Tor Hidden Services

Akash Patel (SJSU)

What is Tor hidden service?[1]

- Tor hidden service allow users to publish their service without revealing their identity (IP address).
- Users can connect to this service using rendezvous point without knowing the publisher of service and revealing their identities.
- This type of anonymity provides protection against distributed DoS attack as attacker would not know the IP address of service.

Four design goals of hidden service[2]

1. Access – Control: Publisher needs to way to filter incoming request so attacker cannot flood the service by making many connections to service.

2. Robustness - Publisher should be able to hide its identity for long time and service should not be bind with only one onion router, publisher should allow to migrate service to different onion routers.

3. Smear – resistance: Attacker should not be able to frame a rendezvous router by offering an illegal or disreputable location hidden service.

4. Application transparency: We are forcing user to access the service through tor network but we should not force publisher to make any changes in the application.

How Tor hidden service works?[3]

- Lets say Bob want to publish hidden service.
- Bob first need to deploy the service on the server. Then we need to allow Bob to select the contact point for the service, these contact points are known as introduction point.
- Bob's Tor client generate the public key and sends the key to introduction point. Bob made tor circuit with introduction point instead of direct connection so Bob's identity can be kept private. Introduction points only receive public key for the service not the IP address of the service.
- Hidden service assembles a hidden service descriptor which contains public key , introduction point and sign it with service's private key. This descriptor is uploaded in distributed hash table .
- This descriptor can be found by client by requesting xyz.onion where xyz is 16-character name derived from public key of the service.

How Tor hidden service works?[3]

- Lets say Alice has learned about Bob service and she want to access the service. She already knows descriptor of Bob's service
- Alice want to communicate anonymously. Her Tor client creates tor circuit by randomly picking relay and asks it to act as rendezvous point
- Alice send rendezvous cookie to one of the Introduction point. Cookie contains information about rendezvous point and introduction message encrypted using service public key. This communication take place via Tor circuit
- The hidden service get the request and obtain address of rendezvous point and send the one time secret to it in a rendezvous message.
- At this point it is very important that hidden service sticks to same set of entry guard when creating new circuit. Entry guard is set of relays which is always picked as first node while creating circuit. These entry guard are chosen randomly

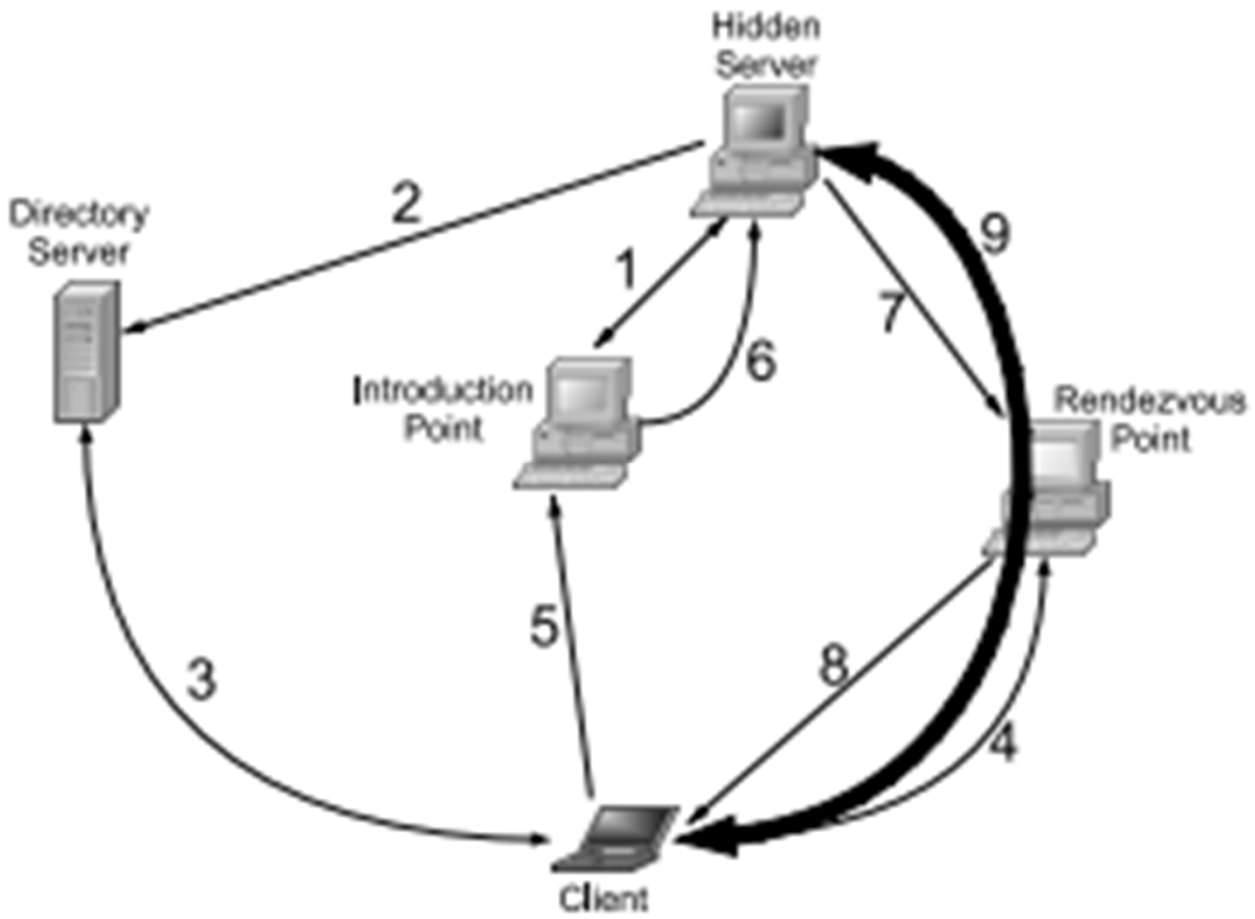
How Tor hidden service works?[3]

- In the last step, the rendezvous point notifies the client about successful connection establishment. After that, both client and hidden service can use their circuits to the rendezvous point for communicating with each other. The rendezvous point simply relays (end-to-end encrypted) messages from client to service and vice versa.

- Once rendezvous point get the response from service it informs client that connection is established and now client and service can talk with each other through rendezvous point using their circuit.

- In general there are 6 relays used in end to end communication. Three of them are chosen by service (a.k.a introduction point) and three are chosen by client including rendezvous point

Overview diagram[4]



How to create Tor Hidden Service[5]

- Install web server locally
- Configure hidden service to point to local web server
- Open torrc file. It is located at \Tor\Tor Browser\Data\Tor \torrc
- Now add entry like below in torrc file
  - HiddenServiceDir D:\Tor\HiddenService
  - HiddenServicePort 80 127.0.0.1:8080
- Hidden Service dir is place where Tor store information about hidden service. Tor will generate hostname file in this folder which contains onion URL for service
- HiddenservicePort is used to specify virtual port, IP address and port for redirecting connections to this virtual port.
- Save torrc file and restart tor client

References

1. https://www.torproject.org/docs/hidden-services.html.en

2. Technical Paper: Section 5 of Tor: The Second-Generation Onion Router by Roger Dingledine, Nick Mathewson and Paul Syverson.

3. Technical Paper: Section 5.1 of Tor: The Second-Generation Onion Router by Roger Dingledine, Nick Mathewson and Paul Syverson.

4. Technical Paper: Figure 1 of Locating Hidden Servers by Lasse Øverlier and Paul Syverson

5. https://www.torproject.org/docs/tor-hidden-service.html.en