

HADOOP-12869

dapengsun

2016-03-03

Contents

Chapter 1

Root issue HADOOP-12869

1.1 Summary

CryptoInputStream#read() may return incorrect result

1.2 Description

Here is the comment of {{FilterInputStream#read()}}:

```
/**
 * Reads the next byte of data from this input stream. The value
 * byte is returned as an <code>int</code> in the range
 * <code>0</code> to <code>255</code>. If no byte is available
 * because the end of the stream has been reached, the value
 * <code>-1</code> is returned. This method blocks until input data
 * is available, the end of the stream is detected, or an exception
 * is thrown.
 * <p>
 * This method
 * simply performs <code>in.read()</code> and returns the result.
 *
 * @return      the next byte of data, or <code>-1</code> if the end
of the
 *              stream is reached.
 * @exception   IOException if an I/O error occurs.
 * @see         java.io.FilterInputStream#in
 */
public int read() throws IOException {
    return in.read();
}
```

Here is the implementation of {{CryptoInputStream#read()}} in Hadoop Common:

```
@Override
public int read() throws IOException {
    return (read(oneByteBuf, 0, 1) == -1) ? -1 : (oneByteBuf[0] & 0xff);
}
```

The return value of `{{read(oneByteBuf, 0, 1)}}` maybe 1, -1 and 0:

For `{{1}}`: we should return the content of `{{oneByteBuf}}`

For `{{-1}}`: we should return `{{-1}}` to stand for the end of stream

For `{{0}}`: it means we didn't get decryption data back and it is not the end of the stream, we should continue to decrypt the stream. But it return `{{0}}` on `{{read()}}` in current implementation, it means the decrypted content is `{{0}}` and it is incorrect.

1.3 Attachments

1. [HADOOP-12869.001.patch](#)
2. [HADOOP-12869.002.patch](#)

1.4 Comments

1. **dapengsun:** Added [~dian.fu] who investigated this issue together with me.

2. **hadoopqa:** | (x) *{color:red}-1 overall{color}* |
\\
\\
|| Vote || Subsystem || Runtime || Comment ||
| {color:blue}0{color} | {color:blue} reexec {color} | {color:blue} 0m 10s {color} | {color:blue}
Docker mode activated. {color} |
| {color:green}+1{color} | {color:green} @author {color} | {color:green} 0m 0s {color} |
{color:green} The patch does not contain any @author tags. {color} |
| {color:red}-1{color} | {color:red} test4tests {color} | {color:red} 0m 0s {color} | {color:red}
The patch doesn't appear to include any new or modified tests. Please justify why no new
tests are needed for this patch. Also please list what manual steps were performed to verify
this patch. {color} |
| {color:green}+1{color} | {color:green} mvninstall {color} | {color:green} 6m 57s {color} |
{color:green} trunk passed {color} |
| {color:green}+1{color} | {color:green} compile {color} | {color:green} 6m 1s {color} |
{color:green} trunk passed with JDK v1.8.0_72 {color} |
| {color:green}+1{color} | {color:green} compile {color} | {color:green} 6m 53s {color} |
{color:green} trunk passed with JDK v1.7.0_95 {color} |
| {color:green}+1{color} | {color:green} checkstyle {color} | {color:green} 0m 21s {color} |
{color:green} trunk passed {color} |
| {color:green}+1{color} | {color:green} mvnsite {color} | {color:green} 1m 6s {color} |
{color:green} trunk passed {color} |
| {color:green}+1{color} | {color:green} mvneclipse {color} | {color:green} 0m 13s {color} |
{color:green} trunk passed {color} |
| {color:green}+1{color} | {color:green} findbugs {color} | {color:green} 1m 35s {color} |
{color:green} trunk passed {color} |
| {color:green}+1{color} | {color:green} javadoc {color} | {color:green} 0m 53s {color} |
{color:green} trunk passed with JDK v1.8.0_72 {color} |
| {color:green}+1{color} | {color:green} javadoc {color} | {color:green} 1m 2s {color} |
{color:green} trunk passed with JDK v1.7.0_95 {color} |
| {color:green}+1{color} | {color:green} mvninstall {color} | {color:green} 0m 41s {color} |
{color:green} the patch passed {color} |
| {color:green}+1{color} | {color:green} compile {color} | {color:green} 5m 58s {color} |
{color:green} the patch passed with JDK v1.8.0_72 {color} |

```

| {color:green}+1{color} | {color:green} javac {color} | {color:green} 5m 58s {color} | {color:green}
the patch passed {color} |
| {color:green}+1{color} | {color:green} compile {color} | {color:green} 6m 46s {color} |
{color:green} the patch passed with JDK v1.7.0_95 {color} |
| {color:green}+1{color} | {color:green} javac {color} | {color:green} 6m 46s {color} | {color:green}
the patch passed {color} |
| {color:green}+1{color} | {color:green} checkstyle {color} | {color:green} 0m 21s {color} |
{color:green} the patch passed {color} |
| {color:green}+1{color} | {color:green} mvnsite {color} | {color:green} 1m 0s {color} |
{color:green} the patch passed {color} |
| {color:green}+1{color} | {color:green} mvneclipse {color} | {color:green} 0m 13s {color} |
{color:green} the patch passed {color} |
| {color:green}+1{color} | {color:green} whitespace {color} | {color:green} 0m 0s {color} |
{color:green} Patch has no whitespace issues. {color} |
| {color:green}+1{color} | {color:green} findbugs {color} | {color:green} 1m 52s {color} |
{color:green} the patch passed {color} |
| {color:green}+1{color} | {color:green} javadoc {color} | {color:green} 0m 55s {color} |
{color:green} the patch passed with JDK v1.8.0_72 {color} |
| {color:green}+1{color} | {color:green} javadoc {color} | {color:green} 1m 4s {color} |
{color:green} the patch passed with JDK v1.7.0_95 {color} |
| {color:green}+1{color} | {color:green} unit {color} | {color:green} 7m 42s {color} | {color:green}
hadoop-common in the patch passed with JDK v1.8.0_72. {color} |
| {color:green}+1{color} | {color:green} unit {color} | {color:green} 7m 53s {color} | {color:green}
hadoop-common in the patch passed with JDK v1.7.0_95. {color} |
| {color:green}+1{color} | {color:green} asflicense {color} | {color:green} 0m 24s {color} |
{color:green} Patch does not generate ASF License warnings. {color} |
| {color:black}{color} | {color:black} {color} | {color:black} 61m 4s {color} | {color:black}
{color} |
\\
\\
|| Subsystem || Report/Notes ||
| Docker | Image:yetus/hadoop:0ca8df7 |
| JIRA Patch URL | https://issues.apache.org/jira/secure/attachment/12791088/HADOOP-
12869.001.patch |
| JIRA Issue | HADOOP-12869 |
| Optional Tests | asflicense compile javac javadoc mvninstall mvnsite unit findbugs check-
style |
| uname | Linux 5b47f0573f72 3.13.0-36-lowlatency #63-Ubuntu SMP PREEMPT Wed Sep
3 21:56:12 UTC 2014 x86_64 x86_64 x86_64 GNU/Linux |
| Build tool | maven |
| Personality | /testptch/hadoop/patchprocess/precommit/personality/provided.sh |
| git revision | trunk / 27941a1 |
| Default Java | 1.7.0_95 |
| Multi-JDK versions | /usr/lib/jvm/java-8-oracle:1.8.0_72 /usr/lib/jvm/java-7-openjdk-
amd64:1.7.0_95 |
| findbugs | v3.0.0 |
| JDK v1.7.0_95 Test Results | https://builds.apache.org/job/PreCommit-HADOOP-Build/8767/testReport/
|
| modules | C: hadoop-common-project/hadoop-common U: hadoop-common-project/hadoop-
common |
| Console output | https://builds.apache.org/job/PreCommit-HADOOP-Build/8767/console
|
| Powered by | Apache Yetus 0.3.0-SNAPSHOT http://yetus.apache.org |

```

This message was automatically generated.

3. **dapengsun:** Updated patch to make logic simple

4. **hadoopqa:** | (x) *{color:red}-1 overall{color}* |

\\
\\

```
|| Vote || Subsystem || Runtime || Comment ||
| {color:blue}0{color} | {color:blue} reexec {color} | {color:blue} 0m 12s {color} | {color:blue}
Docker mode activated. {color} |
| {color:green}+1{color} | {color:green} @author {color} | {color:green} 0m 0s {color} |
{color:green} The patch does not contain any @author tags. {color} |
| {color:red}-1{color} | {color:red} test4tests {color} | {color:red} 0m 0s {color} | {color:red}
The patch doesn't appear to include any new or modified tests. Please justify why no new
tests are needed for this patch. Also please list what manual steps were performed to verify
this patch. {color} |
| {color:green}+1{color} | {color:green} mvninstall {color} | {color:green} 6m 37s {color} |
{color:green} trunk passed {color} |
| {color:green}+1{color} | {color:green} compile {color} | {color:green} 5m 52s {color} |
{color:green} trunk passed with JDK v1.8.0_72 {color} |
| {color:green}+1{color} | {color:green} compile {color} | {color:green} 6m 43s {color} |
{color:green} trunk passed with JDK v1.7.0_95 {color} |
| {color:green}+1{color} | {color:green} checkstyle {color} | {color:green} 0m 20s {color} |
{color:green} trunk passed {color} |
| {color:green}+1{color} | {color:green} mvnsite {color} | {color:green} 1m 3s {color} |
{color:green} trunk passed {color} |
| {color:green}+1{color} | {color:green} mvneclipse {color} | {color:green} 0m 13s {color} |
{color:green} trunk passed {color} |
| {color:green}+1{color} | {color:green} findbugs {color} | {color:green} 1m 32s {color} |
{color:green} trunk passed {color} |
| {color:green}+1{color} | {color:green} javadoc {color} | {color:green} 0m 53s {color} |
{color:green} trunk passed with JDK v1.8.0_72 {color} |
| {color:green}+1{color} | {color:green} javadoc {color} | {color:green} 1m 3s {color} |
{color:green} trunk passed with JDK v1.7.0_95 {color} |
| {color:green}+1{color} | {color:green} mvninstall {color} | {color:green} 0m 40s {color} |
{color:green} the patch passed {color} |
| {color:green}+1{color} | {color:green} compile {color} | {color:green} 5m 49s {color} |
{color:green} the patch passed with JDK v1.8.0_72 {color} |
| {color:green}+1{color} | {color:green} javac {color} | {color:green} 5m 49s {color} | {color:green}
the patch passed {color} |
| {color:green}+1{color} | {color:green} compile {color} | {color:green} 6m 42s {color} |
{color:green} the patch passed with JDK v1.7.0_95 {color} |
| {color:green}+1{color} | {color:green} javac {color} | {color:green} 6m 42s {color} | {color:green}
the patch passed {color} |
| {color:red}-1{color} | {color:red} checkstyle {color} | {color:red} 0m 21s {color} | {color:red}
hadoop-common-project/hadoop-common: patch generated 2 new + 32 unchanged - 0 fixed
= 34 total (was 32) {color} |
| {color:green}+1{color} | {color:green} mvnsite {color} | {color:green} 1m 0s {color} |
{color:green} the patch passed {color} |
| {color:green}+1{color} | {color:green} mvneclipse {color} | {color:green} 0m 14s {color} |
{color:green} the patch passed {color} |
| {color:green}+1{color} | {color:green} whitespace {color} | {color:green} 0m 0s {color} |
```

```

{color:green} Patch has no whitespace issues. {color} |
| {color:green}+1{color} | {color:green} findbugs {color} | {color:green} 1m 50s {color} |
{color:green} the patch passed {color} |
| {color:green}+1{color} | {color:green} javadoc {color} | {color:green} 0m 52s {color} |
{color:green} the patch passed with JDK v1.8.0_72 {color} |
| {color:green}+1{color} | {color:green} javadoc {color} | {color:green} 1m 1s {color} |
{color:green} the patch passed with JDK v1.7.0_95 {color} |
| {color:green}+1{color} | {color:green} unit {color} | {color:green} 6m 48s {color} | {color:green}
hadoop-common in the patch passed with JDK v1.8.0_72. {color} |
| {color:green}+1{color} | {color:green} unit {color} | {color:green} 7m 11s {color} | {color:green}
hadoop-common in the patch passed with JDK v1.7.0_95. {color} |
| {color:green}+1{color} | {color:green} asflicense {color} | {color:green} 0m 22s {color} |
{color:green} Patch does not generate ASF License warnings. {color} |
| {color:black}{color} | {color:black} {color} | {color:black} 58m 26s {color} | {color:black}
{color} |
\\
\\
|| Subsystem || Report/Notes ||
| Docker | Image:yetus/hadoop:0ca8df7 |
| JIRA Patch URL | https://issues.apache.org/jira/secure/attachment/12791107/HADOOP-
12869.002.patch |
| JIRA Issue | HADOOP-12869 |
| Optional Tests | asflicense compile javac javadoc mvninstall mvnsite unit findbugs check-
style |
| uname | Linux 84991624b8ac 3.13.0-36-lowlatency #63-Ubuntu SMP PREEMPT Wed Sep
3 21:56:12 UTC 2014 x86_64 x86_64 x86_64 GNU/Linux |
| Build tool | maven |
| Personality | /testptch/hadoop/patchprocess/precommit/personality/provided.sh |
| git revision | trunk / 27941a1 |
| Default Java | 1.7.0_95 |
| Multi-JDK versions | /usr/lib/jvm/java-8-oracle:1.8.0_72 /usr/lib/jvm/java-7-openjdk-
amd64:1.7.0_95 |
| findbugs | v3.0.0 |
| checkstyle | https://builds.apache.org/job/PreCommit-HADOOP-Build/8771/artifact/patchprocess/diff-
checkstyle-hadoop-common-project_hadoop-common.txt |
| JDK v1.7.0_95 Test Results | https://builds.apache.org/job/PreCommit-HADOOP-Build/8771/testReport/
|
| modules | C: hadoop-common-project/hadoop-common U: hadoop-common-project/hadoop-
common |
| Console output | https://builds.apache.org/job/PreCommit-HADOOP-Build/8771/console
|
| Powered by | Apache Yetus 0.3.0-SNAPSHOT http://yetus.apache.org |

```

This message was automatically generated.

5. **busbey:** Please add a test that shows the issue.

The docs for the `read` method we call claim <https://docs.oracle.com/javase/7/docs/api/java/io/FilterInputStream.L>

¹ `public int read(byte[] b,`

```

2         int off,
3         int len)
4         throws IOException
5
6 Reads up to len bytes of data from this input stream into an array of bytes. If len
   is not zero, the method blocks until some input is available; otherwise, no
   bytes are read and 0 is returned.

```

We pass in a len of 1. If FIS is blocking until "some" input is available, shouldn't that mean it has to have ≥ 1 byte available?

6. **busbey:** FYI, I've been trying to write a test that reproduces this problem. So far I haven't managed to get anything other than correct results and -1s.
7. **cmccabe:** The documentation for `{{InputStream#read}}` says:

`{{If len is zero, then no bytes are read and 0 is returned; otherwise, there is an attempt to read at least one byte. If no byte is available because the stream is at end of file, the value -1 is returned; otherwise, at least one byte is read and stored into b.}}`

It seems clear that an `{{InputStream}}` which returns `{{0}}` when `{{len != 0}}` is not following the spec. Of course, we might have such an implementation somewhere in Hadoop, but it would be a bug. Perhaps we should throw an exception rather than looping, to make users aware of the bug in the wrapped `{{InputStream}}`. There is no guarantee that looping is the right thing to do in the case of a buggy `{{InputStream}}` (say, one which returns 0 on EOF rather than -1).

I don't see how we can have a unit test for this, unless we deliberately mock up a buggy `{{InputStream}}`.

8. **vinodkv:** [[~dapengsun](#)] / [[~busbey](#)] / [[~cmccabe](#)], do you think 2.7.3 (later in the week) should wait for this? Is this any close to getting ready in that time frame? Please let me know either ways, tx.
9. **busbey:** I don't think it needs to wait. As Colin mentioned, this is more a symptom than the root problem. Unless we know what the root issue is it's too hard to tell the severity or how long it's been around.
10. **vinodkv:** Okay, then I'll drop the target-versions. Once there is a root-cause, please add it back to the appropriate release-trains. Tx.