Mutual collection of clouds growth for data scalability in SOA

Candidate name: Divya manuguri . divyamanuguri@gmail.com Guide name: Mrs padmaja rani

www.vspt.in

Abstract: In service oriented architecture normally clients(s) will be having direct access to services(skeleton) in proxy. So we propose SOA to over come the previous architecture to migrate to SOA(proxy architecture is not there in the previous architecture). We useLDAP(lightweight directory access protocol). This will protect the direct access to services. And coming to service side, services will be made mutually deployed data with scalable constraints to reduce hassle ness in maintaining the data.

Once client(s) sends the request the DUPRED(duplicate reduction technique) will identify the duplicate requests and will be processed once and resolute to number of requests. All requests will be encrypted/decrypted using ROTA technique.

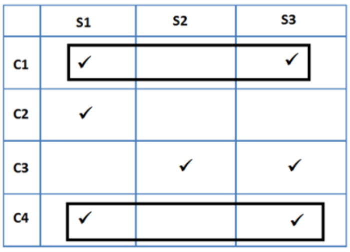
Modules:

- Services scalability with segmentation
- DUPRED
- ROTA for encryption/decryption

Description:

Services scalability with segmentation: When we update the services we take data sensitivity in 2-level segmentations (sensitive and non sensitive) to have good scalable services to maintain the data. This will be maintained by central service to update/modify/delete the data in services.

DUPRED: This will ignore the duplicated requests. If we have ***{S1,S2,S3}*** services and ***{C1,C2,C3,C4}*** clients. Following is the table for the requests.



	S1	S2	S3
C1	✓		✓
C2	✓		
C3		✓	✓
C4	✓		✓

PIC1:

In the above table C1 and C4 is accessing the same requests to S1 and S3. So DUPRED will find this requests to processed by LDAP for routing and get the response from services. So clients will be sending the duplicate calls but the DUPRED will reduce to one and resolute to number of requests.

ROTA: This is one technique to encrypt the data. Normally the requests will be placed over the network directly for processing. The pseudo code will be in the following algorithms section.

Example in the above table:

Enter Your String

India is my country

Original Test is

India is my country

Encrypt Test is

$ÜÈ40@49@6<@17:Üèä<

Decryption Test is

India is my country

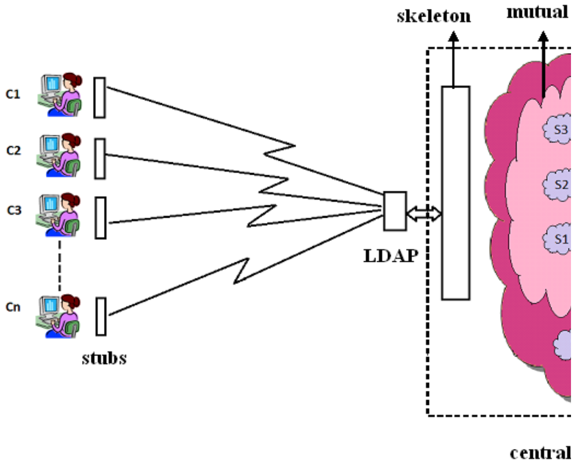
.

Introduction:

- In service oriented architectures all requests will be directed to services without any intermediatery layers which is unsecured. So in this new architecture we propose SOA with proxy(LDAP) integrated architecture. LDAP is light weight directory access protocol which redirects clients encrypted requests to relevant services which are there in services side which are getting monitored and controlled by central service. This central service will control all the services in one scenario which is segmentation technique. We have sensitive/non sensitive data categorization to stabilize the data services in a proper way to stabilize and have good scalability at the services side for fast and time saved processing. Here LDAP will be having the routing technique to route the non duplicated calls to services to ignore direct calls to services from clients.

- When ever the new service added or updation of existing service central service will give permission to do in segmented architecture. This leads for scalable and stabilized SOA with proxy mechanism.
- All clients requests and responses will be over the network in the form of only encryption . Here a new ROTA encryption/decryption technique will be used.

Architectural design:



Analysis:

In the above architecture services are mutually cooperative with central service monitoring. The central service always monitors and controls the existing services in the system. The central LDAP restricts the direct calls from clients to services. The clients calls, if any duplications will be in the checked and one request will be processed once and resoluted relevant clients. The main analysis is done in two ways. 1. Clients calls will be checked for duplications, and they will be processed once. 2. Services will mutually cooperative and no direct access to clients for better security.

Disadvantages of existing system:

1. Various services are existing at services side(SOA ie cloud) which are holding the huge data in the un recognized way.
2. As in the current system the files or not organized in the form of latest technologies like apache hadoop , google white papers, google file system the processing for the searching and traversing through the data.
3. There is no service such to route to valid service.
4. Once the client(s) requests for the particular request the request will be reached to relevant services and gets the chunks of data in the form of holomorphic structure which is of some threshold value to get the proper chunks of data relevant to request.
5. The consolidated chunks which are formed will be integrated with some format to present to client(s).
6. There is no proper security in processing the data or in the request routing.

Advantages of proposed system:

1. We will categorize the services in the proper organized index way.
2. The new algorithm will put the data in novel way while we deploying the data at the beginning itself and this reduces time complexity for searching.
3. The data will be put up by using 2 dedicated rules 1. The data will have the categories 2. Data will have the log for modifications (updation from service provider).
4. Services will be having no direct contact to any of the client(s) instead they will be under hybrid service which will be integrated with LDAP (light weight directory access protocol).
5. Service layer with LDAP will be having an intelligent system which will route to the valid service which is relevant to services with reducing the duplicate request processing.

6. The service middle layer will maintain the log for ignoring the duplicate search and process for the client(s) request.
7. Coming to the client part, client will send the request which will be categorized at the client side to reduce the burden on the middle layer.
8. So over all smooth/redundancy architecture will be established though search log grows.

Algorithms:

ROTA encryption:

Rota Encrypt/Decrypt Algorithm:

Encryption:

Input: Plain text

Output: Cipher Text

```
Initialization:
n←0              //total number of characters.
∑D←0             //total data
Cipher←1
∑E←0             //total Encrypted data
t1←null          //temp variable
t2←null          //temp variable
Loop for each c in D
        n=0
        t1←LSHIFT(c,n,CIPHER)
        t2←RSHIFT(c,n+1,CIPHER)
                n+1
        E←APPEND(t1)
        E←APPEND(t2-1)
End loop
```

Decryption:

Input: Cipher text

Output: Plain text

```
Initialization:
N←0 // initialization for total number of characters.
∑E←0 //Cipher texts
∑N←0         // plain text
T1←null
T2←null
Loop for each c in E
n=0
t1←RSHIFT(c,n,CIPHER)
t2←LSHIFT(c,n+1,CIPHER)
                n+1
        D←APPEND(t1)
        D←APPEND(t2+1)
End loop
```

In **Rota** Encryption Algorithm the Data Files will be encrypted in the form of cipher text. When transmitting the data from node to node the source file data was encrypted by the sender using Rota Encryption Algorithm.

Here the Data Characters will be converted into based ASCII Values. These values are internally turned into binary bits. We perform operations such as Left Shift and Right Shift on Binary Data. If the ASCII values are Even number digit then perform 1LeftShift operation on Even ASCII value of given character (B<<1).its generate Cipher text, and then we again perform 1RightShift operation on cipher text. Now we got Plain text. If the ASCII values are odd number digit then perform 1RightShift operation on odd ASCII value of given character (A>>1). Its generate Cipher text, and then we again perform 1LeftShift operation on cipher text. Now we got Plane text.

DUPREDapproach:

Basically DUPRED is the duplication reduction technique to reduce the burden on the central or intermediatory LDAP service. Normally clients will keep sending the requests to middle service which is LDAP where DUPREDfunctions to reduce the clients duplicate request. Normally DUPRED will maintain the data structure which will be logged continuously for the requests as well as responses. Once client sends the request DUPRED will check the duplications from the existing data structure if the response for that request available will be routed to relevant client else the request will be processed by remote service and response will be routed to client.

```
Input :   $\sum C , \sum S , \sum R$
Output:   $\sum Res (C)$

Step1:  Flag = RESET
Step2: For each  r  in  R
Step3: Loop  start
Step4: Flag = DUPLICATE (r)
Step5: If (flag)
Step6: Res (Ci)    <—  RESPONSE (r)
Step7: Else
Step8: Res (Ci)    <—  SRESPONSE (r , Si)
Step9: flag=SET
Step10: end  if
Step11: end for
```

Overview of this work:

In this project we have used an architecture where we will categorize the services. This helps in reducing the overall congestion at each and every node. This architecture also has the advantage of maintaining the load balancing technique in the form of logs. These are logs are helpful in doing modifications. So with the usage of SOA we are creating an enhanced version of architecture where we can reduce the traffic overhead. Also the services are no way directly connected with the clients and so for that purpose we have another hybrid layer where it has LDAP protocol. This reduces the insecure transmissions. So the clients cannot communicate directly with service provider.Clients do not interact with the services directly which is a known fact and all their requests are being processed by the middle layer. This middle layer maintains a log which reduces the burden over the top layer and it also reduces redundancy.

Conclusion

The infrastructure services layer within the SOI Reference Architecture is oftenglossed over and rarely discussed other than as a layer within the overall referencearchitecture. But the infrastructure services layer can have a positive impact on thereliability, security, and performance of a SOA implementation. The infrastructureshould be given consideration in all phases of the maturity process, but especiallyin the web application development stage. Consider carefully those services thatcan be enabled, integrated with, or moved to the infrastructure layer and whatimpact such a decision may have on the organizational SOA.This paper proposed a service-oriented cloud computingarchitecture SOA that allows an application to run ondifferent clouds and interoperate with each other. The SOAis a 4-layer architecture that supports both SOA and cloudcomputing. SOA supports easy application migration fromone cloud to another and service redeployment to differentclouds by separating the roles of service logic provider andservice hosting/cloud providers. It promotes an open platformon which open standards, ontology are embraced. The paperalso introduced related topics for future research, such as servicedemand prediction and SLA negotiation, and service requestdispatching algorithms. More work will also be conducted todevise the ontology systems needed for a working SOA, anda prototype with all features discussed.

References

[1] Wikipedia - Cloud Computing. [Online].

[2]Parascale. [Online].

[3] Elastra. [Online].

[4] Appirio. [Online].

[5] Rochwerger B et al., "The RESERVOIR Model and Architecture for," IBM Systems Journal, 2009.

[6] Christian Vecchiola, Xingchen Chu, and Rajkumar Buyya, "Aneka: ASoftware Platform for.NET-based Cloud Computing," in High Speed andLarge Scale Scientific Computing, 2010.

[7] Rajkumar Buyya and Chee Shin Yeo, "Cloud Computing and EmergingIT Platforms: Vision, Hype, and Reality for Delivering Computing as the5th Utility," Future Generation Computer Systems, pp. 599-616, 2009.

[8] Ying Huang et al., "A Framework for Building a Low Cost, Scalable andSecured Platform for Web-Delivered Business Services," , 2009.

[9] Bernard Golden. (2009, January) Computer World. [Online].

[10] Mark Kyrnin. About.com. [Online].