**LSA Segment Publisher**

There will be an *NlsrSegmentPublisher* class that will take an Nlsr’s *certificate-name* as parameter and this *Nlsr’s* identity will be used for signing each data segment. This way if we later have other publisher that needs to be signed by the NLSR identity, we can directly inherit from *NlsrSegmentPublisher*. *LsaPublisher* will inherit from *NlsrSegmentPublisher* class, which is an abstract class, and the former has to implement the *generate*() abstract method of the latter.

The current implementation of *SegmentPublisher* will be modified and will be turned into *NlsrSegmentPublisher.*



**LSA Segment Fetcher**

For implementing LSA Segment Fetcher, we just need to write four methods (one for each LSAs and one more for listing all the LSAs) in the *LSDB*class that will construct *SegmentFetcher* class and invoke *fetch()* method of the latter class with necessary interest names.