

Tutorial on txThings (CoAP Libraries)

Internet of Things (2IMN15) 2016-2017, Eindhoven University of Technology

By Leila F. Rahman

In this tutorial, we will describe how to run a CoAP server on Raspberry Pi and run the CoAP client on a Windows PC/Laptop. The CoAP server and clients are implemented using txThings, which is a Python implementation of CoAP. This tutorial also describes Copper, a GUI CoAP client tool for interacting with a CoAP server.

1 ABOUT TXTHINGS

txThings - CoAP library for Twisted framework

txThings is a Python implementation of Constrained Application Protocol (CoAP):

<http://tools.ietf.org/html/rfc7252>

txThings is based on Twisted - asynchronous I/O framework and networking engine written in Python.

<http://twistedmatrix.com/>

txThings uses MIT License (like Twisted itself).

<http://opensource.org/licenses/mit-license.php>

Copyright (c) 2012 Maciej Wasilak

<http://sixpinetrees.blogspot.com/>

txThings has the following features:

- support for draft-ietf-core-coap-13 - including automatic piggyback/separate response handling. No caching support.
- support for draft-ietf-core-block-12 (no support for server initiative though - waiting for the resolution)
- limited support for RFC6690 (Core Link Format) - server only.

Other nice things:

- txThings works nicely on RaspberryPi
- txThings is compatible with [Kivy](#) - brilliant new Python GUI library
- txThings is fully asynchronous (thanks to twisted framework)

2 INSTALLING TXTHINGS ON RASPBERRY PI

txThings is posted on [Github](#). txThings example codes are available on the Raspberry Pi on [reference/TxThings/examples](#)

Three examples are available:

- server.py - CoAP server that starts on localhost, port 5683 and hosts several resources
- client_GET.py - example client which performs GET request to localhost, port 5683
- client_PUT.py - example client which performs PUT request to localhost, port 5683

Client_GET and client_PUT both use port 61616 - to use them simultaneously change port number in one of the clients. Server will send blockwise responses for default settings. To use txThings you need Python 2.7 with Twisted installed (I suggest using the latest Twisted version, but older releases also work - tested with 11.1).

To install Twisted and txThings on your Raspberry Pi shell, run the following commands:

1. `sudo pip install twisted==15.1.0`
2. `sudo pip install txthings`

3 INSTALLING TXTHINGS IN CYGWIN ON WINDOWS

1. Run Cygwin Setup
2. Install python-setuptools (see **Figure 1**)

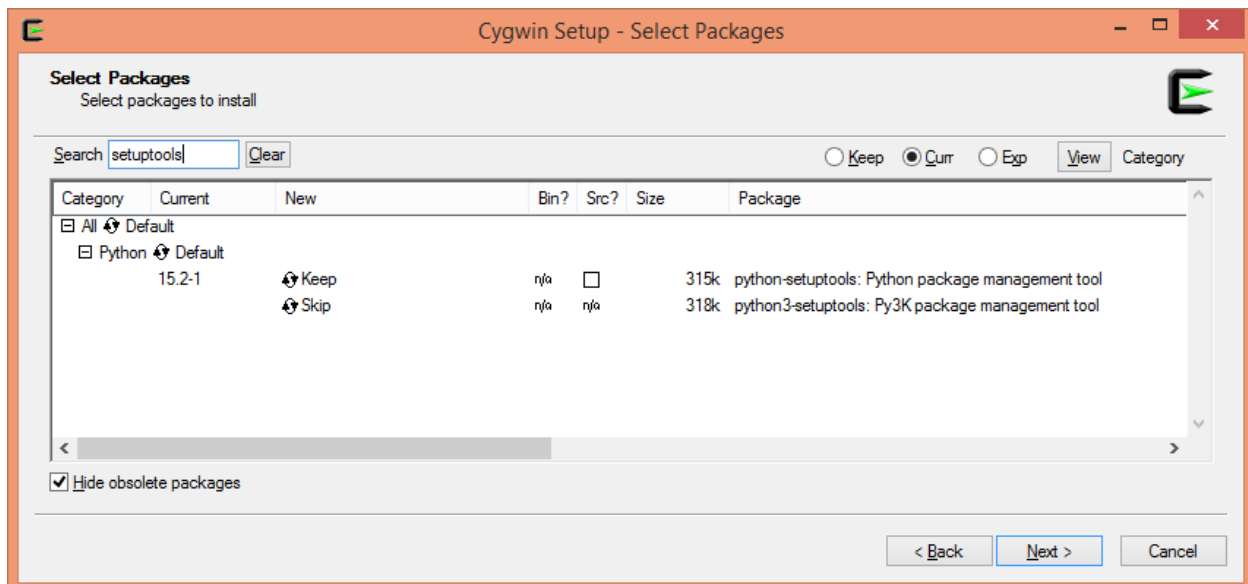


Figure 1. Install setuptools on Cygwin

TUTORIAL ON TXTHINGS

3. Install wget (see **Figure 2**)

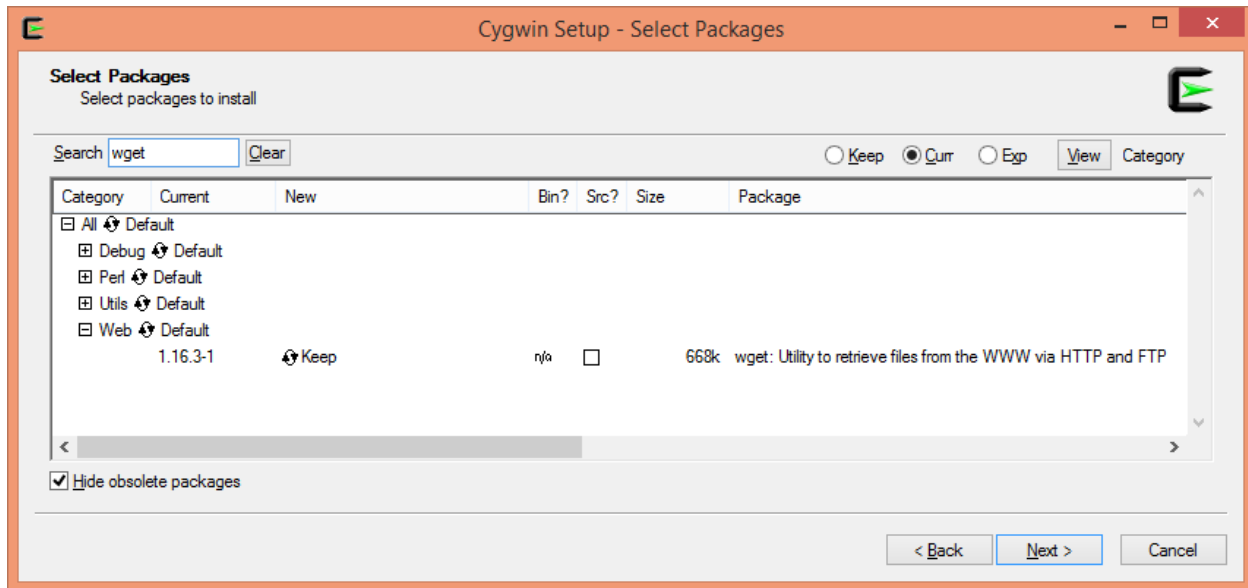
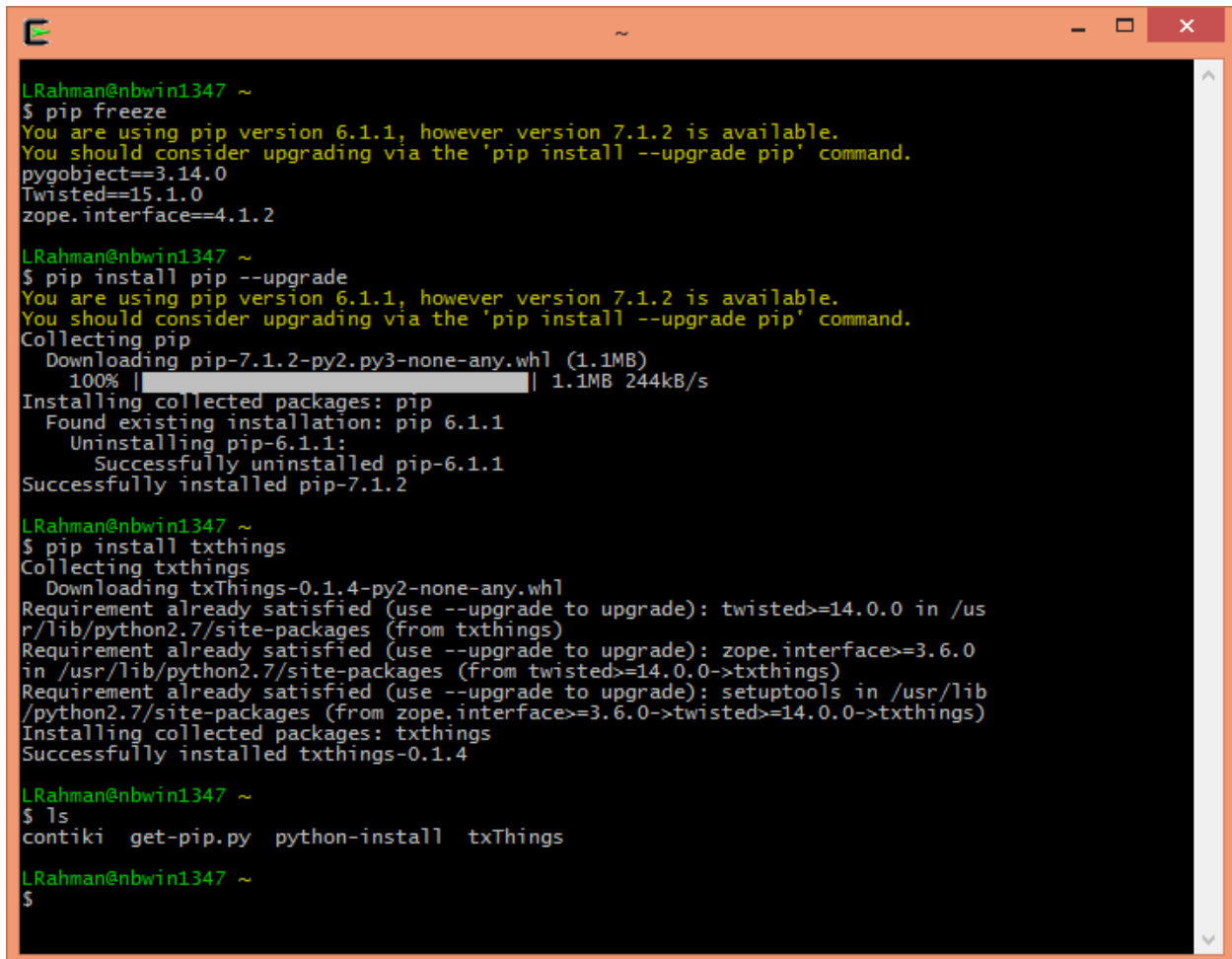


Figure 2. Install wget on Cygwin

4. Run Cygwin
5. Run the following commands (see **Figure 3**):
 - a. `wget https://bootstrap.pypa.io/get-pip.py` (to download get-pip.py)
 - b. `python get-pip.py`
 - c. `pip freeze` (to see which Python tools are installed)
 - d. `pip install pip --upgrade`
 - e. `pip install twisted==15.1.0`
 - f. `pip install txThings`

TUTORIAL ON TXTHINGS

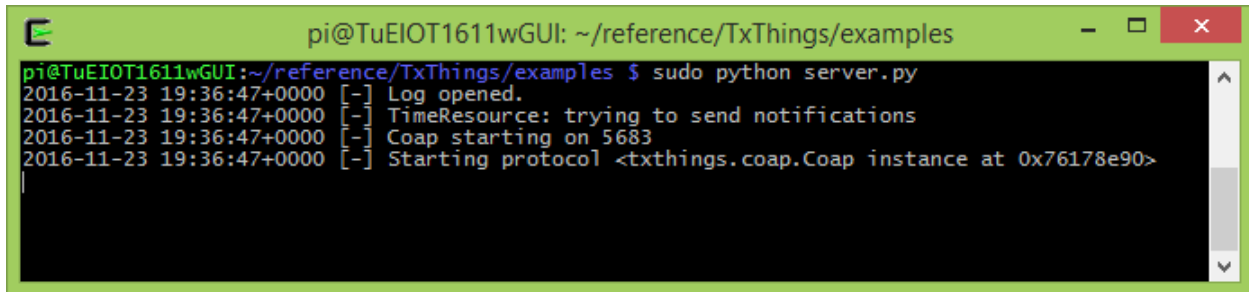
A terminal window with an orange title bar and standard window controls. The terminal text shows the user 'LRahman@nbwin1347' running several commands. First, 'pip freeze' shows installed packages. Then, 'pip install pip --upgrade' upgrades pip from 6.1.1 to 7.1.2, with a progress bar for the download. Finally, 'pip install txthings' installs txThings-0.1.4, showing that its dependencies (twisted, zope.interface, setuptools) are already satisfied. The session ends with 'ls' showing files in the current directory.

```
LRahman@nbwin1347 ~  
$ pip freeze  
You are using pip version 6.1.1, however version 7.1.2 is available.  
You should consider upgrading via the 'pip install --upgrade pip' command.  
pygobject==3.14.0  
Twisted==15.1.0  
zope.interface==4.1.2  
  
LRahman@nbwin1347 ~  
$ pip install pip --upgrade  
You are using pip version 6.1.1, however version 7.1.2 is available.  
You should consider upgrading via the 'pip install --upgrade pip' command.  
Collecting pip  
  Downloading pip-7.1.2-py2.py3-none-any.whl (1.1MB)  
    100% |#####| 1.1MB 244kB/s  
Installing collected packages: pip  
  Found existing installation: pip 6.1.1  
    Uninstalling pip-6.1.1:  
      Successfully uninstalled pip-6.1.1  
Successfully installed pip-7.1.2  
  
LRahman@nbwin1347 ~  
$ pip install txthings  
Collecting txthings  
  Downloading txThings-0.1.4-py2-none-any.whl  
Requirement already satisfied (use --upgrade to upgrade): twisted>=14.0.0 in /usr  
r/lib/python2.7/site-packages (from txthings)  
Requirement already satisfied (use --upgrade to upgrade): zope.interface>=3.6.0  
in /usr/lib/python2.7/site-packages (from twisted>=14.0.0->txthings)  
Requirement already satisfied (use --upgrade to upgrade): setuptools in /usr/lib  
/python2.7/site-packages (from zope.interface>=3.6.0->twisted>=14.0.0->txthings)  
Installing collected packages: txthings  
Successfully installed txthings-0.1.4  
  
LRahman@nbwin1347 ~  
$ ls  
contiki  get-pip.py  python-install  txThings  
  
LRahman@nbwin1347 ~  
$
```

Figure 3. Installing Twisted and txThings

4 RUN COAP SERVER ON RASPBERRY-PI

1. From the Raspberry's console or using *ssh* to the Raspberry-Pi, go to the examples directory at : *reference/txThings/examples*
2. Run the CoAP server : *python2 server.py* or *sudo python server.py*
(See **Figure 4**)

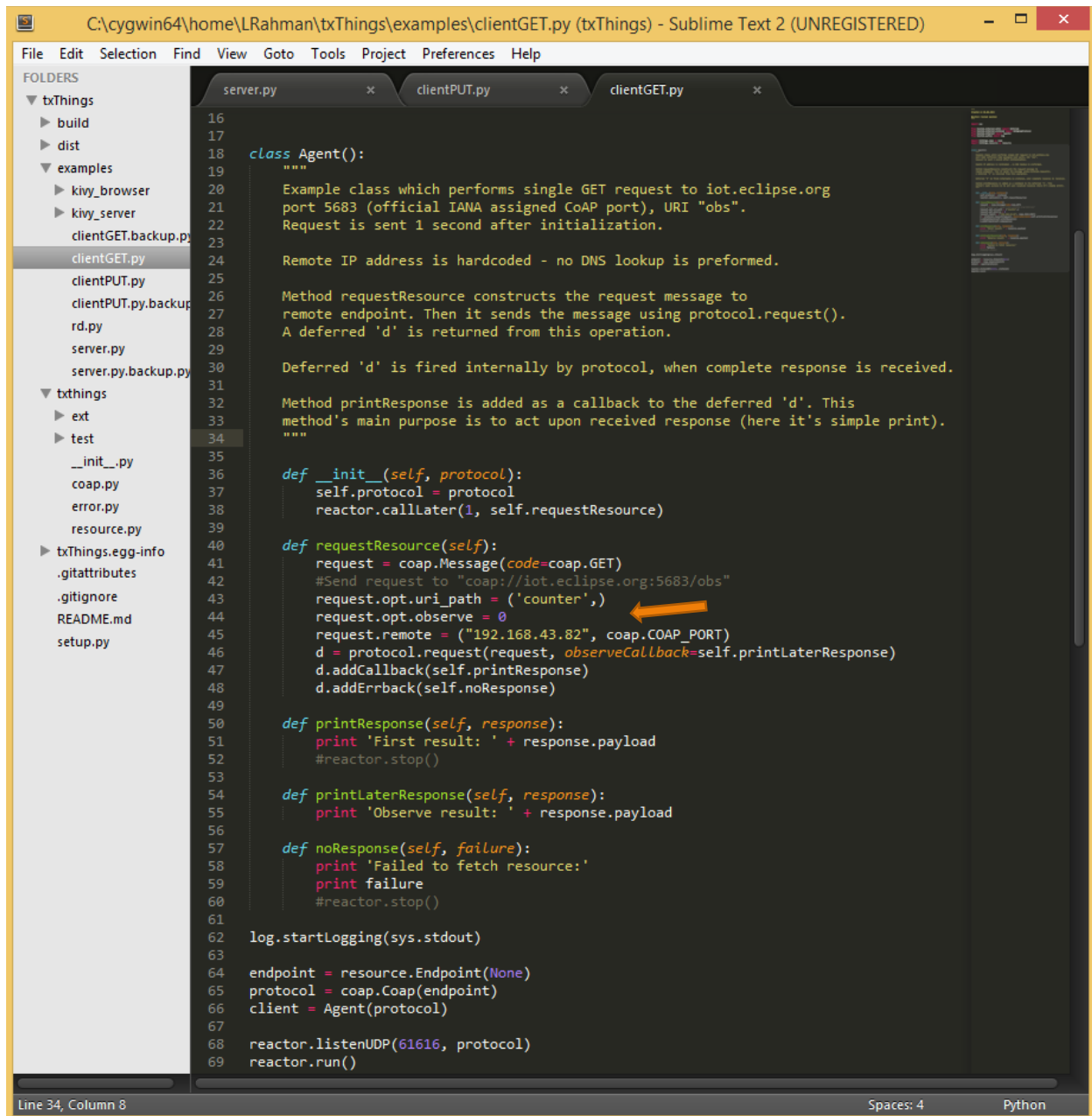
A terminal window with a green title bar. The title bar text is 'pi@TuEIoT1611wGUI: ~/reference/TxThings/examples'. The terminal content shows the command 'sudo python server.py' being executed. The output consists of four lines of log messages, each preceded by a timestamp '2016-11-23 19:36:47+0000' and a status indicator '[-]'. The messages are: 'Log opened.', 'TimeResource: trying to send notifications', 'Coap starting on 5683', and 'Starting protocol <txthings.coap.Coap instance at 0x76178e90>'.

```
pi@TuEIoT1611wGUI: ~/reference/TxThings/examples $ sudo python server.py
2016-11-23 19:36:47+0000 [-] Log opened.
2016-11-23 19:36:47+0000 [-] TimeResource: trying to send notifications
2016-11-23 19:36:47+0000 [-] Coap starting on 5683
2016-11-23 19:36:47+0000 [-] Starting protocol <txthings.coap.Coap instance at 0x76178e90>
```

Figure 4. Running the CoAP Server on Raspberry-Pi using SSH

5 RUN COAP CLIENT ON PC

1. If using Windows, open Cygwin
2. Go to the examples directory at : *txThings/examples*
3. Change the resource name (in this example from 'obs' to 'counter') and remote IP address in the source code of *clientGET.py* or *clientPUT.py* (see **Figure 5**)

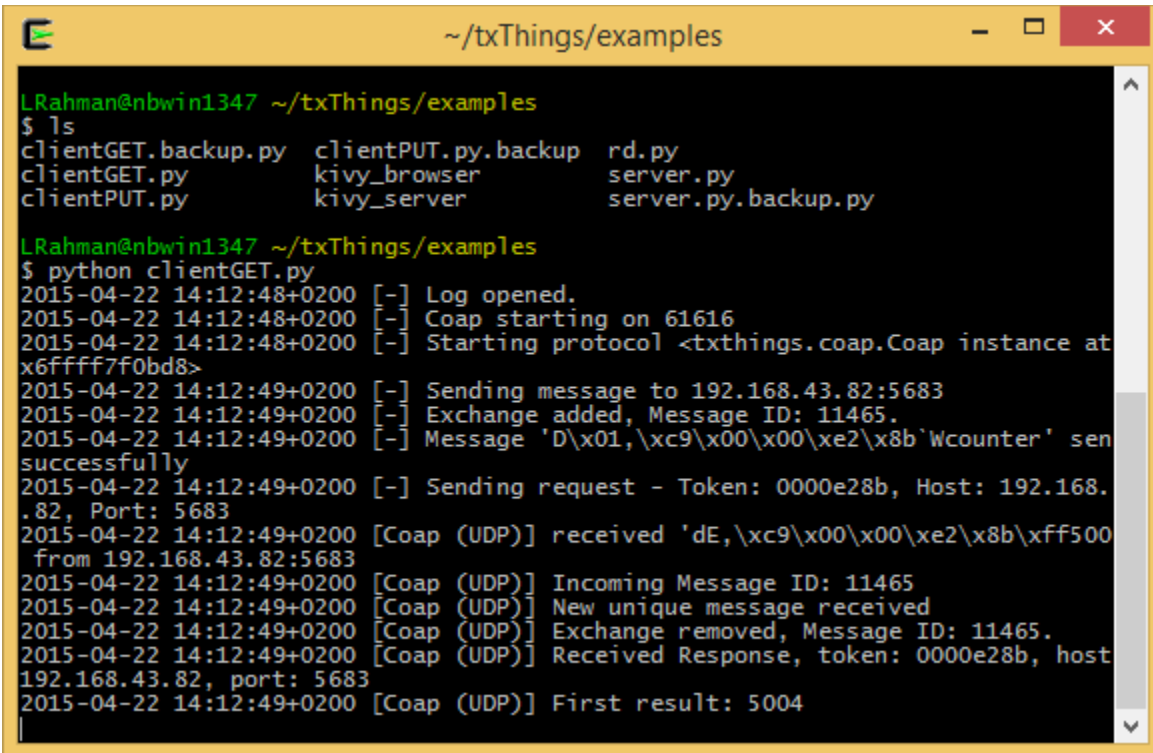


```
16
17
18 class Agent():
19     """
20     Example class which performs single GET request to iot.eclipse.org
21     port 5683 (official IANA assigned CoAP port), URI "obs".
22     Request is sent 1 second after initialization.
23
24     Remote IP address is hardcoded - no DNS lookup is preformed.
25
26     Method requestResource constructs the request message to
27     remote endpoint. Then it sends the message using protocol.request().
28     A deferred 'd' is returned from this operation.
29
30     Deferred 'd' is fired internally by protocol, when complete response is received.
31
32     Method printResponse is added as a callback to the deferred 'd'. This
33     method's main purpose is to act upon received response (here it's simple print).
34     """
35
36 def __init__(self, protocol):
37     self.protocol = protocol
38     reactor.callLater(1, self.requestResource)
39
40 def requestResource(self):
41     request = coap.Message(code=coap.GET)
42     #Send request to "coap://iot.eclipse.org:5683/obs"
43     request.opt.uri_path = ('counter',)
44     request.opt.observe = 0
45     request.remote = ("192.168.43.82", coap.COAP_PORT)
46     d = protocol.request(request, observeCallback=self.printLaterResponse)
47     d.addCallback(self.printResponse)
48     d.addErrback(self.noResponse)
49
50 def printResponse(self, response):
51     print 'First result: ' + response.payload
52     #reactor.stop()
53
54 def printLaterResponse(self, response):
55     print 'Observe result: ' + response.payload
56
57 def noResponse(self, failure):
58     print 'Failed to fetch resource:'
59     print failure
60     #reactor.stop()
61
62 log.startLogging(sys.stdout)
63
64 endpoint = resource.Endpoint(None)
65 protocol = coap.Coap(endpoint)
66 client = Agent(protocol)
67
68 reactor.listenUDP(61616, protocol)
69 reactor.run()
```

Figure 5. In the source code of *clientGET.py*, change the resource name to 'counter' and remote IP Address to the IP Address of the Raspberry Pi running *server.py*

TUTORIAL ON TXTHINGS

4. Run the CoAP clients (see **Figure 6**):
 - a. For GET operation: `python clientGET.py`
 - b. For PUT operation: `python clientPUT.py`



```
~ /txThings/examples
LRahman@nbwin1347 ~/txThings/examples
$ ls
clientGET.backup.py  clientPUT.py.backup  rd.py
clientGET.py         kivy_browser         server.py
clientPUT.py         kivy_server          server.py.backup.py

LRahman@nbwin1347 ~/txThings/examples
$ python clientGET.py
2015-04-22 14:12:48+0200 [-] Log opened.
2015-04-22 14:12:48+0200 [-] Coap starting on 61616
2015-04-22 14:12:48+0200 [-] Starting protocol <txthings.coap.Coap instance at
x6ffff7f0bd8>
2015-04-22 14:12:49+0200 [-] Sending message to 192.168.43.82:5683
2015-04-22 14:12:49+0200 [-] Exchange added, Message ID: 11465.
2015-04-22 14:12:49+0200 [-] Message 'D\x01,\xc9\x00\x00\xe2\x8b`wcounter' sen
successfully
2015-04-22 14:12:49+0200 [-] Sending request - Token: 0000e28b, Host: 192.168.
.82, Port: 5683
2015-04-22 14:12:49+0200 [Coap (UDP)] received 'dE,\xc9\x00\x00\xe2\x8b\xff500
from 192.168.43.82:5683
2015-04-22 14:12:49+0200 [Coap (UDP)] Incoming Message ID: 11465
2015-04-22 14:12:49+0200 [Coap (UDP)] New unique message received
2015-04-22 14:12:49+0200 [Coap (UDP)] Exchange removed, Message ID: 11465.
2015-04-22 14:12:49+0200 [Coap (UDP)] Received Response, token: 0000e28b, host
192.168.43.82, port: 5683
2015-04-22 14:12:49+0200 [Coap (UDP)] First result: 5004
```

Figure 6. Run *clientGET.py* on a Windows PC using Cygwin

6 RUN COPPER ON FIREFOX

The Copper (Cu) CoAP user-agent is an add-on for the Firefox Web browser. It allows browsing, bookmarking, and direct interaction with CoAP resources. Install Copper on Firefox from <https://addons.mozilla.org/en-US/firefox/addon/copper-270430/>

Once installed, simply enter a CoAP URI into the address bar of Firefox Web browser (See **Figure 7**, **Figure 8** and **Figure 9**)

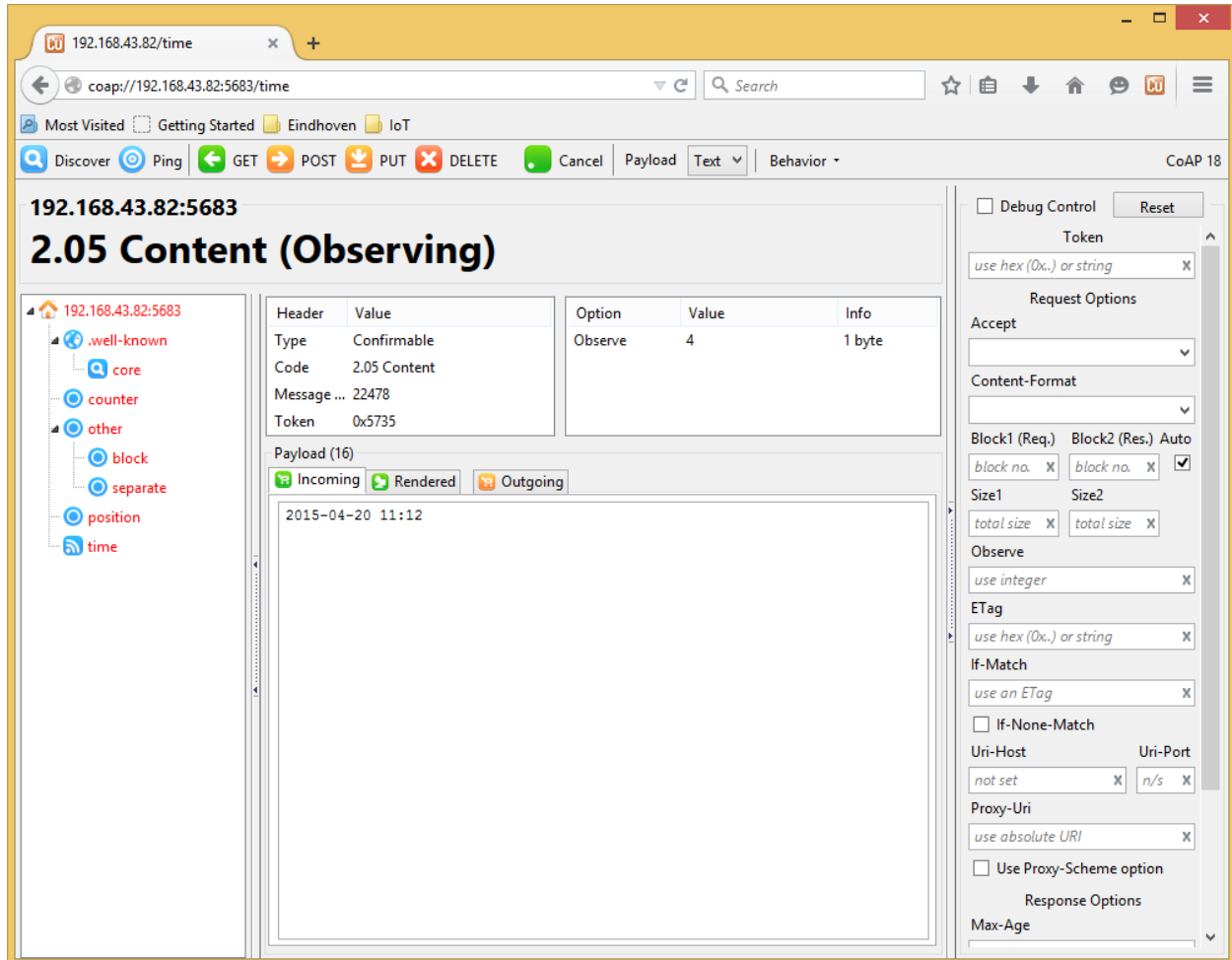


Figure 7. Running the OBSERVE operation on the resource “time” using Copper

TUTORIAL ON TXTHINGS

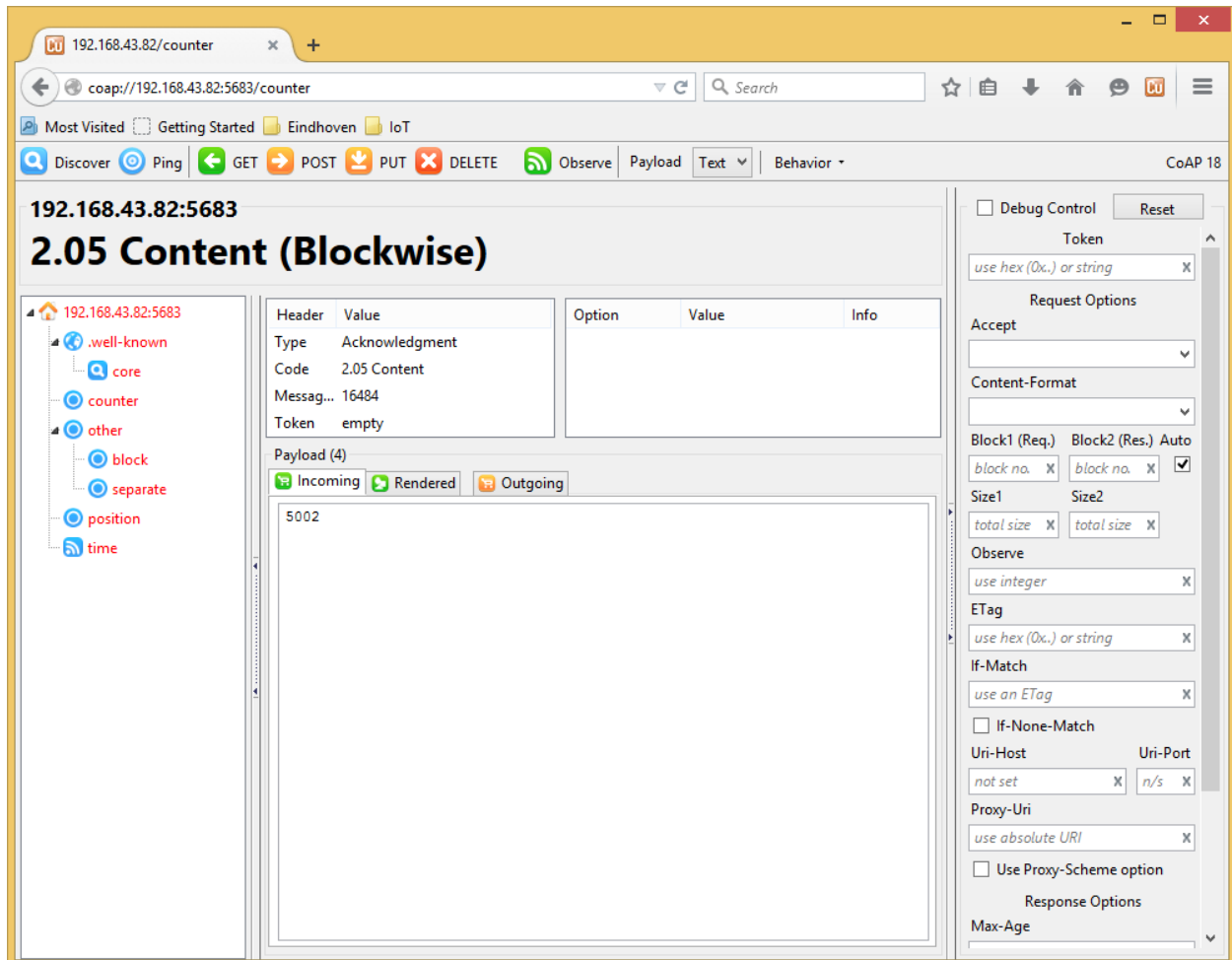


Figure 8. Running the GET operation on the resource “counter” using Copper

TUTORIAL ON TXTHINGS

The screenshot shows the Copper REST client interface. The main window displays the response for a PUT operation on the resource "block". The response is "2.04 Changed (Blockwise) (Download finished)". The headers table shows:

Header	Value
Type	Acknowledgment
Code	2.04 Changed
Message-ID	35959
Token	empty

The options table shows:

Option	Value	Info
Block2	1 (64 B/block)	1 byte

The combined payload (122) is displayed as:

Mr. and Mrs. Dursley of number four, Privet Drive, were proud to say that they were perfectly normal, thank you very much.

A terminal window in the foreground shows the CoAP log messages for the PUT operation:

```
pi@raspberrypi2: ~/twisted/txThings/examples
2015-04-20 11:17:15+0200 [Coap (UDP)] Message ''D\x8c\x01\n\n\xffMr. and Mrs. Dursley of number four, Privet Drive, were proud to say that they were perfectly normal, thank you very much.' sent successfully
2015-04-20 11:17:15+0200 [Coap (UDP)] received '@\x03\x8c\x05other\x05block\x01\x12' from 192.168.43.209:59426
2015-04-20 11:17:15+0200 [Coap (UDP)] Incoming Message ID: 35959
2015-04-20 11:17:15+0200 [Coap (UDP)] New unique message received
2015-04-20 11:17:15+0200 [Coap (UDP)] Request pertains to earlier blockwise requests.
2015-04-20 11:17:15+0200 [Coap (UDP)] Request with Block2 option received, number = 1, more = 0, size_exp = 2.
2015-04-20 11:17:15+0200 [Coap (UDP)] Token:
2015-04-20 11:17:15+0200 [Coap (UDP)] Sending response, type = ACK (request type = CON)
2015-04-20 11:17:15+0200 [Coap (UDP)] Sending message to 192.168.43.209:59426
2015-04-20 11:17:15+0200 [Coap (UDP)] Message ''D\x8c\x01\n\n\xff say that they were perfectly normal, thank you very much.' sent successfully
```

Figure 9. Running the PUT operation on the resource "block" using Copper