1. **Introduction**

While surfing the internet, privacy like our IP address (to keep from the website) and the content we are browsing (to keep from the network provider) are valued by people with certain purposes. Commonly, we use VPN to achieve such purpose, but it has its own limitation. What if the VPN server is compromised? What if an eavesdropper is doing traffic and timing analysis? Any of the circumstances coming to truth will jeopardize our location and communication. Hence, we come up with another solution – Tor network.

Tor network provides a bunch of servers to act as relays between the client and the host. None of the relays connects to the client or the host directly, instead virtual tunnel is used, thus protecting the IP address of both ends even though one of the relays is compromised. As several relays are used at the same time, it is hard to do the timing analysis, not to mention the intentional noise added to the network delay. The relays will try to conceal the information about the client, such as screen resolution, so that all users look the same to the eavesdropper, thus building barriers for traffic analysis.

To understand how Tor network works, we first start with the connection protocol on Tor – onion service protocol.

**2 Onion Routing**

Onion routing is a network connection between the client and the host by using multiple relays in between so that the actual location of the client is protected, and prevents traffic analyses and DNS attacks since the onion service’s address is encrypted as well. An onion service protocol should go through the following steps:

**2.1 Step One**

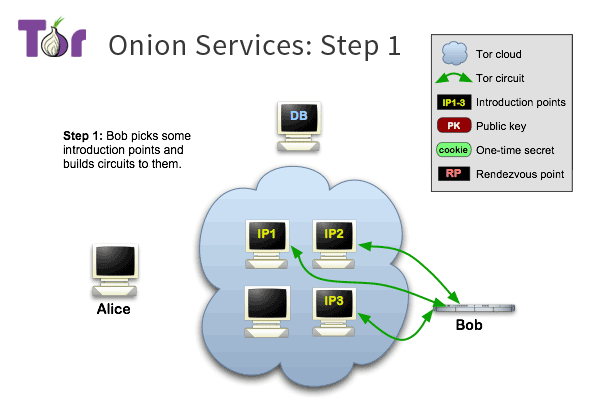


Figure 1. Onion services: step 1

To establish an onion service, it needs to announce its own existence to the Tor network. The onion service contacts several relays in the Tor network and asks them to act as introduction points by telling them its identity public key. The onion server’s location is secure with the introduction points because the identity public key is used instead of the actual IP address. From now on, only access from the introduction points will be approved so that the onion service is protected behind the Tor network.

**2.2 Step Two**

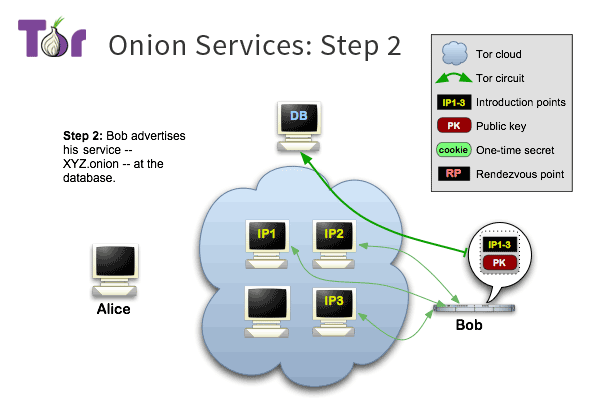


Figure 2. Onion services: step 2

The onion service publishes a descriptor containing a list of the introduction points, its public key and a signature signed by its private key. The descriptor is uploaded to a distributed hash table which serves as a dictionary for the client seeking access to the onion service.

**2.3 Step Three**

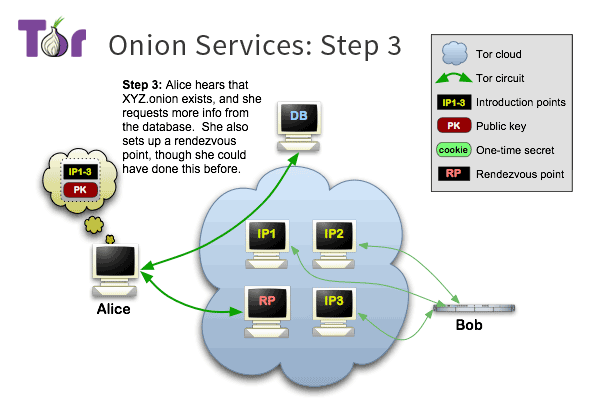


Figure 3. Onion services: step 3

The client requests the onion service’s descriptor from the distributed hash table, learns the introduction points, and verifies the signature with the public key. This procedure should ensure the end-to-end authentication security. Then the client randomly chooses a relay from the Tor network as the rendezvous point by telling it a one-time secret.

**2.4 Step Four**

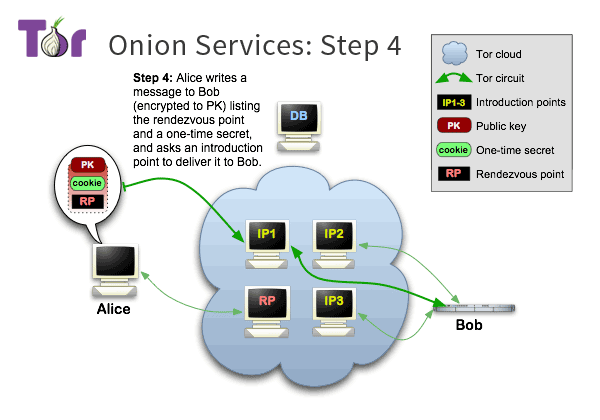


Figure 4. Onion services: step 4

The client composes a message with the address of the rendezvous point and the one-time secret, encrypt it with the public key, and send to one of the introduction points. Then the introduction point will transfer the message to the onion service.

**2.5 Step Five**

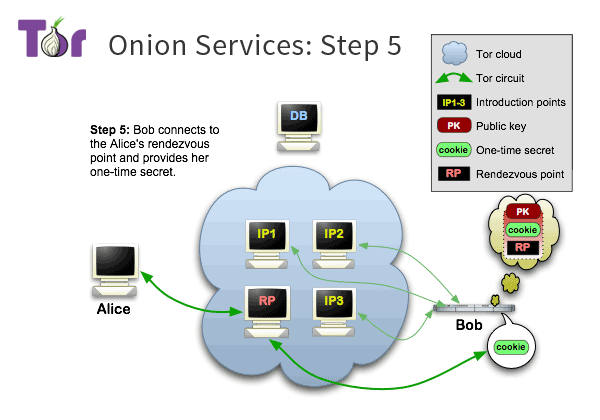


Figure 5. Onion services: step 5

The onion service receives the message from the introduction point and decrypts it with the private key. Then the onion service should connect to the rendezvous point specified in the message and send it the one-time secret in the message.

**2.6 Step Six**

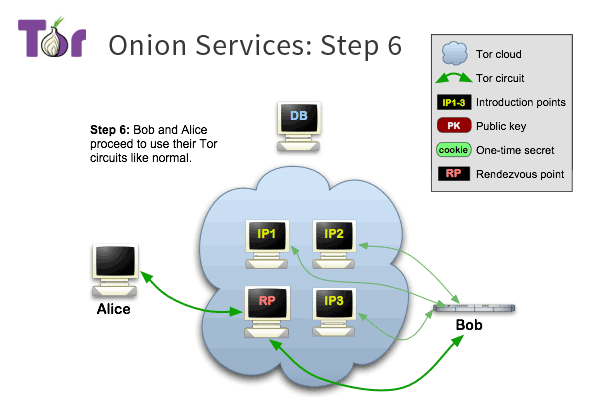


Figure 6. Onion services: step 6

Once the rendezvous point confirms that the one-time secrets from the client and the onion service are able to match, it notifies the client about the successful establishment of the connection. From now on, the client and the onion service can communicate through the circuits to the rendezvous point.

It is worth noticing that neither the client or the onion service connects to the rendezvous point directly. There are six relays in between: the first two are chosen by the client, the third is the rendezvous point, and the last three are chosen by the onion service, as shown in Figure 7.

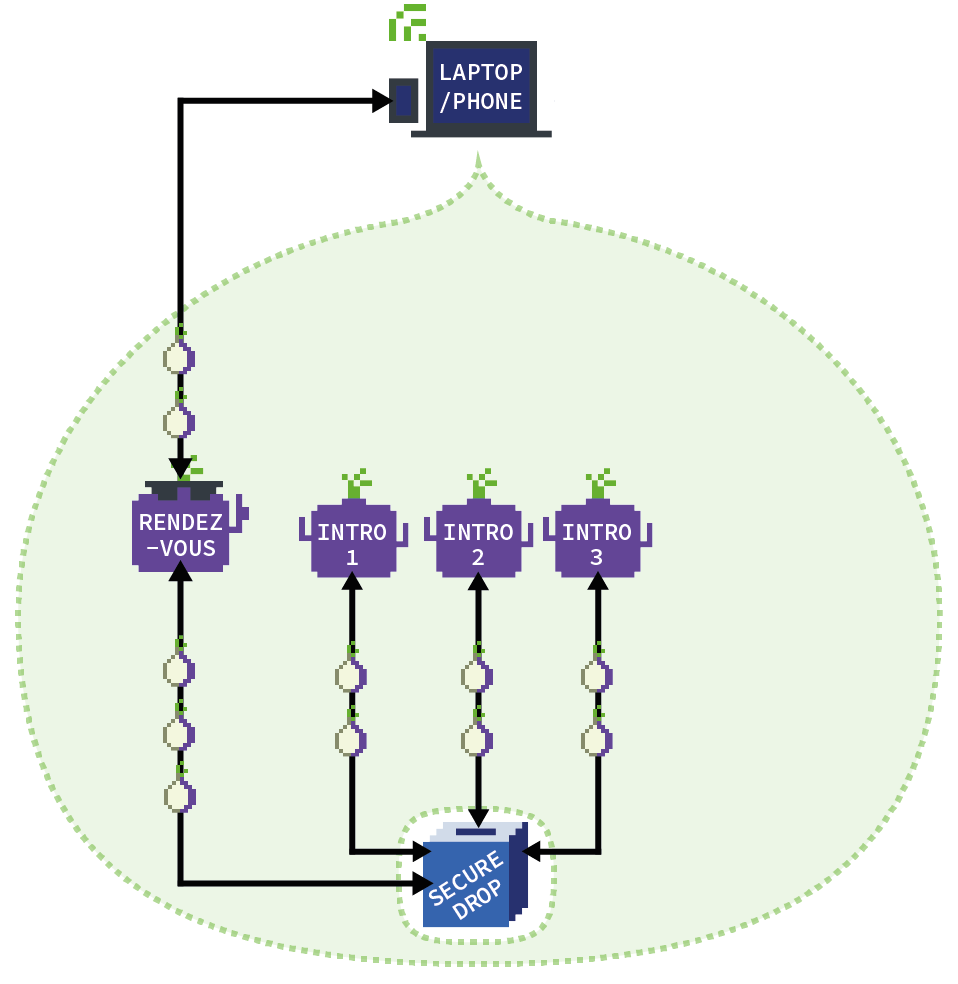


Figure 7. Full picture of onion routing

**References**

“Tor: onion service protocol”, *Tor Project*, 2019.www.torproject.org/docs/onion-services.html.en

“How do onion services work”, *Tor Project*, community.torproject.org/onion-services/overview/