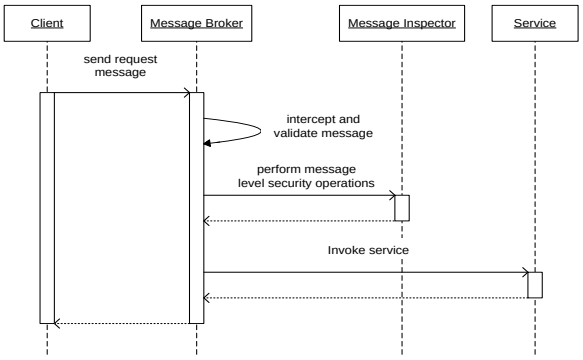

Figure 4. UML sequence diagram – Message Broker component

VI. GUISET AUTHORIZATION FRAMEWORK

Authorization in a distributed environment such as GRID needs to be flexible and scalable to support multiple security policies [6]; we made use of the XACML (eXtensible Access Control Markup Language) and SAML (Secure Assertion Markup Language), which are the two recognized important authorization related standards [1, 6].

As shown in Figure 5, authorizing a request from subject is first intercepted by PEP (Policy Enforcement Point).PEP constructs an authorization decision query and passes it to authorization handler. The result of this query determines if the request is to be granted/deny access to requested Service/Resource. The authorization decision query has details of about the identity of the subject, the service requested and the purpose for which service is requested [1].

Authorization Engine passes this information to PDP (Policy Decision Point). The policy is retrieved by PDP from PRP (Policy Retrieval Point). If the policy information is not available at PRP, it may be retrieved from policy store. The policies are written by administration using PAP (Policy Administration Point). PIP (Policy Information Point) is used by authorization engine to retrieve attributes of resources, subject and environment. After retrieving this information, authorization engine prepares a final result and passes it to PEP. If subject conforms to established privacy and other policies, PEP grants access of service/resource to subject, otherwise the access is denied [1].

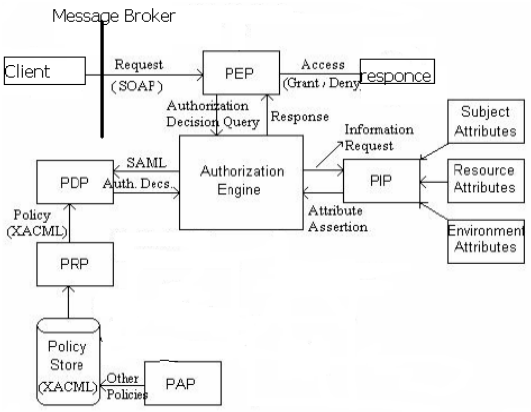

Figure 5. Authorization Framework for GUISET

VII. AUTHORIZATION MODEL

Authorization Engine Component

The authorization model must be able to support multiple security policies and need to have the flexibility to support dynamic changes in security policies [4]**.** The Authorization Engine component is responsible for making decision based on the authorization policies and access control policies stored in the policy repository, these ensure that authorized client requesting for services are only granted access to services/resources they are only requesting. Every Service Provider within a Domain has its own policies and he can change them dynamically [1].

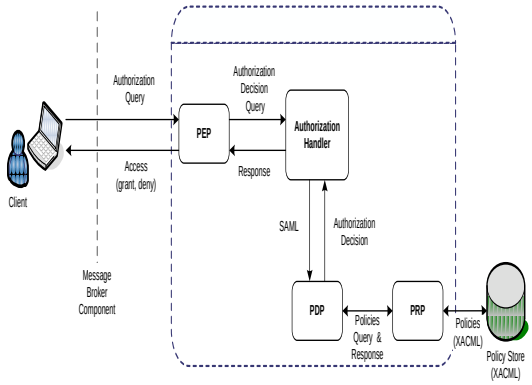

Figure 6. Authorization Engine component

Figure 6 shows a typical application environment where a user would want to access a particular service or resource, the authorization decision request from the Message Broker component is first intercepted by Policy Enforcement Point (PEP), this constructs an authorization decision query which contains information on the acquired token of the service requestor and also the details of the service being requested. This basically validates a client whether to access the resource being requested, and then this query is passed onto the Authorization Handler (AH) which then passes this information to the Policy Enforcement

 ISBN: 978-1-61208-106-9

Point (PDP), which then looks up the security access control policies [1].

The results of this query determine whether access to resource/service is granted. The PDP responsibility is to retrieve policies from the policy store using a Policy Retrieval Point (PRP) and if the policy is not available in the PRP, it may then be retrieved from the Policy Store, the Policy Store has capabilities of importing/exporting these policies in an XACML form and are constructed as a set of rules against the target service, i.e., (Client, resource, action) Client refers to the requestor, resource refers to the service or resource being requested and action refers to the kind of action to be performed on the requested service [1].

The significance of using XACML is due to the fact that it provides both a policy language and an access-control decision request/response language to meet the security access control requirements. With XACML, the PEP forms a query language to ask the PDP whether or not a given action should be allowed. The Authorization Engine get information from the PDP and after that the Authorization Handler prepares the final results then passes it to the Policy Enforcement Point which then passes it to the Message Broker component, based on these results the Policy Enforcement Point then returns a value of either (grant or deny) access to the requested resource [1].

VIII. IMPLEMENTATION

The main focus of this section is to show how clients interact with the system and how an authorized client accesses the services that are provided in the GUISET. We use diagrams to show how client interacts with the system before granting a service.

A. GUISET Authorization

One may want to ask how a client is granted an access for a service. This is easily achieved through the following steps:

a) Receive the data request and authentication tokens of the client that is requesting for service.

b) After receiving data request, make an authorization request for the service to the authorization engine.

c) The authorization engine will then return the authorization decision, and based on that decision, it will then know whether the client is authorized to access the services or not.

The above mentioned steps will now be presented through pictorial narration of client user interfaces.

B. GUISET Main page

This is the main page of the GUISET and it is the entry point to the GUISET Infrastructure. The main page of a GUISET has a Login button and register button. When a client clicks the login button, a Login dialogue appears.

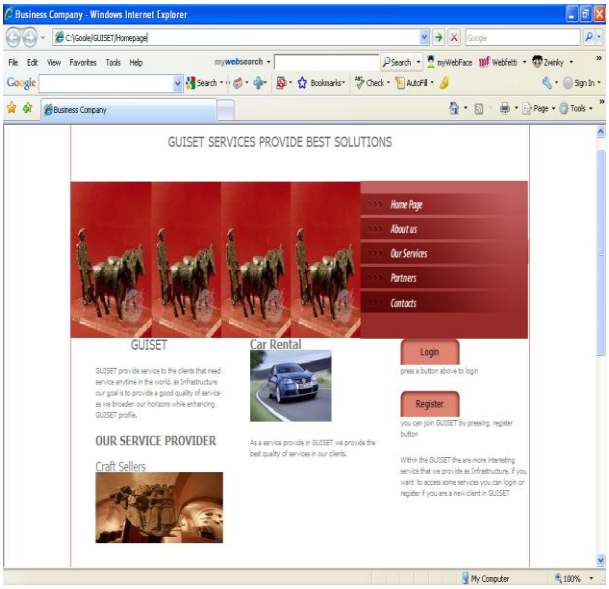

Figure 7. GUISET Main page

C. THE RESULT OF IMPLEMENTATION:

Figure 8 shows how authorization takes place in GUISET when a client wants to access services that are available.

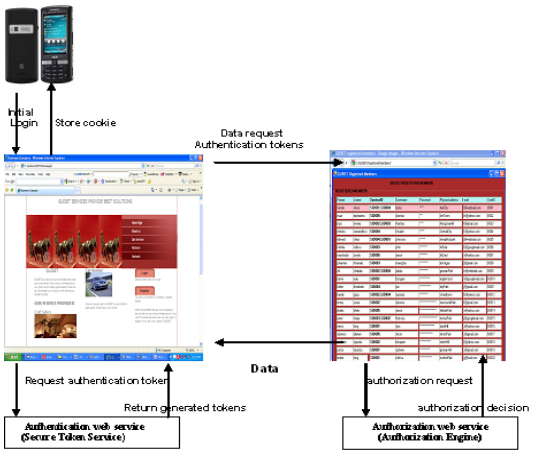

Figure 8: Result of Implementation

IX. CONCLUSION

Grid computing has become a very useful and interesting approach to enhance solution provisioning, sharing of resources, data and services in a distributed manner. The provision of large scale distributed services requires high level of authentication and authorization for access control. Available methods are not suitable to our GUISET environment as they seem to be dynamic to the environment of operation. GUISET with its dynamic nature also, and as part of our research focus, requires its framework and

implementation for its operation. Consequently, as reported in this paper, we have designed and developed a framework whose contribution is geared towards acting as a gatekeeper for access control in our GUISET service provisioning environment.

REFERENCES

[1] S. Singh. and S. Bawa, "A Privacy, Trust and Policy-based Authorization framework for Services in Distributed Environments". International Journal of Computer Science, Volume 2 Number 2 2007, ISSN 1306-4428, pp 85-92

[2] L. Marchel, et al., "Optimal Bandwidth Sharing in GRID Environment," Proceedings of the 15th IEEE International Symposium on High Performance Distributed Computing, HPDC-15, Paris, France, June 19-23, 2006, pp 153-163.

[3] T. Ziebermayr and S. Probst, "Web Service Authorization Framework." IEEE International Conference on Web Service (ICWS'04), San Diego, California, June 6 -9, 2004, pp.67-74

[4] M. Adigun.; J. Emuoyibofarhe and O. Migiro, "Challenges to Access an Opportunity to use SMME enabling technology in Africa". 1st ALL African Diffusion Conference, Johannesburg, South Africa, June 12-14, 2006, pp. 34-43.

[5] I. Foster,et al.,"A Multipolicy Authorization Framework for GRID Security". Proceedings of the 5th International Symposium on Network Computing and Applications. IEEE Computer and Society, Washington, DC, USA, July 2006, pp 269-272.

[6] R. Tatyana, N. Clifford, and L. Zhou, "Integrated Access Control and Intrusion Detection (IACID) Framework for Secure Grid Computing". Technical Report, USC Internet and GRID Computing Lab (TR 2004-6) May 21, 2004

 ISBN: 978-1-61208-106-9