Please use this document to add your questions related to registry integration and digital signatures.

1. **How does a subscriber get Registry’s public keys to decrypt information on on\_subscribe?**

*On an immediate note, the registry’s public keys will be made available on a publicly accessible location like the developers.beckn.org website. Later the registry’s public key will be available on a country level registry.*

1. **What to be used as a subscriber\_id: http domain or some text id in a particular format?**

*The subscriber\_id’s default value will be the Fully Qualified Domain Name of the subscriber unless specified otherwise as a network policy.*

*In case the subscriber does not have an FQDN, like say, an IP address, then addition of IP address on the that must be allowed as a network policy*

1. **We need example, for a given public\_key and private\_key combo generate the following combinations for multiple json payloads , so we can add test cases against them**
   1. **Payload**
   2. **BLAKE of Payload**
   3. **Signature**
   4. **Authorization headers**

***Ed25518 PrivateKey(pem):***

*MFECAQEwBQYDK2VuBCIEIDi8BMqCB5VuApOoJf6Ysi8\/QQ3IW\/+bR+rOUd1\/kWdvgSEA91QWhuTEdFt26t+gWSxSGjI1BY5lZxgfjeqt5jVzQEg=*

***Ed25518 Public Key(pem format)"***

*MCowBQYDK2VuAyEA91QWhuTEdFt26t+gWSxSGjI1BY5lZxgfjeqt5jVzQEg=*

***Ed25518 Public Key(raw format)***

*4W8Y8ELQ6LpmfiGeI/VUzOlp5aRswc5ayGWPLostvsw=*

***Request PayLoad:***

*{"context":{"transaction\_id":"20748706-7a57-495c-8b8e-08a0a3143f83","country":"IND","city":"080","domain":"nic2004:52110","action":"search","message\_id":"20748706-7a57-495c-8b8e-08a0a3143f83","ttl":"PT1M","core\_version":"0.9.1","bap\_uri":"https:\/\/beckn-one.succinct.in\/local\_retail\_bap","bap\_id":"beckn-one.succinct.in.local-retail.bap","timestamp":"2021-09-18T18:59:15+05:30"},"message":{"intent":{"item":{"descriptor":{"name":"Upma"}},"fulfillment":{"end":{"location":{"gps":"19.035182168032843,72.86099418347541"}}}}}}*

**Signing String**

*(created): 1631971756*

*(expires): 1631971816*

*digest: BLAKE-512=oYyfESkYs8o+0q9n5hhJTQxhMLv5+mXr8JprCCOEoO3tbs5bpK8V8Fq3/6QcMqs+PrKZYCLo4fwrllSCbzU+7Q==*

*(You can use* [*https://8gwifi.org/MessageDigest.jsp*](https://8gwifi.org/MessageDigest.jsp) *to check your hash computation)*

**Request Headers**

Format for auth header is:

Signature keyId="{subscriber\_id}|{unique\_key\_id}|{algorithm}",algorithm="ed25519",created=1636463446,expires=1636464046,headers="(created) (expires) digest",signature="JQsYBjkQmMNFPzQmbG4aMANR/kISpC2jYY049TiaZ0PixPLZo5poMPILTubdi/kdtx3Is3oTx5W/C5ZwY0KcAg=="

Example:

*"Authorization":"Signature keyId=\"beckn-one.succinct.in.local-retail.bap|beckn-one.succinct.in.local-retail.bap.k1|35|ed25519\",algorithm=\"ed25519\",created=\"1631971756\",expires=\"1631971816\",headers=\"(created)(expires)digest\",signature=\"ShhKro8hebuNAWGg0JT3tyhNhtbiA3HP9qgkzi0IKFAA6uJUYbCRwqoS5a4hGDaB\/dXH8UiEnNUySZJzz0ChAg==\"",*

*"Accept":"application\/json",*

*"Content-Type":"application\/json"*

1. **Code example for the export format of the public key**

There are fundamentally 2 formats in which Ed25519 & X25519 Keys are exported.

1. Raw (simply base64 encoded of the raw bytes)
2. Pem (Typically used by JCA/JCE here it is bas64 encoded of a X509encoded public key )

\* If you do an online key pair generation for Ed25519, you will generally get in raw format. Both these formats are interconvertible. However since Raw is more prevalent , we could use the same for our communication with the registry.

Below link shows java code for various beckn usecases.

<https://github.com/venkatramanm/beckn-sdk-java/blob/master/src/test/java/in/succinct/beckn/SampleUseCase.java>

1. **Which algorithm will BAP use for decrypting Challenge String?**

*In general, X25519 Key Agreement algorithm is used to derive a symmetric AES key for encryption and decryption between any two parties.*

*We will use it between registry and participants to encrypt and decrypt challenges.*

*testEncryption function in the link below shows how to use X25519 to derive AES encryption key by any two parties.*

[*https://github.com/venkatramanm/beckn-sdk-java/blob/ce63cf11e8a4fbd358cf05b338414a157400aac8/src/test/java/in/succinct/beckn/SampleUseCase.java#L103*](https://github.com/venkatramanm/beckn-sdk-java/blob/ce63cf11e8a4fbd358cf05b338414a157400aac8/src/test/java/in/succinct/beckn/SampleUseCase.java#L103)

1. **Do we need to add Base64(XEd25519(BLAKE-512(signing string))) or do we only add the output with signed string using xed25519?**

We decided to sign the signing String directly instead of computing another hash

So it would be base64(xed25519(signing string))

1. **Please Share the format of the signing string with an example?**

Refer to point 3.

1. **Is subscriber\_id unique across the network for a BAP or is it the key\_id?**

subscriber\_id is unique across a network from the point of view of **lookup**. key\_id

is used for getting a particular key

1. **The bpp\_id or bap\_id , in the context of the payload is the same as key\_id right?**

key\_id is different from bap\_id or bpp\_id. If a subscriber uploads more than one key on a registry, the key\_id will be the unique identifier of a public key of a subscriber on a network

1. **Should we validate the bpp\_id or bap\_id in the context against the Authorization ?**

When BPP receives a request, If bap\_id in context does not match bap\_id in signature, the BPP must return a 401 unauthorised response

When BAP receives a request, If bpp\_id in context does not match bpp\_id in signature, the BAP must return a 401 unauthorised response

1. **What cipher is used to encrypt a challenge? We need to know it to be able to decrypt the challenge.**

The encryption would be AES but it is derived using the key agreement mechanism of x25519.

Every participant registers with the registry 2 public keys . An ed25519 and x25519 and would keep the private key with them

Registry would use its Private key and participant's public key to derive the aes key for encrypting the challenge.

Participants will use their own Private key and registry's public key to derive the same AES key that was used for encryption. And decrypt the challenge using this key.

X25519 could be also used to encrypt all interactions with every pair of participants in non https scenarios. This use case does not concern this group for now.

1. **As per the received sample use case code from Venky, as per that actor (BAP,BPP,BG) needs to maintain 2 private keys for decryption of challenge string in on\_subscribe call?**

As per the response by Venky, there are two formats that can be used to represent a public key. There is only one key.

1. **If subscriber A needs to get the public key of Subscriber B when doing a transaction then does it need to call lookup API to check whether Subscriber B is active?**

The **lookup** call should only return the subscriber records that have status = “SUBSCRIBED”.

1. **Does the lookup API need to be called before calling any APIs?**

Lookup is an independent API call that can be called anytime by the subscriber. Depends on the need. If a subscriber receives an API with a keyId that is not present in the local cached copy, then lookup may need to be called.