jndn - Task #4982

Error in sending/receiving parameters in Interest using setApplicationParameters

08/14/2019 02:20 PM - Ritik kumar

Status:	Closed		**Start date:**	08/14/2019
Priority:	Urgent		**Due date:**	
Assignee:			**% Done:**	0%
Category:			**Estimated time:**	0.00 hour
Target version:				

Description

Trying to send an interest using the Java client, to a Cpp Server listening to the same prefix, I got the error,

```
ERROR: ParametersSha256DigestComponent does not match the SHA-256 of Interest parameters
```

This behaviour is observed when using the method interest.setApplicationParameters with non-empty Blob.

Meanwhile listening as a Java Server worked fine.

History

#1 - 08/14/2019 02:28 PM - Jeff Thompson

It looks like Cpp Server is written with ndn-cxx which is enforcing a check of the parameters.
Instead of setApplicationParameters, please try appendParametersDigestToName().
https://github.com/named-data/jndn/blob/master/src/net/named_data/jndn/Interest.java#L680

appendParametersDigestToName() was written according to the first spec of computing the Parameters digest. (From what I undersatand the spec is not finalized. I don't know which spec is used by your version of ndn-cxx.)

#2 - 08/14/2019 02:52 PM - Ritik kumar

I have used that here (uncomment!). But even after that, the error continues.
I have built the latest commit of ndn-cxx.

#3 - 08/14/2019 03:17 PM - Jeff Thompson

There have been many changes to ndn-cxx for ApplicationParameters since the last release. I'm still waiting for the new digest calculation spec to be finalized.

In the mean time, can you call Interest::setAutoCheckParametersDigest(false) ?

https://github.com/named-data/ndn-cxx/blob/2b0cc7bf019ed60ef8094aba11db0ef1d7d2015b/ndn-cxx/interest.hpp#L369

#4 - 08/14/2019 03:20 PM - Jeff Thompson

(Call it in the Cpp Server app which is throwing the error.)

#5 - 08/15/2019 04:55 AM - Ritik kumar

Thanks for the help, the Cpp server doesn't crash now but there's a new problem!
The received Block on the Cpp Server gets a size 2 more than that of the Blob sent from the Java Client. Consequently, this results in parsing errors.
(I have updated the files on the link to print this difference)

#6 - 08/15/2019 05:11 AM - Jeff Thompson

Can you please post the two different printouts so that I don't have to compile and run your code?

#7 - 08/15/2019 05:32 AM - Ritik kumar

- File Screenshot from 2019-08-15 17-58-27.png added

- File Screenshot from 2019-08-15 17-56-01.png added

For the Server

and for the Client

```
✗ ritik@rk-desktop  /media/ritik/Ritik/cse_work/ndn/files  java -cp .:jndn-0.22.jar:json.jar Client
Sending blob: {"name":"ritik"}
Sending blob with size: 18
interest: /ndn/CA/_PROBE/params-sha256=875b01c6710da3bdcff3a15e457ad7f91cc76ca94b583b7813744e010693e1e0
Timeout
```

#8 - 08/15/2019 06:15 AM - Jeff Thompson

Can you print the hex of the Received Block? I suspect that it is the entire TLV of the ApplicationParameters, including the type and length bytes 0x24 0x12. You need to use the correct ndn-cxx code to just get the value from the Received Block TLV.

Also, why do you use paramFromJson? The parameters are already encoded in a JSON string (of 16 bytes). You don't need to encode them again into a TLV. You can just put the JSON string as the value of the ApplicationParameters.

https://gist.github.com/dev-ritik/d5ba9ea2bbebad0a2bfd9cb0fe03bb0f#file-client-java-L27

Instead of

```
Blob blob = paramFromJson(jo.toString());
```

just use

```
Blob blob = new Blob(jo.toString().getBytes());
```

#9 - 08/16/2019 02:44 PM - Ritik kumar

Thanks for the prompt help, the error was fixed following what you suggested above.

#10 - 08/20/2019 10:12 AM - Jeff Thompson

- Status changed from New to Closed

You're welcome. Closing.

Files

Screenshot from 2019-08-15 17-58-27.png	32.7 KB	08/15/2019	Ritik kumar
Screenshot from 2019-08-15 17-56-01.png	26.2 KB	08/15/2019	Ritik kumar