ZeroNet Browser Bundle

Amol Kulkarni[1] Abhishek Bagde[2] Bhagyesh Dhatrak[3] Akshay Tiwari[4] Saguna Ingle[5]
[1,2,3,4,5]Department of Computer Engineering
[1,2,3,4,5]Ramrao Adik Institute of Technology, Nerul, India

Abstract— The Internet is the global system of interconnected computer network that links billions of devices worldwide. It is a network of networks that consist of millions of private, public academic business and government network of local and global scope, linked by a broad array of electronics wireless and optical networking technologies. ZeroNet is a new emerging overlay network that could provide alternative to the existing centralized system. ZeroNet provides open free and uncensored service using bitcoin cryptography and BitTorrent network. In ZeroNet content is distributed directly to other visitor without any central server. It is decentralized web platform using Bitcoin cryptography and BitTorrent network. Currently while using ZeroNet protocol through existing browser user has to go through long and tedious process which could discourage a user to opt for ZeroNet in day to day use. We have developed a self-configuring browser especially for ZeroNet protocol which will encourage users to opt for ZeroNet.

Key words: ZeroNet, Browser Bundle, Peer-to-Peer, BitTorrent

I. Introduction

With the multiple-point failures and also the increasing amount of packets flooding a single server and effectively reducing its bandwidth, it is high time that the Internet needs to be transformed into a “By the People, For the People and from the People” concept.

ZeroNet Browser Bundle would effectively implement the ZeroNet protocol and would help manifest the importance of ZeroNet to the common masses. Some influential applications are:

- Figuratively zero point of failure.
- Easy Access of ZeroNet Websites.
- Download, Install and Run, simple steps to initialize ZeroNet.
- Easy Create Website and Update Website Plugins.

For services to become popular amongst the masses they should be easily available to everyone and also simple to use. Currently, ZeroNet uses Command line-interface for initialization; it also uses CLI for creating and publishing Websites. This is definitely a hassle for the non-technical person which effectively decreases the reach of ZeroNet. To avoid this, we are developing a browser bundle, which would simplify the process of using ZeroNet.

II. ZeroNet

ZeroNet is a decentralized network for peer-to-peer users. Currently, there is no self-configuring web-browser for ZeroNet. To setup ZeroNet requires a lot commands to be executed as well as configuring the current web-browser. This would be a problem for a non-technical person. If a normal user wants to access a ZeroNet site, it turns out to be a complex process.

III. Objective of the Study

Following are the objectives that were considered during the development of this project:

- Easy access of a free peer to peer network.
- Reducing configuration steps for accessing ZeroNet.
- Making ZeroNet available to majority of the masses.

IV. Proposed Methodology

From analysis we can say that current browsers do not satisfy the exact requirement of access to ZeroNet sites. Also, it solely depends on the active user to Configure their browser. Our proposed ZeroNet Browser overcomes these limitations of content based filtering such as the ability to provide recommendations is negatively affected if there are not enough user friendly features to distinguish the item that the user likes or dislikes. So we propose to work with modularized approach, creating following modules:

A. Visiting Websites

Visiting of a site in ZeroNet happens through peer exchange (PEX).

Peerexchange or PEX is a communications protocol that augments the BitTorrent file sharing protocol. It allows a group of users (or peers) that are collaborating to share a given file to do so more swiftly and efficiently.

In the original design of the BitTorrent file sharing protocol, peers (users) in a file sharing group (known as a "swarm") relied upon a central computer server called a tracker to find each other and to maintain the swarm. PEX greatly reduces the reliance of peers on a tracker by allowing each peer to directly update others in the swarm as to which peers are currently in the swarm. By reducing dependency on a centralized tracker, PEX increases the speed, efficiency, and robustness of the BitTorrent protocol.

B. Creating Websites

To create a website, currently the user run a command siteCreate which returns the public key and the private key of a newly created site. The user must save the Keys and must not disclose the private key to anyone. Once the user confirms that, he/she saved the keys, 2 files of the site are created i.e. index.html and content.json.

C. Updating Website

After changing the contents of a website, the user must sign it using the private key. Following is the format for site updating:

1) Parameter Description

a) SiteAddress

- 1EU1tbG9oC1A8jz2ouVwGZyQ5asrNsE4Vr)
- inner_path File path relative to site directory
- body Full content of the updated file

D. Publishing Website

Publishing a website means sending the updated contents of the website to all the serving clients. Currently the user must enter the command sitePublish to do so.

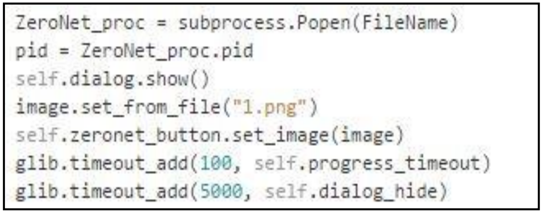
V. IMPLEMENTATION

The design of the proposed system has three main modules:

A. Browser Tab

This module is a PyGtk layout of every tab of the Browser, it includes functionalities like load_url, start_zeronet and history/bookmarks.

The most important feature is the ZeroNet Subprocess initialization which uses the Popen subclass from subprocess. It stores the process_id in a global variable which is useful later to stop or debug the ZeroNet subprocess.



```
ZeroNet_proc = subprocess.Popen(FileName)
pid = ZeroNet_proc.pid
self.dialog.show()
image.set_from_file("1.png")
self.zeronet_button.set_image(image)
glib.timeout_add(100, self.progress_timeout)
glib.timeout_add(5000, self.dialog_hide)
```

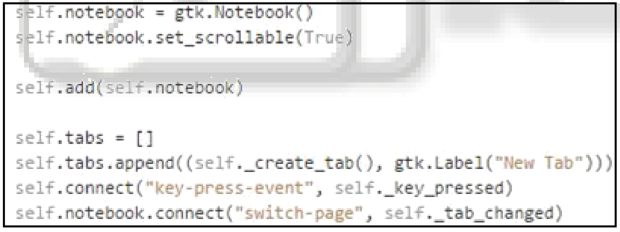
Fig. 1: Browser Tab

B. Browser

This python file is the primary file of the ZeroNet Browser Bundle which contains the rudimentary functionality required for the functioning of a normal Browser.

WebKit is imported in this file to make use of the WebView for displaying and rendering HTML pages.

It also makes use of NoteBook widget from PyGtk. Browser Tab tuples are appended in a python list which can be referred anytime to open/create/close the tabs

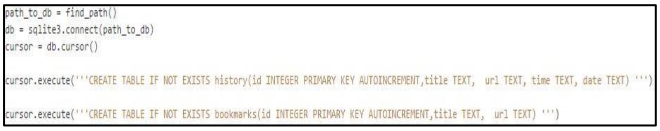


```
self.notebook = gtk.Notebook()
self.notebook.set_scrollable(True)

self.add(self.notebook)

self.tabs = []
self.tabs.append((self._create_tab(), gtk.Label("New Tab")))
self.connect("key-press-event", self._key_pressed)
self.notebook.connect("switch-page", self._tab_changed)
```

Fig. 2: Browser

C. Database

This is a python file which is responsible for the implementation of the database required for the browser. SQLite3 is used as the DDL, DML, DCL.



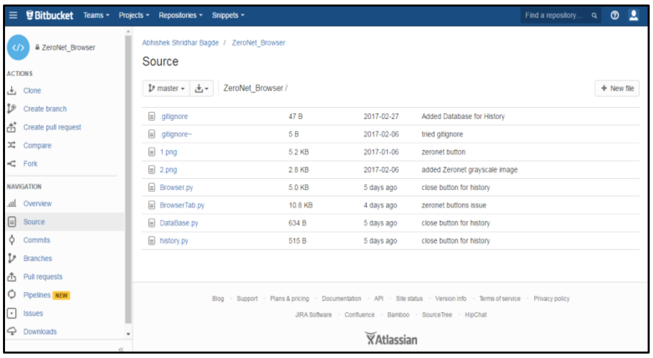
```
path_to_db = find_path()
db = sqlite3.connect(path_to_db)
cursor = db.cursor()

cursor.execute('''CREATE TABLE IF NOT EXISTS history(id INTEGER PRIMARY KEY AUTOINCREMENT,title TEXT,  url TEXT, time TEXT, date TEXT) ''')

cursor.execute('''CREATE TABLE IF NOT EXISTS bookmarks(id INTEGER PRIMARY KEY AUTOINCREMENT,title TEXT,  url TEXT) ''')
```

Fig. 3: Database

VI. COLLABORATION

Since this project is opensource, the online repository management system bitbucket.org is being used.Also,git is being used as the distributed version control system..

The following are the screenshots of the online repository:



Abhishek Shridhar Bagde / ZeroNet_Browser

Source

ZeroNet_Browser /

.gitignore	47 B	2017-02-27	Added Database for History
.gitignore~	5 B	2017-02-06	tried gitignore
1.png	5.2 KB	2017-01-06	zeronet button
2.png	2.8 KB	2017-02-06	added Zeronet grayscale image
Browser.py	5.0 KB	5 days ago	close button for history
BrowserTab.py	10.8 KB	4 days ago	zeronet buttons issue
DataBase.py	634 B	5 days ago	close button for history
history.py	515 B	5 days ago	close button for history

Fig. 4: Collaboration

VII. USER INTERFACE

The User Interface is designed for both ZeroNet and the normal client-server Internet, the functionality can be switched to ZeroNet using the ZeroNet Logo-Button provided at the top-right.

This Logo-button is used to toggle between ZeroNet and the mundane Internet. Additionally, Create, Update and Publish buttons have been provided for ZeroNet functionality.

The UI is as follows:

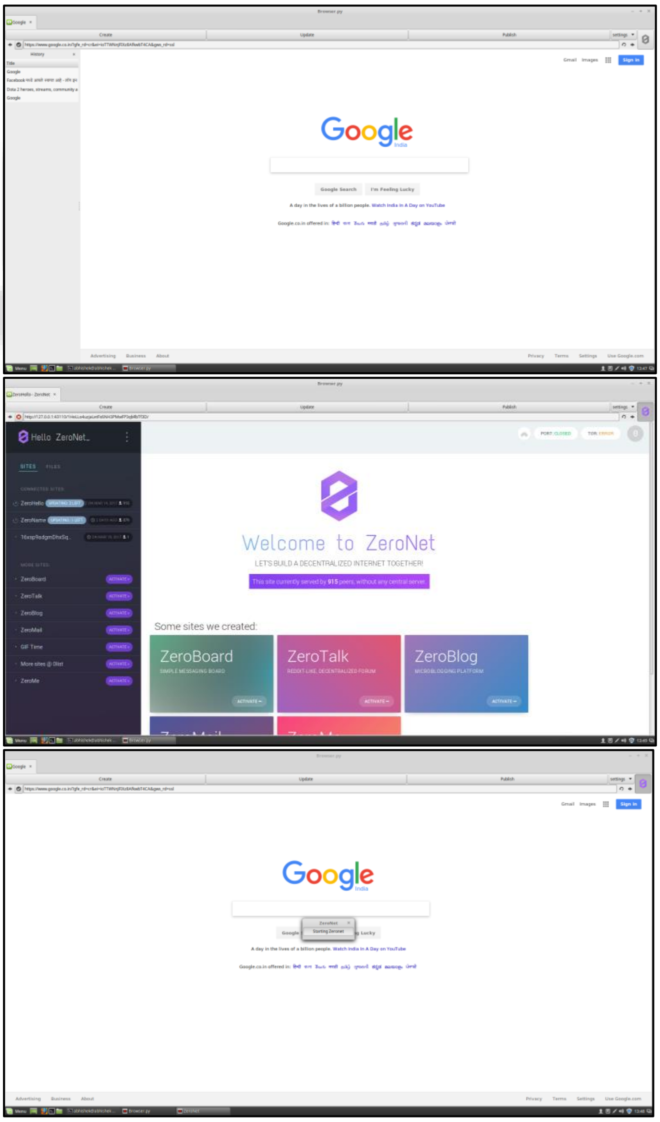


Fig. 5: UI

VIII. Conclusion

We have studied the ZeroNet network protocol along with how to utilize PyGTK to create user friendly GUI for Browser Bundle. We have also studied existing problem in ZeroNet Site Accessing and found counter measures. A study of existing browser systems and their compatibility with ZeroNet protocol helped us to reach the conclusion of creating a New Browser Bundle for ZeroNet Protocol. We have successfully created a functional browser which will act as basic building block in creation of ZeroNet Browser Bundle.

References

[1] Vivek Rai, Swaminathan Sivasubramanian, Sandjai Bhulai, Pawel Garbacki, Maarten van Steen;” A Multiphased Approach for Modeling and Analysis of the BitTorrent Protocol.”

[2] Tim Ruffing, Pedro Moreno-Sanchez, Aniket Kate; “CoinShuffle Practical Decentralized Coin Mixing for Bitcoin.”

[3] Ian Clarke,Oscar Sandberg,Brandon Wiley,Theodore W. Hong; “ Freenet: A Distrinuted Anonymous Information Storage and Retrieval System”