BEEM - Android XMPP - Bug #449

Beem Authentication Bug

08/12/2012 12:39 PM - Anonymous

Status:	Closed	**Start date:**	08/12/2012
Priority:	Normal	**Due date:**	
Assignee:	Frédéric Barthéléry	**% Done:**	0%
Category:	Communication	**Estimated time:**	0.00 hour
Target version:	0.1.8		
Affected version:	0.1.7	**Locale:**	

Description

Hi All,
So my JID is either in the form of 555@sub.myname.net or just 555 (I can't tell) and I'm trying to log into host sub.myname.net with a slightly modified xmpp server, but I don't think that's the problem and I can't seem to figure out what's wrong.. Any help/guidance would be apprciated..

When I set the JID to 555 in Beem (with the option to "Use my full JID as username" in either setting), I see a convo with my XMPP server that is like:

```
<stream:stream to="555" xmlns="jabber:client" xmlns:stream="http://etherx.jabber.org/streams" vers
ion="1.0">
<stream:stream xmlns='jabber:client' xmlns:stream='http://etherx.jabber.org/streams' xml:lang='en'
 id='0314dcea-1729-4f45-8272-820fb14594f3' from='555' version='1.0'><stream:error><host-unknown xm
lns="urn:ietf:params:xml:ns:xmpp-streams"/></stream:error></stream:stream>
<presence id="eU8sz-14" type="unavailable"></presence>
</stream:stream>
```

When I set my JID to the form of 555@sub.myname.net with "Use my full JID as username" **off** I see something like:

```
<stream:stream xmlns="jabber:client" xmlns:stream="http://etherx.jabber.org/streams" to="sub.mynam
e.net" version="1.0"/>
<stream:stream xmlns='jabber:client' xmlns:stream='http://etherx.jabber.org/streams' xml:lang='en'
 id='1c2d492b-7c31-42d2-8946-bf268981dbeb' from='sub.myname.net' version='1.0'>
<stream:features>
  <mechanisms xmlns="urn:ietf:params:xml:ns:xmpp-sasl">
    <mechanism>PLAIN</mechanism>
  </mechanisms>
<auth xmlns="urn:ietf:params:xml:ns:xmpp-sasl" mechanism="PLAIN">NTU1NTU1VTlKTnhGb2JCRmp5RW1FSg==<
/auth>
<failure xmlns="urn:ietf:params:xml:ns:xmpp-sasl"><not-authorized/></failure>
```

There, "NTU1NTU1VTlKTnhGb2JCRmp5RW1FSg==" is the base64 encoding of "555555U9JNxFobBFjyEmEJ", with 555 being my JID before the @ and "U9JNxFobBFjyEmEJ" being my example password.

When I set my JID to the form of 555@sub.myname.net with "Use my full JID as username" **on** I see something identical to the previous, except the base64 encoding translates to: "555@sub.myname.net<NULL>555@sub.myname.net <NULL>U9JNxFobBFjyEmEJ" (WITH delimiters).

Finally, when I use my Desktop PC and Jabber, using all normal settings, I see something like:

```
<stream:stream xmlns="jabber:client" xmlns:stream="http://etherx.jabber.org/streams" to="sub.mynam
e.net" version="1.0"/>
<stream:stream xmlns='jabber:client' xmlns:stream='http://etherx.jabber.org/streams' xml:lang='en'
 id='24672be6-d751-4928-a750-fbc20cce6e5f' from='sub.myname.net' version='1.0'>
<stream:features>
  <mechanisms xmlns="urn:ietf:params:xml:ns:xmpp-sasl">
    <mechanism>PLAIN</mechanism>
  </mechanisms>
</stream:features>
```

```
<auth xmlns="urn:ietf:params:xml:ns:xmpp-sasl" xmlns:ga="http://www.google.com/talk/protocol/auth"
 mechanism="PLAIN" client-uses-full-bind-result="true">NTU1VTlKTnhGb2JCRmp5RW1FSg==</auth>
<success xmlns="urn:ietf:params:xml:ns:xmpp-sasl"/>
```

This base64 encoding translates simply into "<NULL>555<NULL>U9JNxFobBFjyEmEJ" with no domain and no duplicated JID. This is the only thing that authenticates successfully against my server.

(Note that I also don't understand how the server is supposed to discern between the end of the JID and the beginning of the password with the auth stanzas from Beem, since they lack the NULL byte delimieters, but maybe I'm missing something..)

Please advise.
Thanks!
Mike

(Note: I've gone and replaced all usernames, passwords and domains for privacy reasons.)

PS:
Looking at RFC 4616 (the spec for PLAIN SASL AUTH), the format of the auth XMPP stanza's base64-encoded contents should be "authzid<NULL>authcid<NULL>password", where the relationship between authzid and authcid are described as follows (sorry if this is remedial.. these are new terms to me):

```
Upon receipt of the message, the server will verify the presented (in the message) authentication
identity (authcid) and password (passwd) with the system authentication database, and it will veri
fy that the authentication credentials permit the client to act as the (presented or derived) auth
orization identity (authzid).
```

Looking at my code, it looks like authzid is disregarded. So it seems this whole problem is fixed by allowing some way for the transmitted base64-encoded string to read "noonecares<NULL>555<NULL>U9JNxFobBFjyEmEJ"

History

#1 - 08/12/2012 12:42 PM - Mike Vastola

Oops. Looks like my session got timed out while creating this bug report. This is mine. Can someone with superpowers make me the owner? :-p (Or just leave it if it doesn't really have ramifications on this reporting system.. I don't know.)

Thanks!

#2 - 08/12/2012 06:09 PM - Frédéric Barthéléry

Anonyme wrote:

> When I set the JID to 555 in Beem (with the option to "Use my full JID as username" in either setting), I see a convo with my XMPP server that is like:

The jid is in the form user@domain so just 555 is wrong. In this case 555 is considered as the domain, and the xmpp client try to connect to 555 which failed either with no connection at all or with the <host-unknow> error

> When I set my JID to the form of 555@sub.myname.net with "Use my full JID as username" **off** I see something like:
> [...]
> There, "NTU1NTU1VTlKTnhGb2JCRmp5RW1FSg==" is the base64 encoding of "555555U9JNxFobBFjyEmEJ", with 555 being my JID before the @ and "U9JNxFobBFjyEmEJ" being my example password.
>
> When I set my JID to the form of 555@sub.myname.net with "Use my full JID as username" **on** I see something identical to the previous, except the base64 encoding translates to: "555@sub.myname.net<NULL>555@sub.myname.net<NULL>U9JNxFobBFjyEmEJ" (WITH delimiters).

In Beem, the option "Use my full JID as username", only change the variable username. This is exactly the same code so you should have delimiters in both case. However, I just looked at the code with no real try as I do not have a server which supports only PLAIN SASL to test my assertion.

> Finally, when I use my Desktop PC and Jabber, using all normal settings, I see something like:
> [...]
>
> This base64 encoding translates simply into "<NULL>555<NULL>U9JNxFobBFjyEmEJ" with no domain and no duplicated JID. This is the only thing that authenticates successfully against my server.
> (Note that I also don't understand how the server is supposed to discern between the end of the JID and the beginning of the password with the auth stanzas from Beem, since they lack the NULL byte delimieters, but maybe I'm missing something..)

Regarding this you don't have to set the "Use my full JID as username" option to login with Beem.

> PS:
> Looking at RFC 4616 (the spec for PLAIN SASL AUTH), the format of the auth XMPP stanza's base64-encoded contents should be "authzid<NULL>authcid<NULL>password", where the relationship between authzid and authcid are described as follows (sorry if this is remedial.. these are new terms to me):
>
> [...]
>
> Looking at my code, it looks like authzid is disregarded. So it seems this whole problem is fixed by allowing some way for the transmitted base64-encoded string to read "noonecares<NULL>555<NULL>U9JNxFobBFjyEmEJ"

The authzid element is optional but the first <NULL> is mandatory. You should not set the noonecares because the server may use it to decide if you have the right to login as noonecares.

So this may be a bug in Beem with the PLAIN SASL mechanism. I will comment on this when I will be able to investigate more.

#3 - 08/12/2012 10:23 PM - Mike Vastola

Thanks! We're totally on the same page and I agree completely with your assessment.

Frédéric Barthéléry wrote:

> The authzid element is optional but the first <NULL> is mandatory. You should not set the noonecares because the server may use it to decide if you have the right to login as noonecares.

Point taken. That should have been "my server doesn't care". Also, re the "you should not set" part, I'm sorry to say I put in my best effort (a couple of hours) trying to track down the place in the code that needed patching so that I could be helpful and submit my own fix along with this report, but I came up emptyhanded. (Thus the session timeout in the interim.) As a result, I think I have to leave it to you guys to go about not setting the noonecares. :-\

#4 - 10/08/2012 11:31 PM - Frédéric Barthéléry

- Status changed from New to Resolved

- Assignee set to Frédéric Barthéléry

I have got some time to investigate a little on this.
In Beem we always use the "authzid<NULL>authcid<NULL>password" form for SASL plain authentication. This is done in a very internal things coming from aSmack (actually it is an sasl implementation from Apache Qpid). As this behaviour is correct, we will not change it.
Hope this help you.

#5 - 10/30/2012 12:05 AM - Frédéric Barthéléry

- Target version set to 0.1.9

#6 - 10/30/2012 12:55 AM - Mike Vastola

Hi Frédéric,
I seem to have missed your update earlier this month (but I just saw this now). I'm not sure I understand your comment 4. Could you clarify (as I'd like to learn and patch my server if there's a bug on my end..)? Maybe I'm forgetting exactly what the problem was because it was a while ago, but I could have sworn the issue was that when "Use my full JID as username" was turned off, the NULL bytes were mysteriously missing from the authentication strings Beem was transmitting after base64-decoding it..

We are agreed that "authzid<NULL>authcid<NULL>password" is correct.

Thanks,
Mike

#7 - 10/30/2012 07:58 AM - Frédéric Barthéléry

Both "authzid<NULL>authcid<NULL>password" and "<NULL>authcid<NULL>password" forms are correct. In Beem, we **always** use the first form event if "authzid==authcid" . This is the behavior you describe in the bug report.
But Gajim uses the second form if "authzid==authcid".

#8 - 10/30/2012 11:19 AM - Mike Vastola

I think we misunderstood each other (aka my bug report was flawed/unclear)... But I officially have no idea what's going on any more. I actually had the misfortune of losing my Droid two days ago, but I just tried Beem 0.1.7 on my PC's Android emulator logging into my XMPP server (which hasn't been updated) and it suddenly worked. So I give up. But in a good way, I guess, because I can presumably use Beem now once I get my new Droid. :-)

Thanks for looking into this!

#9 - 10/30/2012 01:36 PM - Frédéric Barthéléry

Good news then :-)

#10 - 11/12/2012 09:34 PM - Frédéric Barthéléry

- Target version changed from 0.1.9 to Dev

#11 - 01/06/2013 10:49 PM - Frédéric Barthéléry

- Status changed from Resolved to Closed

#12 - 02/24/2013 09:12 PM - Frédéric Barthéléry

- Target version changed from Dev to 0.1.8