March 2022

PRIVACY POLICIES FOR DISTRIBUTED OBJECT-BASED STORAGE IN MULTI-CLOUD DEPLOYMENTS

NIRANJAN M M

PRIVACY POLICIES FOR DISTRIBUTED OBJECT-BASED STORAGE IN MULTI-CLOUD DEPLOYMENTS

AUTHOR:
NIRANJAN M M

ABSTRACT

Object-Based Storage (OBS) has become the main type of storage system in the cloud. With applications moving to cloud-based services (viz., micro services architecture, Software as a Service [5G, SD-WAN], Security as a Service etc.,) and large cloud vendors having many data centres globally, where-in data can be stored across data centres belonging to different countries or states, hence it is challenging to provide data security and privacy. Also, with different laws for providing data protection and privacy we need to guarantee that PII data (it may be government departmental data, personal health records etc.,), is adhering to government laws. Additionally, there are use cases where customers might be using services belonging to different cloud service providers and hence their data also distributed across different cloud service providers. The techniques presented herein propose method to ensure that all PII data stored in the cloud using Object-Based Storage enforced with government data protection and privacy laws, by having de-centralised distribution of OBS policies across the cloud data centres and across different cloud service providers without compromising on GDPR.

DETAILED DESCRIPTION

From the last few years, Object-Based Storage (OBS) has become the main type of storage system in the cloud (used in Amazon S3, IBM, GCP, Rackspace etc.,). OBS provides flexible number of metadata and attributes for each storage object, which is lacking in traditional file system-based storage.

With applications moving to cloud-based services (viz., micro services architecture, Software as a Service [5G, SD-WAN], Security as a Service etc.,) and large cloud vendors such as GCP, AWS, Azure etc., having many data centres globally, where-in data can be stored across data centres belonging to different countries or states, hence it is challenging to provide data security and privacy.

Also, with different laws/acts for providing data protection and privacy i.e.,

- General Data Protection Regulation (GDPR) Act 2018, Data Protection Law Enforcement Directive
- 1974 U.S. Privacy Act
- 1996 Health Insurance Portability and Accountability Act (HIPAA)
- In general, adhering Personally Identifiable Information (PII) data to local government laws (it could be per country/per state/per province) for providing data protection and privacy.

we need to guarantee that PII data (it may be government departmental data, personal health records etc.,), is adhering to government laws. Also, there are use cases where customers might be

using services belonging to different cloud service providers and hence their data also distributed across different cloud service providers.
There are techniques to provide authentication and security to the cloud services, to secure the stored data using encryption methods and to provide privacy to the stored data, but all of them lack the following aspects:

- None of the existing techniques consider privacy across different cloud service providers.
- Existing techniques use centralised approach (e.g., having OBS Controller configured through centralised server etc.,)

Hence, need to ensure that all PII data stored in the cloud using Object-Based Storage enforced with government data protection and privacy laws. Also, should consider privacy across different cloud service providers and applicable to different use cases.

As explained above, it is challenging to guarantee that PII data which is stored in the cloud data storage adheres to government data protection and privacy laws. For example, currently to provide data privacy, an organisation may sign an agreement with cloud service provider where all applications and data storage must meet GDPR Act or any other Privacy Act (of that country/state/province), where sensitive data (government department data, personal health records etc.,) is kept within the country.
But with large cloud vendors such as GCP, AWS, Azure etc., having many data centres globally where-in data can be stored across data centres belongs to different countries or states, hence it is challenging to provide data security and privacy. This will be multi-folded, if customers use services belongs to different cloud service providers and hence their data also distributed across cloud service providers.

The techniques presented herein propose method to ensure that all PII data stored in the cloud using Object-Based Storage enforced with government data protection and privacy laws, by having de-centralised distribution of OBS policies across the cloud data centres and across different cloud service providers without compromising on GDPR.

In this method, Holochain is used to efficiently distribute the data privacy policies across the cloud data centres and across different cloud service providers, where-in service providers specific privacy is achieved using validation rules of the Holochain. In place of Holochain, any de-centralized mechanism can be used to distribute policies across DCs and SPs.
Currently OBS is used as main storage system in the cloud. OBS include attributes and metadata used to describe the data. This method leverages metadata and attributes of OBS to provide unique identifier and attach policies to the data objects by encoding the policies into the metadata of these data objects by retrieving them from the Holochain DHT, to ensure all data is adhered to data protection and privacy act/law (locally applicable for the country/state/province).

This method involves the following steps:

1. Customer (administrator) write Service Type and Privacy Policy (including region information) as transaction to the Holochain DHT. Transaction is mutually authenticated with the Service Instances or users.

Example: Administrator defines the privacy policy as:

```
Policy Definition = {
    Service Type    = Medical Record.
    Policy ID       = PolicyID-1.
    Region Allowed = Country: India.
    Users Allowed   = User Group: UG-1.
}
```

1.1. Peer-to-Peer communication in Holochain happens using mutual-authentication of transactions and both peers update the signed transaction to the Holochain DHT. (Mutual authentication of transaction happens as explained in Figure-3)

2. Service instance or user would fetch Service Type and Privacy Policies from the Holochain DHT.

3. Based on Service Type, cloud service provider is selected (Let us say, cloud service provider-1 is selected here).

4. Based on the region-specific Privacy Policies set by the administrator (at step (1)), select the Cloud PoP having geographical compliance. (Let us say, Cloud PoP in India is selected).

5. Mutual authentication happens between Service instance and cloud provider. Mutual authenticated transaction contains privacy policies, identifiers, and region information to be stored as metadata into the Object describing where it can be stored. (Mutual authentication of transaction happens as explained in Figure-3).

6. Based on the policies in Metadata of the Object, the policies allow or disallow data being stored in remote location. To allow, remote location must be compliant with policy.

Figure-1 describe how privacy policies for Object-Based Storage are provided in multi-cloud deployments.

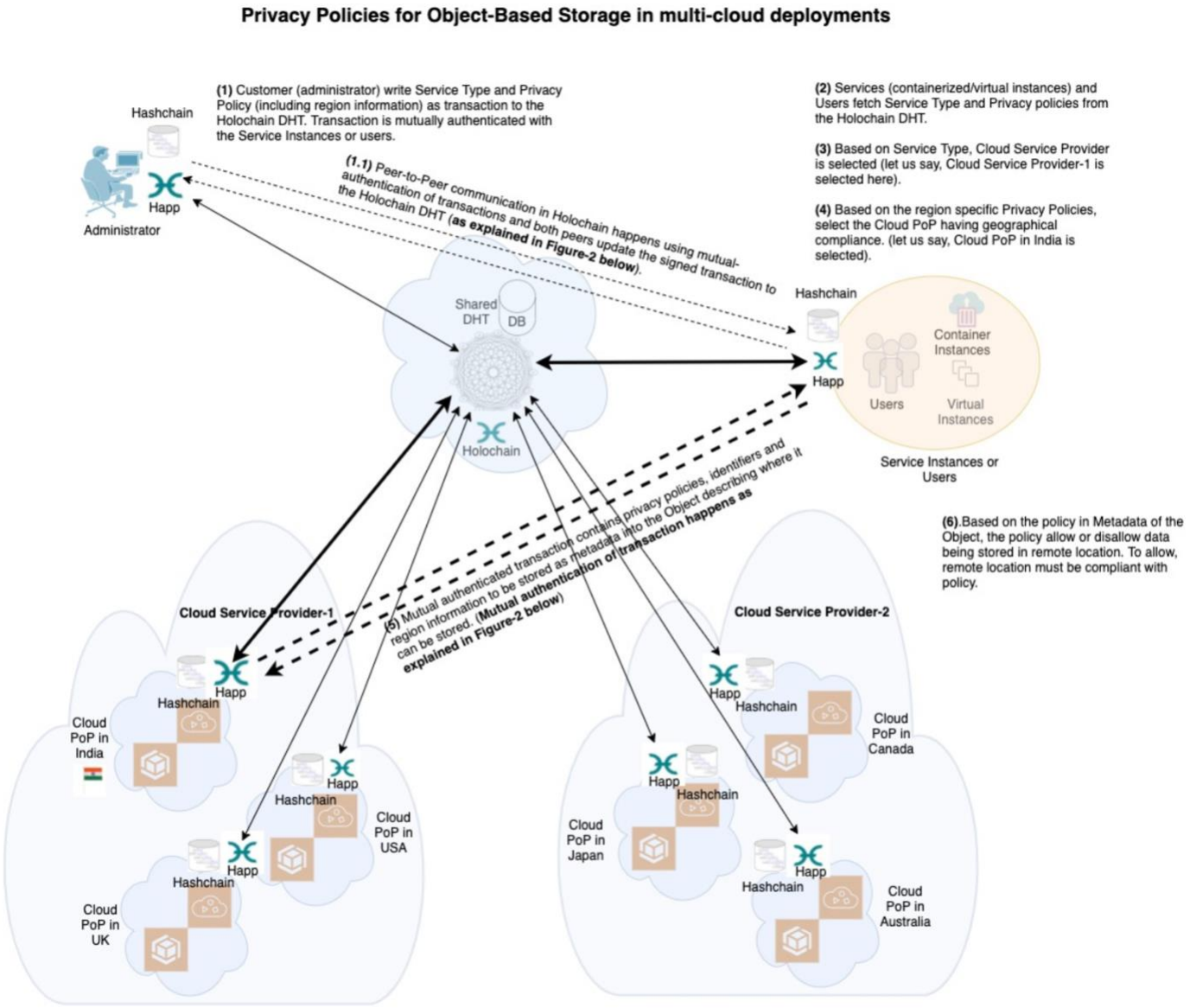

Figure-1

Figure-2 describe Data Object Template and sample Data Object for Health Record along with Metadata carrying Privacy Policies.

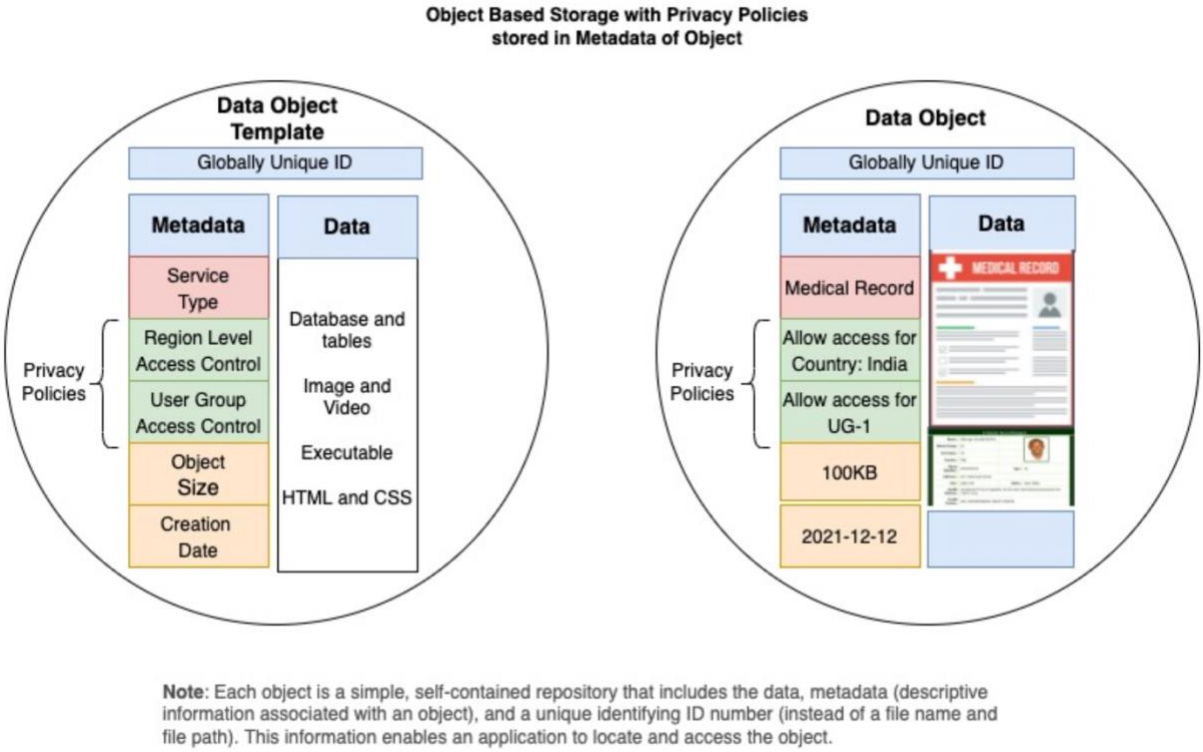

Figure-2

Example: Consider health record belongs to a person ABC with country of origin is INDIA.

- Let us say, there would be government law to store all health-related records within the country (here it is INDIA).
- Let us say, hospital administrator wants to add policy to the cloud Point of Presence (PoP) of AWS (a cloud service provider), so that data of the health record of person ABC is attached with policy to store only in cloud PoP Object-Based storage within India.
- Using Holochain, hospital administrator adds privacy policy to the Holochain DHT as transaction upon mutual authentication between Administrator and Service Instance.
- Service Instance selects AWS and India Cloud PoP to store the privacy policies in the metadata of the object upon mutual authentication between Service Instance and Cloud PoP.
- Now AWS India Cloud PoP read the policies from Holochain DHT and update the policies (saying only within India it is accessible) to the metadata of the data object which stores health record of person ABC whose country of origin is INDIA.

The techniques presented herein is explained in detail as below:

- Holochain provides different functionalities such as local Hashchain to store transaction at the source, DHT to store mutually signed transactions, gossip protocol to communicate between Holochain apps and Holochain provided privacy method using validation rules etc., These are all accessed/controlled through Halochain App (Happ).
- Each Cloud Service Provider (e.g., Amazon S3) and their Cloud PoP (of different region) would run Happ. Also, the configuration and management system would run the Happ.
- Customer (administrator) use Happ (or using wrapper APIs provided) to create privacy policies for the data to be stored in OBS.
- Happ running on the service instance would add the details of privacy policies of the data (to be stored in OBS) as transaction T1.
- This transaction T1 is mutually signed by Service Instance and Cloud PoP as explained in Figure-3.
- Cloud PoP of Cloud Service Provider would access the Holochain DHT to fetch/read the privacy policies of the data object.
- As we know, the advantages of OBS are that every object includes attribute and metadata describing that object.
- Cloud PoP write the metadata details (containing privacy policies) of the object identifying the Data Privacy and/or PII requirements of the data stored. This may include details such as country/state/province etc.,
- In short, the data object is now having metadata attached to it describing its Data Privacy and PII requirements.
- As we know, Object-Based Storage is a distributed file system (i.e., object or parts of the data object can be stored anywhere). Hence, need to prevent any sensitive data (containing PII) from being stored in a place that violates data protection and privacy policy. Thus, when the Cloud Provider attempts to store/delete/update/modify the object, a check is made with the data privacy policy (which was attached to the object as metadata). Only data centres that have geographical compliance with the privacy policy may be used for storage. All other sites that do not comply will not be used for storage.
- With Holochain in place, customer can get the reports with respect to where the data is stored globally, provides auditing for verifying the compliance time-to-time.
- Access to the data storage (OBS) on different Cloud PoP would be validated via the Holochain validation rules.
- Holochain could be externalised across Cloud Service Providers i.e., other cloud service providers can register their Cloud PoP as part of the Holochain. The privacy across different cloud service providers can be controlled through Holochain validation rules.

Figure-3 describe mutual authentication of transactions between Service Instance and Cloud PoP in Holochain. Same thing holds good for mutual authentication between Administrator and Service Instance.

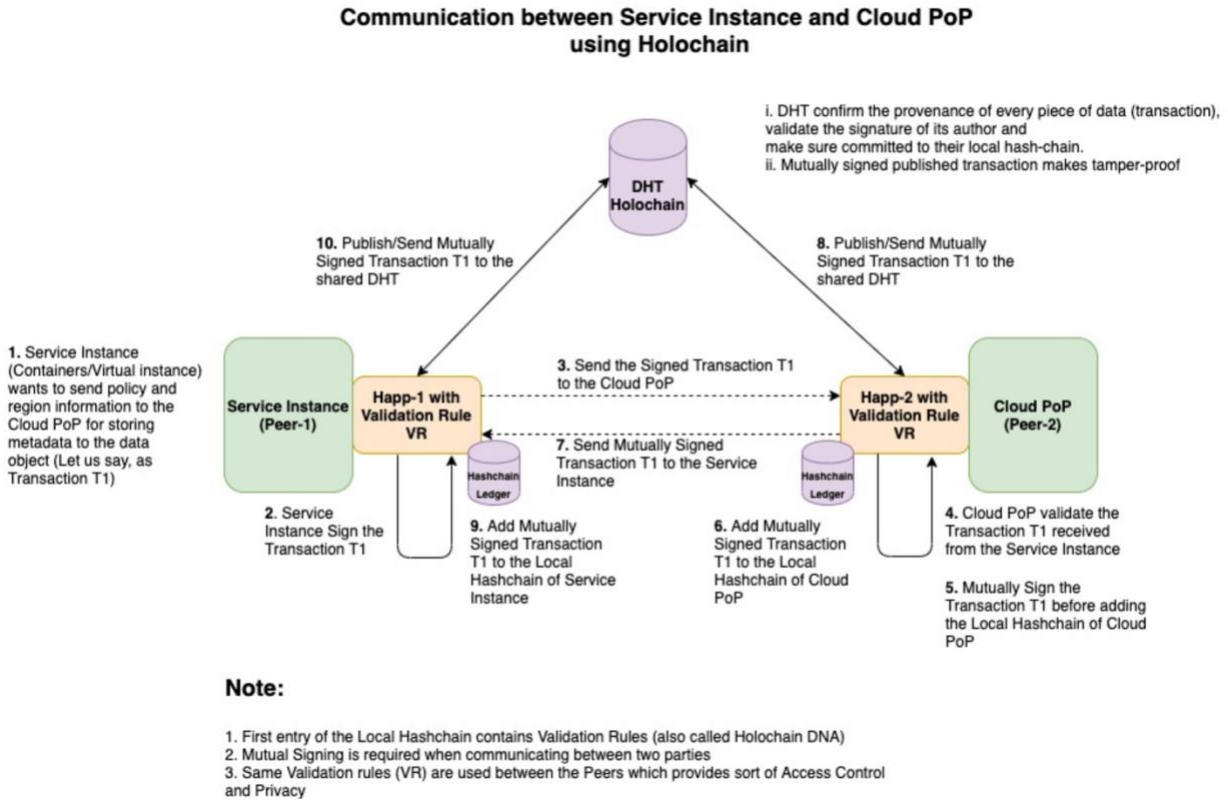

Figure-3

The techniques presented herein helps to define privacy policies for data storage (OBS) as per the GDPR and other Privacy acts, where-in public sector (defence, health etc.,) must adhere to country/state/province specific privacy policies. Policies may be different for each region and industry. This method helps to distribute the policies for each POP of cloud Service provider and even across the cloud service providers. This method is useful for providing data protection and privacy policies across the cloud using Object-Based storage. Additionally, this method is useful for customer deployments having multiple cloud service providers, wherein privacy is provided using validation rules. Moreover, this method is scalable with the use of distributed Holochain.