Obscure Communication using Tor Relay and Hidden Service

Priyank Jani, Graduate Capstone Advisor: Dr. Sumita Mishra

Introduction

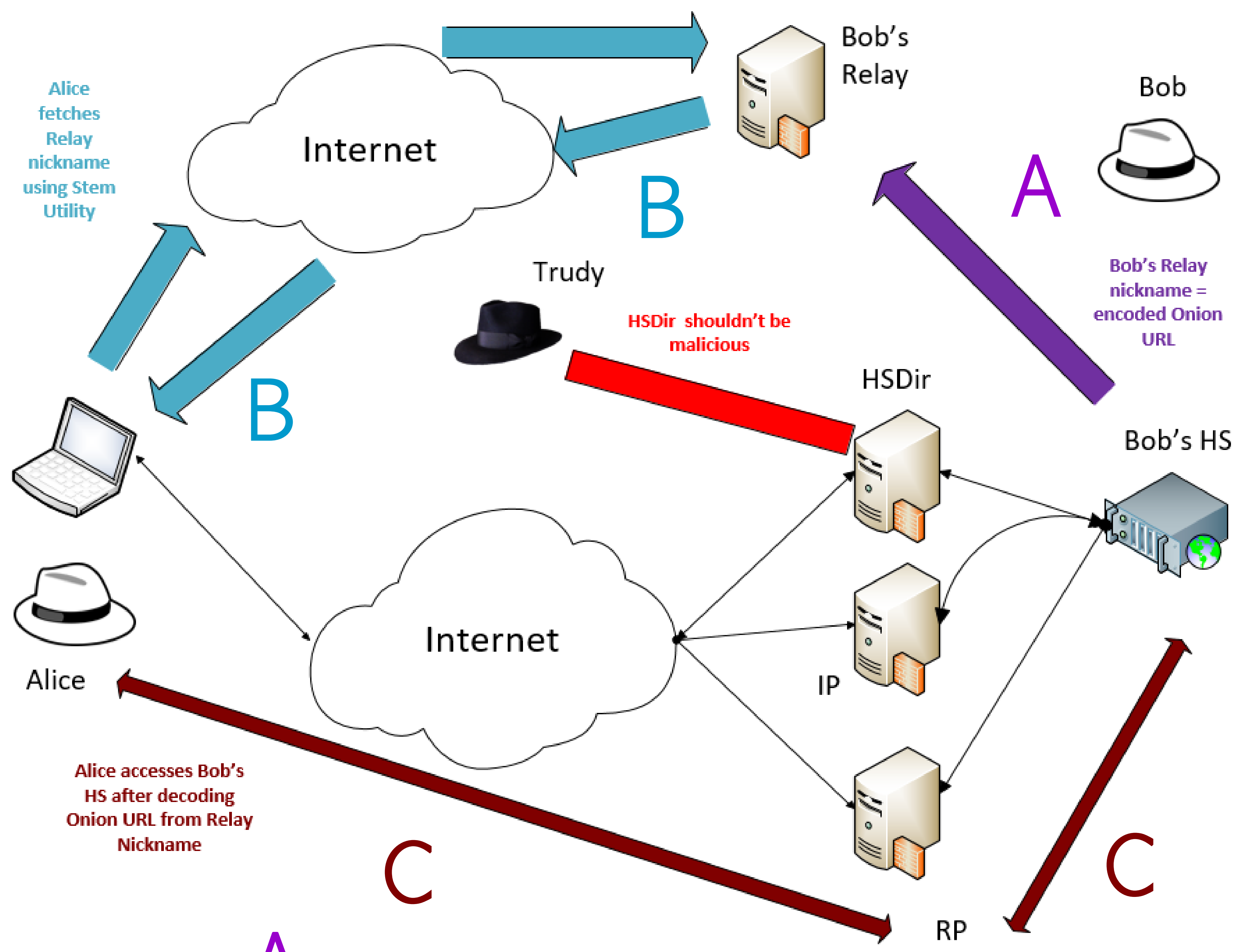
- Hidden Services in the Tor network provide responder anonymity, protecting the location and identity of service providers by hiding their IP address. Hidden Services are crucial as they are used by military, dissidents, whistleblowers among many others.
- From the Tor website "When the Internet was designed by DARPA, its primary purpose was to be able to facilitate distributed, robust communications in case of local strikes. However, some functions must be centralized, such as command and control sites. It's the nature of the Internet protocols to reveal the geographic location of any server that is reachable online. Tor's hidden services capacity allows military command and control to be physically secure from discovery and takedown."

Project Description

- To enable obscure communication between Alice and Bob, the implementation of the communication system was aimed to avoid the detection of a Hidden Service, the existence of which, has not been made public.
- This project aims to establish a communication mechanism where Onion URL's need not be advertised out-of-band, in fact, they need not be advertised at all.
- The 16-character onion URL is encoded and made part of the Tor Consensus by using it as a Relay nickname for a Tor Relay that is owned by Bob. Bob passes on the IP Address of his Tor Relay and Directory Port to Alice out-of-band. Alice and Bob have a pre-decided encoding/decoding scheme in place.
- A Malicious HSDir's intentions can be foiled by implementing Hidden Services in stealth mode or migrating to rendezvous specification version 3.

Conclusion

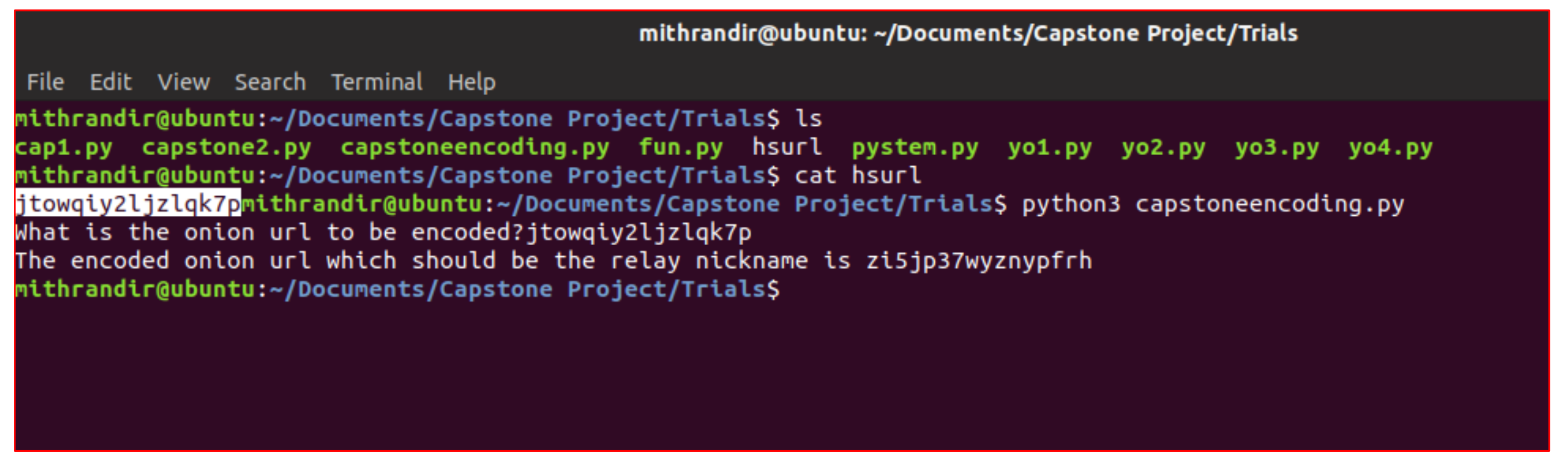
- This project uses the STEM utility which is an implementation of Tor Control specification in Python.
- A Tor Relay was actively deployed and made part of the Tor Consensus. The Hidden Service was also hosted on a cloud service provider.
- Implementing the Hidden Service in stealth mode and deploying a stronger encoding/decoding scheme will improve this communication system.


Bob's Relay
Bob
Alice fetches Relay nickname using Stem Utility
Internet
B
B
A
Bob's Relay nickname = encoded Onion URL
Trudy
HSDir shouldn't be malicious
HSDir
Bob's HS
Alice
Internet
IP
Alice accesses Bob's HS after decoding Onion URL from Relay Nickname
C
C
RP

A

```
encoding_dict = {
        'a':'g', 'b':'d', 'c':'c', 'd':'b',
        'e':'a', 'f':'q', 'g':'l', 'h':'o',
        'i':'3', 'j':'z', 'k':'f', 'l':'y',
        'm':'4', 'n':'u', 'o':'5', 'p':'h',
        'q':'p', 'r':'x', 's':'6', 't':'i',
        'u':'e', 'v':'s', 'w':'j', 'x':'2',
        'y':'7', 'z':'n', '2':'w', '3':'v',
        '4':'t', '5':'m', '6':'k', '7':'r'
}
```

Encoding scheme deployed by Bob

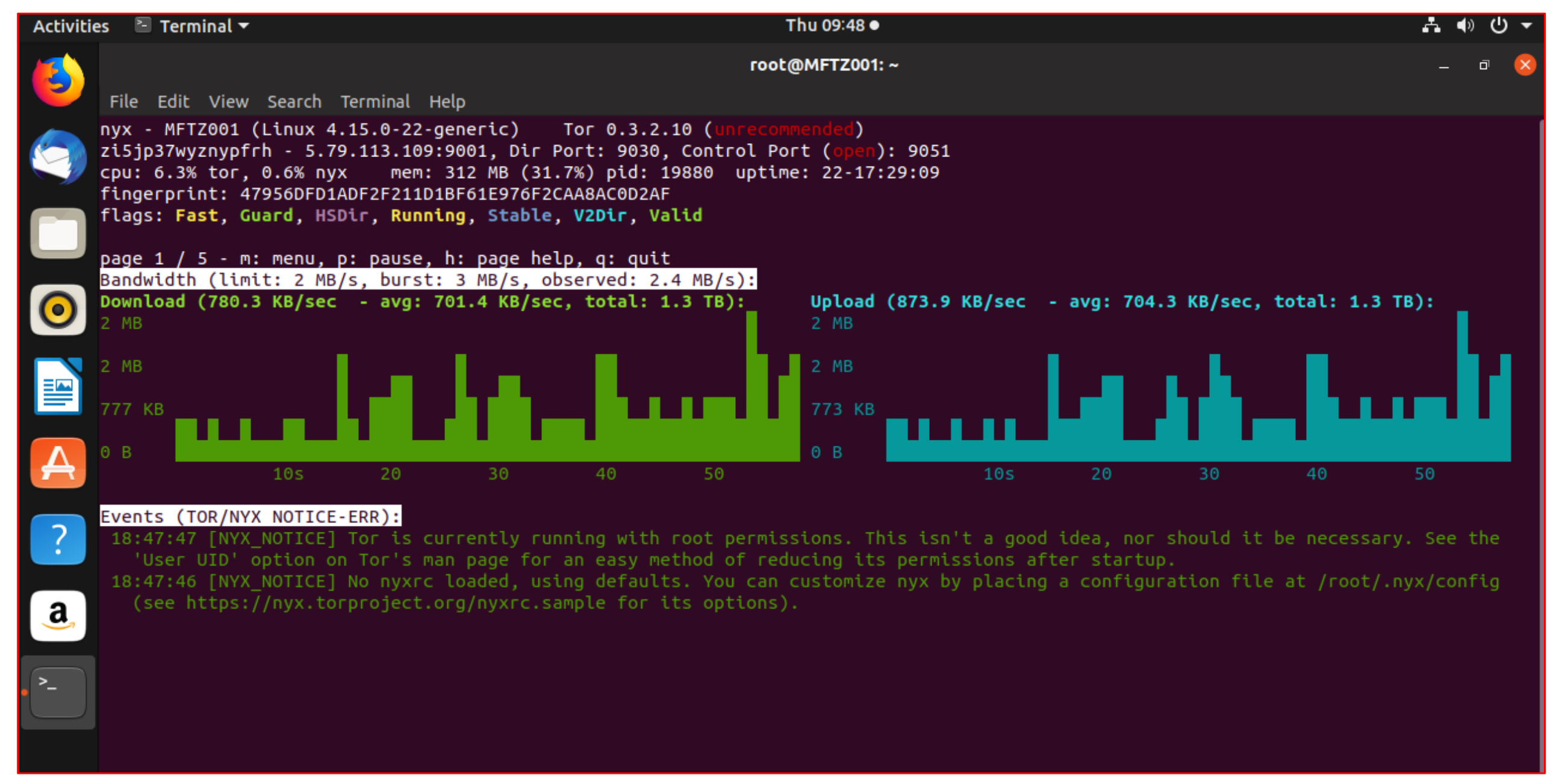


```
mithrandir@ubuntu: ~/Documents/Capstone Project/Trials
File  Edit  View  Search  Terminal  Help
mithrandir@ubuntu:~/Documents/Capstone Project/Trials$ ls
cap1.py  capstone2.py  capstoneencoding.py  fun.py  hsurl  pystem.py  yo1.py  yo2.py  yo3.py  yo4.py
mithrandir@ubuntu:~/Documents/Capstone Project/Trials$ cat hsurl
jtowqiy2ljzlqk7pmithrandir@ubuntu:~/Documents/Capstone Project/Trials$ python3 capstoneencoding.py
What is the onion url to be encoded?jtowqiy2ljzlqk7p
The encoded onion url which should be the relay nickname is zi5jp37wyznypfrh
mithrandir@ubuntu:~/Documents/Capstone Project/Trials$
```

Python script deployed at Bob's end to obtain encoded Relay nickname

```
def decode_str(string):
    res2 = ''
    for ele in string:
        for key,value in encoding_dict.items()
            if (value == ele):
                res2 = res2 + key
    return res2
```

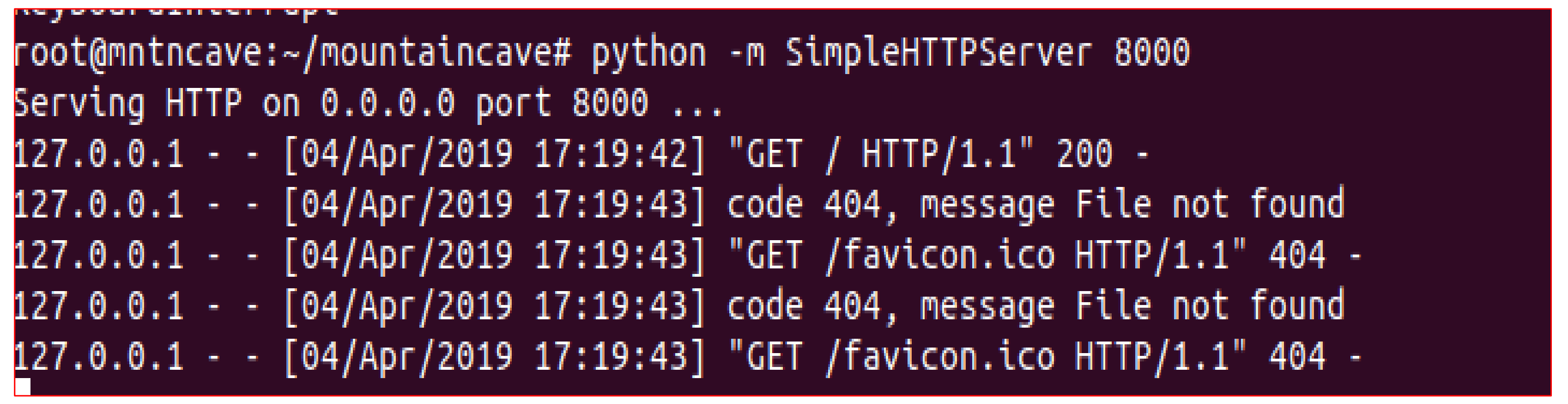
A section of the Python script deployed at Alice's end to obtain decoded Onion URL



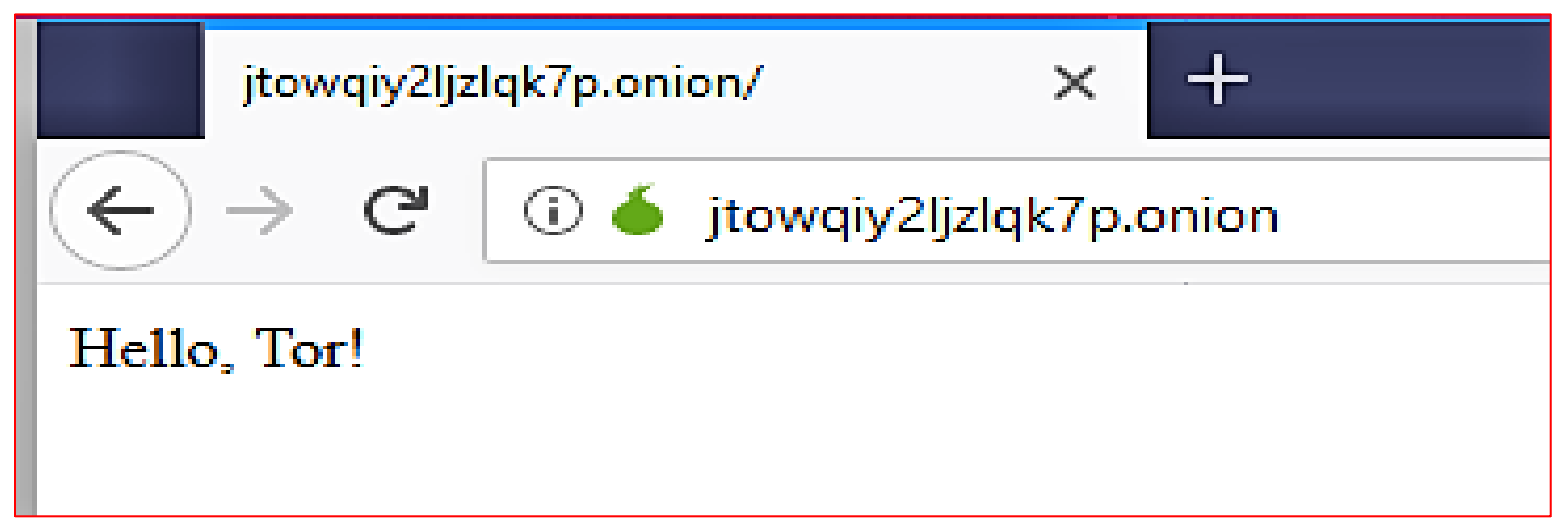
```
Activities   Terminal ▾                    Thu 09:48 ●
root@MFTZ001: ~
File  Edit  View  Search  Terminal  Help
nyx - MFTZ001 (Linux 4.15.0-22-generic)    Tor 0.3.2.10 (unrecommended)
zi5jp37wyznypfrh - 5.79.113.109:9001, Dir Port: 9030, Control Port (open): 9051
cpu: 6.3% tor, 0.6% nyx    mem: 312 MB (31.7%) pid: 19880  uptime: 22-17:29:09
fingerprint: 47956DFD1ADF2F211D1BF61E976F2CAA8AC0D2AF
flags: Fast, Guard, HSDir, Running, Stable, V2Dir, Valid

page 1 / 5 - m: menu, p: pause, h: page help, q: quit
Bandwidth (limit: 2 MB/s, burst: 3 MB/s, observed: 2.4 MB/s):
Download (780.3 KB/sec  - avg: 701.4 KB/sec, total: 1.3 TB):     Upload (873.9 KB/sec  - avg: 704.3 KB/sec, total: 1.3 TB):
2 MB                                                              2 MB

2 MB                                                              2 MB

777 KB                                                            773 KB

0 B                                                               0 B
           10s     20     30     40     50                                   10s     20     30     40     50

Events (TOR/NYX NOTICE-ERR):
 18:47:47 [NYX_NOTICE] Tor is currently running with root permissions. This isn't a good idea, nor should it be necessary. See the
   'User UID' option on Tor's man page for an easy method of reducing its permissions after startup.
 18:47:46 [NYX_NOTICE] No nyxrc loaded, using defaults. You can customize nyx by placing a configuration file at /root/.nyx/config
   (see https://nyx.torproject.org/nyxrc.sample for its options).
```

Details of the torrc file for Bob's Relay observed using the Nyx utility

C



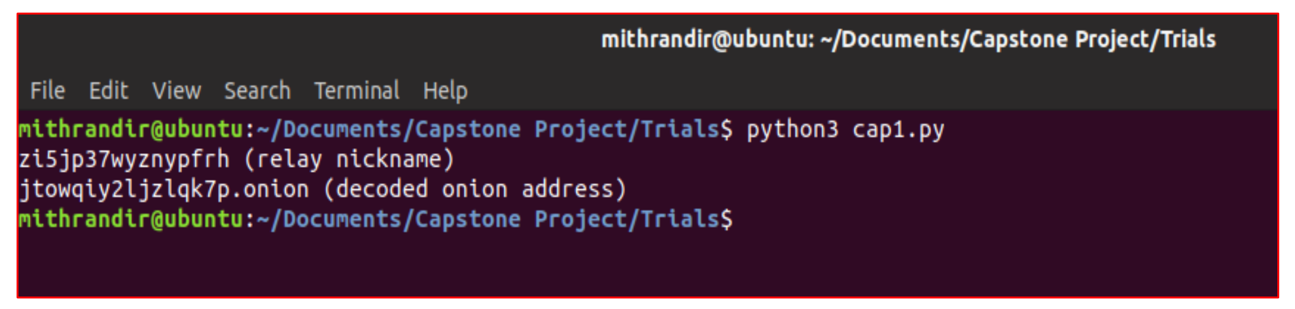
```
root@mntncave:~/mountaincave# python -m SimpleHTTPServer 8000
Serving HTTP on 0.0.0.0 port 8000 ...
127.0.0.1 - - [04/Apr/2019 17:19:42] "GET / HTTP/1.1" 200 -
127.0.0.1 - - [04/Apr/2019 17:19:43] code 404, message File not found
127.0.0.1 - - [04/Apr/2019 17:19:43] "GET /favicon.ico HTTP/1.1" 404 -
127.0.0.1 - - [04/Apr/2019 17:19:43] code 404, message File not found
127.0.0.1 - - [04/Apr/2019 17:19:43] "GET /favicon.ico HTTP/1.1" 404 -
```



Hello, Tor!

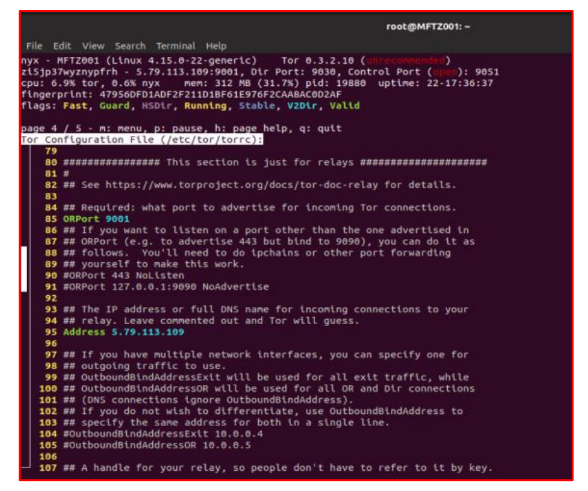
Alice accessing the Hidden Service using the decoded Onion URL and corresponding logs on Bob's Hidden Service

B

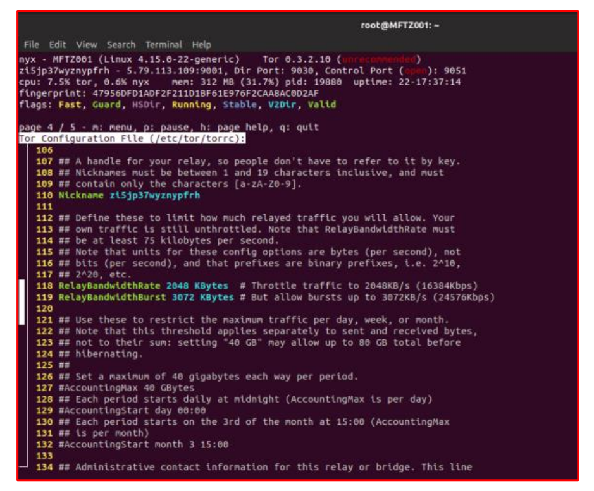


```
mithrandir@ubuntu: ~/Documents/Capstone Project/Trials
File  Edit  View  Search  Terminal  Help
mithrandir@ubuntu:~/Documents/Capstone Project/Trials$ python3 cap1.py
zi5jp37wyznypfrh (relay nickname)
jtowqiy2ljzlqk7p.onion (decoded onion address)
mithrandir@ubuntu:~/Documents/Capstone Project/Trials$
```

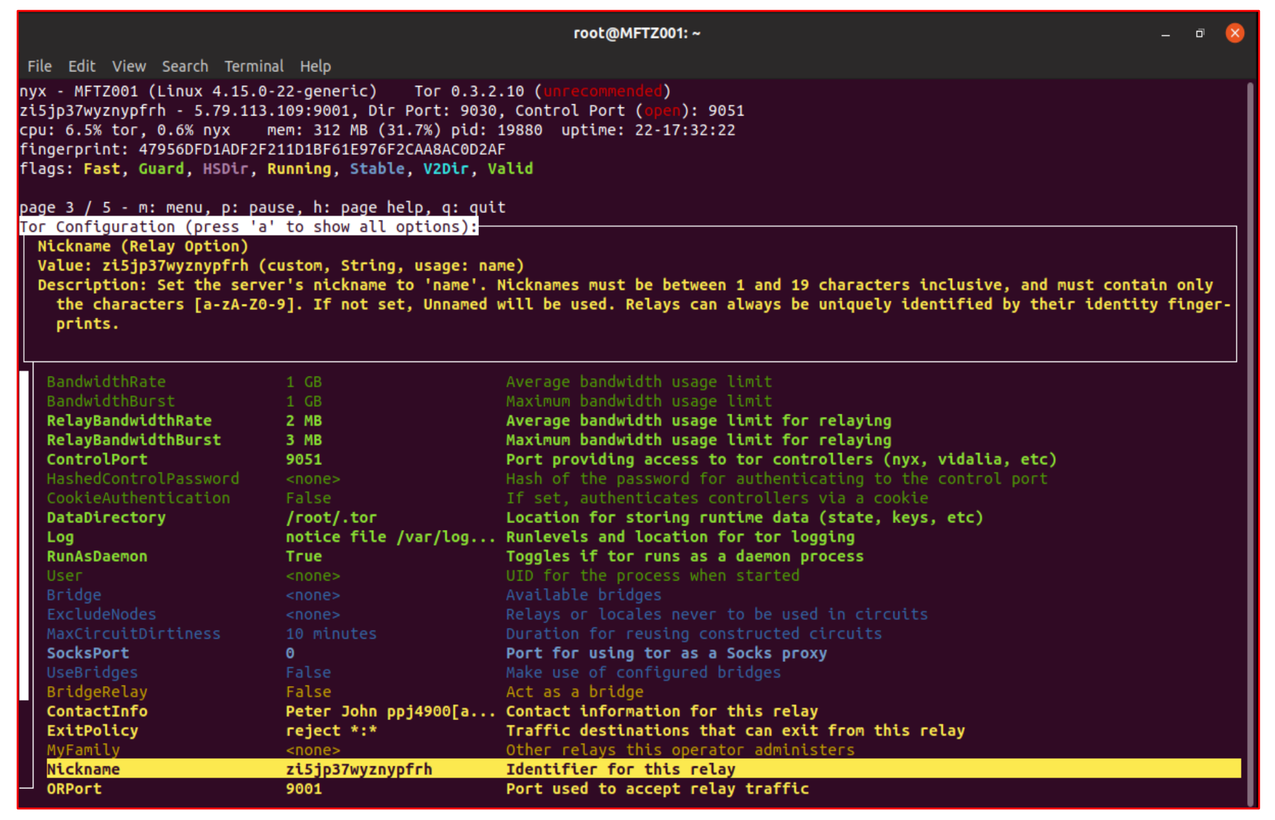
Result of running the decoding/STEM Python script at Alice's end



```
root@MFTZ001: ~
File  Edit  View  Search  Terminal  Help
nyx - MFTZ001 (Linux 4.15.0-22-generic)    Tor 0.3.2.10 (unrecommended)
zi5jp37wyznypfrh - 5.79.113.109:9001, Dir Port: 9030, Control Port (open): 9051
cpu: 6.9% tor, 0.6% nyx    mem: 312 MB (31.7%) pid: 19880  uptime: 22-17:36:37
fingerprint: 47956DFD1ADF2F211D1BF61E976F2CAA8AC0D2AF
flags: Fast, Guard, HSDir, Running, Stable, V2Dir, Valid

page 4 / 5 - m: menu, p: pause, h: page help, q: quit
Tor Configuration File (/etc/tor/torrc):

 ################ This section is just for relays #####################
 #
 ## See https://www.torproject.org/docs/tor-doc-relay for details.

 ## Required: what port to advertise for incoming Tor connections.
 ORPort 9001
 ## If you want to listen on a port other than the one advertised in
 ## ORPort (e.g. to advertise 443 but bind to 9090), you can do it as
 ## follows.  You'll need to do ipchains or other port forwarding
 ## yourself to make this work.
 #ORPort 443 NoListen
 #ORPort 127.0.0.1:9090 NoAdvertise

 ## The IP address or full DNS name for incoming connections to your
 ## relay. Leave commented out and Tor will guess.
 Address 5.79.113.109

 ## If you have multiple network interfaces, you can specify one for
 ## outgoing traffic to use.
 ## OutboundBindAddressExit will be used for all exit traffic, while
 ## OutboundBindAddressOR will be used for all OR and Dir connections
 ## (DNS connections ignore OutboundBindAddress).
 ## If you do not wish to differentiate, use OutboundBindAddress to
 ## specify the same address for both in a single line.
 #OutboundBindAddressExit 10.0.0.4
 #OutboundBindAddressOR 10.0.0.5

 ## A handle for your relay, so people don't have to refer to it by key.
```



```
root@MFTZ001: ~
File  Edit  View  Search  Terminal  Help
nyx - MFTZ001 (Linux 4.15.0-22-generic)    Tor 0.3.2.10 (unrecommended)
zi5jp37wyznypfrh - 5.79.113.109:9001, Dir Port: 9030, Control Port (open): 9051
cpu: 7.5% tor, 0.6% nyx    mem: 312 MB (31.7%) pid: 19880  uptime: 22-17:37:14
fingerprint: 47956DFD1ADF2F211D1BF61E976F2CAA8AC0D2AF
flags: Fast, Guard, HSDir, Running, Stable, V2Dir, Valid

page 4 / 5 - m: menu, p: pause, h: page help, q: quit
Tor Configuration File (/etc/tor/torrc):

 ## A handle for your relay, so people don't have to refer to it by key.
 ## Nicknames must be between 1 and 19 characters inclusive, and must
 ## contain only the characters [a-zA-Z0-9].
 Nickname zi5jp37wyznypfrh

 ## Define these to limit how much relayed traffic you will allow. Your
 ## own traffic is still unthrottled. Note that RelayBandwidthRate must
 ## be at least 75 kilobytes per second.
 ## Note that units for these config options are bytes (per second), not
 ## bits (per second), and that prefixes are binary prefixes, i.e. 2^10,
 ## 2^20, etc.
 RelayBandwidthRate 2048 KBytes  # Throttle traffic to 2048KB/s (16384Kbps)
 RelayBandwidthBurst 3072 KBytes # But allow bursts up to 3072KB/s (24576Kbps)

 ## Use these to restrict the maximum traffic per day, week, or month.
 ## Note that this threshold applies separately to sent and received bytes,
 ## not to their sum: setting "40 GB" may allow up to 80 GB total before
 ## hibernating.
 ##
 ## Set a maximum of 40 gigabytes each way per period.
 #AccountingMax 40 GBytes
 ## Each period starts daily at midnight (AccountingMax is per day)
 #AccountingStart day 00:00
 ## Each period starts on the 3rd of the month at 15:00 (AccountingMax
 ## is per month)
 #AccountingStart month 3 15:00

 ## Administrative contact information for this relay or bridge. This line
```



```
root@MFTZ001: ~
File  Edit  View  Search  Terminal  Help
nyx - MFTZ001 (Linux 4.15.0-22-generic)    Tor 0.3.2.10 (unrecommended)
zi5jp37wyznypfrh - 5.79.113.109:9001, Dir Port: 9030, Control Port (open): 9051
cpu: 6.5% tor, 0.6% nyx    mem: 312 MB (31.7%) pid: 19880  uptime: 22-17:32:22
fingerprint: 47956DFD1ADF2F211D1BF61E976F2CAA8AC0D2AF
flags: Fast, Guard, HSDir, Running, Stable, V2Dir, Valid

page 3 / 5 - m: menu, p: pause, h: page help, q: quit
Tor Configuration (press 'a' to show all options):
  Nickname (Relay Option)
  Value: zi5jp37wyznypfrh (custom, String, usage: name)
  Description: Set the server's nickname to 'name'. Nicknames must be between 1 and 19 characters inclusive, and must contain only
    the characters [a-zA-Z0-9]. If not set, Unnamed will be used. Relays can always be uniquely identified by their identity finger-
    prints.

  BandwidthRate             1 GB                   Average bandwidth usage limit
  BandwidthBurst            1 GB                   Maximum bandwidth usage limit
  RelayBandwidthRate        2 MB                   Average bandwidth usage limit for relaying
  RelayBandwidthBurst       3 MB                   Maximum bandwidth usage limit for relaying
  ControlPort               9051                   Port providing access to tor controllers (nyx, vidalia, etc)
  HashedControlPassword     <none>                 Hash of the password for authenticating to the control port
  CookieAuthentication      False                  If set, authenticates controllers via a cookie
  DataDirectory             /root/.tor             Location for storing runtime data (state, keys, etc)
  Log                       notice file /var/log... Runlevels and location for tor logging
  RunAsDaemon               True                   Toggles if tor runs as a daemon process
  User                      <none>                 UID for the process when started
  Bridge                    <none>                 Available bridges
  ExcludeNodes              <none>                 Relays or locales never to be used in circuits
  MaxCircuitDirtiness       10 minutes             Duration for reusing constructed circuits
  SocksPort                 0                      Port for using tor as a Socks proxy
  UseBridges                False                  Make use of configured bridges
  BridgeRelay               False                  Act as a bridge
  ContactInfo               Peter John ppj4900[a... Contact information for this relay
  ExitPolicy                reject *:*             Traffic destinations that can exit from this relay
  MyFamily                  <none>                 Other relays this operator administers
  Nickname                  zi5jp37wyznypfrh       Identifier for this relay
  ORPort                    9001                   Port used to accept relay traffic
```

RIT | Golisano College of Computing and Information Sciences
Department of Computing Security