Smart Proxy - Bug #12555

Only first FreeIPA XMLRPC call succeeds Foreman proxy 1.10 and FreeIPA, version: 4.1.4

11/20/2015 02:33 PM - Michael Eklund

Status:	Closed		
Priority:	Normal		
Assignee:			
Category:	Realm		
Target version:	1.10.1		
Difficulty:		**Fixed in Releases:**	
Triaged:		**Found in Releases:**	
Bugzilla link:	1305402	**Red Hat JIRA:**	
Pull request:	https://github.com/theforeman/smart-proxy/pull/353		

Description

```
D, [2015-11-20T14:17:00.951816 #10124] DEBUG -- : verifying remote client 1.1.1.1 against trusted_
hosts ["cfg01.atl.XXXX.net"]
I, [2015-11-20T14:17:01.019800 #10124]  INFO -- : freeipa: realm keytab is '/etc/foreman-proxy/fre
eipa.keytab' and using principal 'XXXX@XXXX.NET'
I, [2015-11-20T14:17:01.020059 #10124]  INFO -- : freeipa: realm XXXX.NET
I, [2015-11-20T14:17:01.020636 #10124]  INFO -- : freeipa: server is https://ipa.XXXX.net/ipa/xml
I, [2015-11-20T14:17:01.021306 #10124]  INFO -- : Requesting credentials for Kerberos principal XX
XX@XXXX.NET using keytab /etc/foreman-proxy/freeipa.keytab
D, [2015-11-20T14:17:01.059031 #10124] DEBUG -- : Kerberos credential cache initialised with princ
ipal: XXXX@XXXX.NET
I, [2015-11-20T14:17:02.301035 #10124]  INFO -- : Attempting to host_add test2.atl.XXXX.net in Fre
eIPA
D, [2015-11-20T14:17:02.301183 #10124] DEBUG -- : {:setattr=>[], :random=>1, :force=>1}
E, [2015-11-20T14:17:02.322459 #10124] ERROR -- : Authorization failed.
HTTP-Error: 401 Unauthorized
D, [2015-11-20T14:17:02.322550 #10124] DEBUG -- : /usr/lib/ruby/1.9.1/xmlrpc/client.rb:547:in `do_
rpc'
/usr/lib/ruby/1.9.1/xmlrpc/client.rb:420:in `call2'
/usr/lib/ruby/1.9.1/xmlrpc/client.rb:410:in `call'
/usr/share/foreman-proxy/modules/realm/freeipa.rb:103:in `create'
/usr/share/foreman-proxy/modules/realm/realm_api.rb:28:in `block in <class:Api>'
/usr/lib/ruby/vendor_ruby/sinatra/base.rb:1541:in `call'
/usr/lib/ruby/vendor_ruby/sinatra/base.rb:1541:in `block in compile!'
/usr/lib/ruby/vendor_ruby/sinatra/base.rb:950:in `[]'
/usr/lib/ruby/vendor_ruby/sinatra/base.rb:950:in `block (3 levels) in route!'
/usr/lib/ruby/vendor_ruby/sinatra/base.rb:966:in `route_eval'
/usr/lib/ruby/vendor_ruby/sinatra/base.rb:950:in `block (2 levels) in route!'
/usr/lib/ruby/vendor_ruby/sinatra/base.rb:987:in `block in process_route'
/usr/lib/ruby/vendor_ruby/sinatra/base.rb:985:in `catch'
/usr/lib/ruby/vendor_ruby/sinatra/base.rb:985:in `process_route'
/usr/lib/ruby/vendor_ruby/sinatra/base.rb:948:in `block in route!'
/usr/lib/ruby/vendor_ruby/sinatra/base.rb:947:in `each'
/usr/lib/ruby/vendor_ruby/sinatra/base.rb:947:in `route!'
/usr/lib/ruby/vendor_ruby/sinatra/base.rb:1059:in `block in dispatch!'
/usr/lib/ruby/vendor_ruby/sinatra/base.rb:1041:in `block in invoke'
/usr/lib/ruby/vendor_ruby/sinatra/base.rb:1041:in `catch'
/usr/lib/ruby/vendor_ruby/sinatra/base.rb:1041:in `invoke'
/usr/lib/ruby/vendor_ruby/sinatra/base.rb:1056:in `dispatch!'
/usr/lib/ruby/vendor_ruby/sinatra/base.rb:882:in `block in call!'
/usr/lib/ruby/vendor_ruby/sinatra/base.rb:1041:in `block in invoke'
/usr/lib/ruby/vendor_ruby/sinatra/base.rb:1041:in `catch'
/usr/lib/ruby/vendor_ruby/sinatra/base.rb:1041:in `invoke'
/usr/lib/ruby/vendor_ruby/sinatra/base.rb:882:in `call!'
/usr/lib/ruby/vendor_ruby/sinatra/base.rb:870:in `call'
/usr/lib/ruby/vendor_ruby/rack/commonlogger.rb:33:in `call'
```

```
/usr/lib/ruby/vendor_ruby/sinatra/base.rb:212:in `call'
/usr/share/foreman-proxy/lib/proxy/log.rb:58:in `call'
/usr/lib/ruby/vendor_ruby/rack/protection/xss_header.rb:18:in `call'
/usr/lib/ruby/vendor_ruby/rack/protection/path_traversal.rb:16:in `call'
/usr/lib/ruby/vendor_ruby/rack/protection/json_csrf.rb:18:in `call'
/usr/lib/ruby/vendor_ruby/rack/protection/base.rb:50:in `call'
/usr/lib/ruby/vendor_ruby/rack/protection/base.rb:50:in `call'
/usr/lib/ruby/vendor_ruby/rack/protection/frame_options.rb:31:in `call'
/usr/lib/ruby/vendor_ruby/rack/nulllogger.rb:9:in `call'
/usr/lib/ruby/vendor_ruby/rack/head.rb:11:in `call'
/usr/lib/ruby/vendor_ruby/sinatra/showexceptions.rb:21:in `call'
/usr/lib/ruby/vendor_ruby/sinatra/base.rb:175:in `call'
/usr/lib/ruby/vendor_ruby/sinatra/base.rb:1949:in `call'
/usr/lib/ruby/vendor_ruby/sinatra/base.rb:1449:in `block in call'
/usr/lib/ruby/vendor_ruby/sinatra/base.rb:1726:in `synchronize'
/usr/lib/ruby/vendor_ruby/sinatra/base.rb:1449:in `call'
/usr/lib/ruby/vendor_ruby/rack/builder.rb:138:in `call'
/usr/lib/ruby/vendor_ruby/rack/urlmap.rb:65:in `block in call'
/usr/lib/ruby/vendor_ruby/rack/urlmap.rb:50:in `each'
/usr/lib/ruby/vendor_ruby/rack/urlmap.rb:50:in `call'
/usr/lib/ruby/vendor_ruby/rack/builder.rb:138:in `call'
/usr/lib/ruby/vendor_ruby/rack/handler/webrick.rb:60:in `service'
/usr/lib/ruby/1.9.1/webrick/httpserver.rb:138:in `service'
/usr/lib/ruby/1.9.1/webrick/httpserver.rb:94:in `run'
/usr/lib/ruby/1.9.1/webrick/server.rb:191:in `block in start_thread'
I, [2015-11-20T14:17:02.323003 #10124]  INFO -- : 1.1.1.1 - - [20/Nov/2015 14:17:02] "POST /realm/
XXXX.NET HTTP/1.1" 400 50 1.3738
'
```

Foreman/Foreman Proxy machine is Ubuntu 14, and is joined to the freeIPA realm.

If I mess with freeipa.rb and get a new token each time by doing this prior to each @ipa.call:

```
gssapi = GSSAPI::Simple.new(@ipa_server.host, "HTTP")
token = gssapi.init_context
@ipa.http_header_extra={ 'Authorization'=>"Negotiate #{strict_encode64(token)}",
                                 'Referer' => @ipa_server.to_s,
                                 'Content-Type' => 'text/xml; charset=utf-8'
                               }
```

then all calls work.

The only time this appears to be a problem is when multiple calls to the IPA server are issued. I am not enough of an expert on GSSAPI to know if there is additional negotiation needed after the first call, or if there is session data not being passed.

Associated revisions

Revision 07a6dae6 - 12/22/2015 10:55 AM - Dmitri Dolguikh

Fixes #12555: switched to session-based freeipa api

History

#1 - 11/20/2015 02:46 PM - Michael Eklund

IPA error logs

```
[Fri Nov 20 19:45:08.795455 2015] [wsgi:error] [pid 232] ipa: INFO: [xmlserver] XXXX@XXXX.NET: host_show(u'tes
t2.atl.XXXX.net', version=u'2.51'): NotFound
[Fri Nov 20 19:45:08.819009 2015] [auth_gssapi:error] [pid 423] [client 10.10.50.170:58792] gss_accept_sec_con
text() failed: [Unspecified GSS failure.  Minor code may provide more information (Request is a replay)], refe
rer: https://ipa.XXXX.net/ipa/xml
```

#2 - 11/20/2015 02:55 PM - Michael Eklund

probably related to IPA 4.1 Replay Protection

#3 - 11/23/2015 03:05 AM - Dominic Cleal

- Project changed from Foreman to Smart Proxy

- Category set to Realm

#4 - 11/26/2015 05:23 AM - Daniel O

Hi, any updates on this topic?

Same error with ipa-server 4.2 on rhel7.2 and foreman 1.9.2 on centos6.5.

#5 - 11/30/2015 09:35 AM - Davy Stoffel

Hi, same here too. rhel 7.2, ipa-server 4.2
Given workaround works too for us, please advise.

#6 - 11/30/2015 12:20 PM - Michael Eklund

here is my hack to make it send a new token on each call, this is most likely not the right way to fix this:

```
--- freeipa.rb    2015-11-30 12:14:10.374054744 -0500
+++ freeipa.rb.mod    2015-11-30 12:13:08.581208212 -0500
@@ -39,17 +39,12 @@
       if errors.empty?
         # Get krb5 token
         init_krb5_ccache Proxy::Realm::Plugin.settings.realm_keytab, Proxy::Realm::Plugin.settings.realm_prin
cipal
-        gssapi = GSSAPI::Simple.new(@ipa_server.host, "HTTP")
-        token = gssapi.init_context

         # FreeIPA API returns some nils, Ruby XML-RPC doesn't like this
         XMLRPC::Config.module_eval { const_set(:ENABLE_NIL_PARSER, true) }

         @ipa = XMLRPC::Client.new2(@ipa_server.to_s)
-        @ipa.http_header_extra={ 'Authorization'=>"Negotiate #{strict_encode64(token)}",
-                                 'Referer' => @ipa_server.to_s,
-                                 'Content-Type' => 'text/xml; charset=utf-8'
-                               }
+        set_auth()
       else
         raise Proxy::Realm::Error.new errors.join(", ")
       end
@@ -60,6 +55,7 @@
     end

     def find hostname
+      set_auth()
       @ipa.call("host_show", [hostname])
     rescue XMLRPC::FaultException => e
       if e.message =~ /not found/
@@ -68,6 +64,15 @@
         raise
       end
     end
+
+    def set_auth
+        gssapi = GSSAPI::Simple.new(@ipa_server.host, "HTTP")
+        token = gssapi.init_context
+        @ipa.http_header_extra={ 'Authorization'=>"Negotiate #{strict_encode64(token)}",
+                                 'Referer' => @ipa_server.to_s,
+                                 'Content-Type' => 'text/xml; charset=utf-8'
+                               }
+    end

     def create realm, params
       check_realm realm
@@ -91,6 +96,7 @@
           # disable it in order to revoke existing certs, keytabs, etc.
           if host["result"]["has_keytab"]
             logger.info "Attempting to disable host #{params[:hostname]} in FreeIPA"
+            set_auth()
             @ipa.call("host_disable", [params[:hostname]])
           end
         end
@@ -98,6 +104,9 @@
       end
```

```
       begin
+        logger.info "Attempting to #{operation} #{params[:hostname]} in FreeIPA"
+        logger.debug "#{options.inspect}"
+        set_auth()
         result = @ipa.call(operation, [params[:hostname]], options)
       rescue => e
         if e.message =~ /no modifications/
@@ -114,11 +123,13 @@
       check_realm realm
       raise Proxy::Realm::NotFound, "Host #{hostname} not found in realm!" unless find hostname
       begin
+        set_auth()
         result = @ipa.call("host_del", [hostname], "updatedns" => Proxy::Realm::Plugin.settings.freeipa_remov
e_dns)
       rescue
         if Proxy::Realm::Plugin.settings.freeipa_remove_dns
           # If the host doesn't have a DNS record (e.g. deleting a system in Foreman before it's built)
           # the above call will fail.  Try again with updatedns => false
+          set_auth()
           result = @ipa.call("host_del", [hostname], "updatedns" => false)
         else
           raise
```

#7 - 11/30/2015 12:41 PM - Anonymous

- Status changed from New to Assigned

- Assignee set to Anonymous

#8 - 12/09/2015 10:24 AM - The Foreman Bot

- Status changed from Assigned to Ready For Testing

- Pull request https://github.com/theforeman/smart-proxy/pull/353 added

#9 - 12/09/2015 10:25 AM - Anonymous

There is a fix available in the PR at https://github.com/theforeman/smart-proxy/pull/353. Please give it a try if you have a chance...

#10 - 12/09/2015 03:28 PM - Michael Eklund

Confirmed to work for me.

#11 - 12/10/2015 05:56 AM - Davy Stoffel

Confirmed to work for us too.

#12 - 12/22/2015 10:56 AM - Dominic Cleal

- translation missing: en.field_release set to 104

Thanks very much for the testing Michael and Davy! I've merged it on that basis.

#13 - 12/22/2015 11:02 AM - Anonymous

- Status changed from Ready For Testing to Closed

- % Done changed from 0 to 100

Applied in changeset 07a6dae64560582b423abe921f5c6ff57946a5bb.

#14 - 12/22/2015 12:51 PM - Donny Davis

Fix is also working for me on FreeIPA 4.2.3 with Foreman 1.9.3

#15 - 02/11/2016 01:33 PM - Stephen Benjamin

- Bugzilla link set to 1282539

#16 - 02/11/2016 01:33 PM - Stephen Benjamin

- Bugzilla link changed from 1282539 to 1305402