VE475 Project2: Tor Network and Potential Attacks

Taoyue Xia, 518370910087
Wenjie Xianyu, 518370910133

July 2021

Abstract

In this project we look into the Tor project. Firstly, we give an introduction of the Tor network. Then we dig deep into “onion routing”, which is the realization of an anonymous network. Finally, we will present two practical attacks on the Tor network, the traffic-analysis attack and the bad apple attack, which shows some kind of weakness of the Tor network.

In such an information era, people needs to connect the internet for searching, online shopping, etc., which makes it essential for people to maintain privacy for the sake of data security. This kind of concerning prompts the creation of the Tor network which can preserve users’ privacy and anonymity by the important application of onion routing. Security is always the main consideration of cryptography.

Moreover, cryptography is such a comprehensive subject with super magic, and the exploration of it will never stop.

Key Words: Tor network, onion routing, privacy, traffic-analysis attack, bad apple attack

Contents

1 Introduction of Tor Network 4

2 Onion Routing 5

2.1 Step One . . . 5

2.2 Step Two . . . 6

2.3 Step Three . . . 6

2.4 Step Four . . . 7

2.5 Step Five . . . 7

2.6 Step Six . . . 8

3 Attacks on Tor Network 9

3.1 Traffic-analysis Attack . . . 9

3.1.1 Traditional . . . 9

3.1.2 Setup . . . 9

3.1.3 Methodology . . . 10

3.1.4 Results . . . 11

3.2 Bad Apple Attack . . . 11

3.2.1 BitTorrent . . . 11

3.2.2 Setup . . . 12

3.2.3 Results . . . 12

4 Conclusion 13

Reference 14

1 Introduction of Tor Network

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

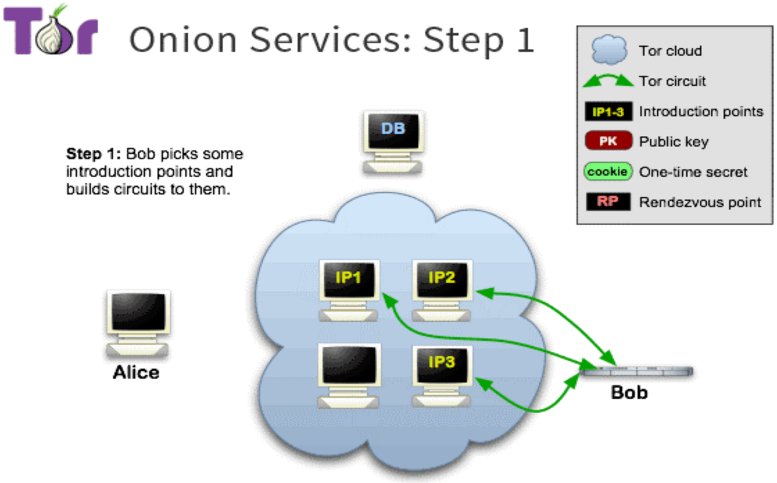


Figure 1: Onion services: step 1 [1].

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

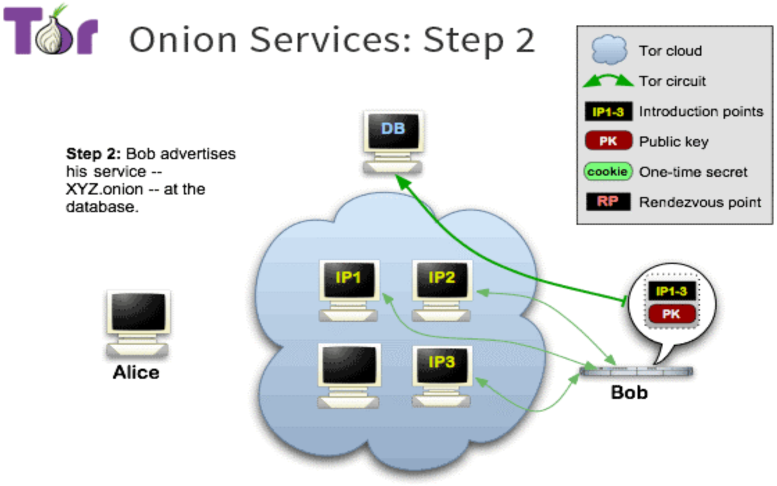


Figure 2: Onion services: step 2 [1].

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

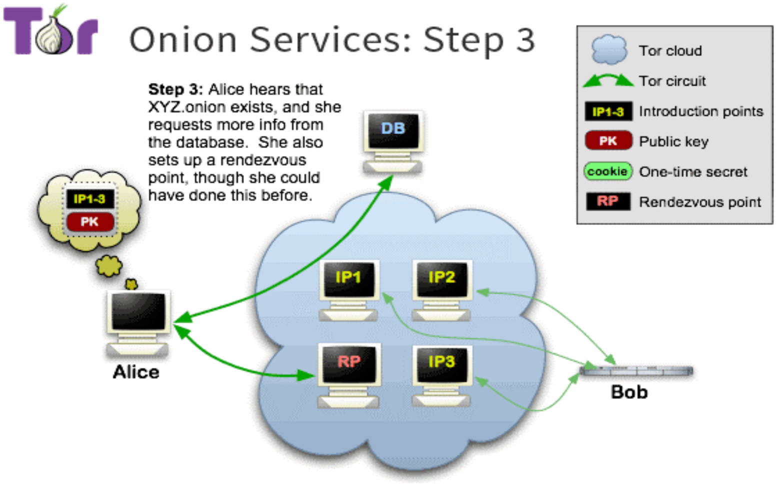

Tor
Onion Services: Step 3
Tor cloud
Tor circuit
IP1-3 Introduction points
PK Public key
cookie One-time secret
RP Rendezvous point
Step 3: Alice hears that XYZ.onion exists, and she requests more info from the database. She also sets up a rendezvous point, though she could have done this before.
DB
IP1-3
PK
IP1
IP2
RP
IP3
Alice
Bob

Figure 3: Onion services: step 3 [1].

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

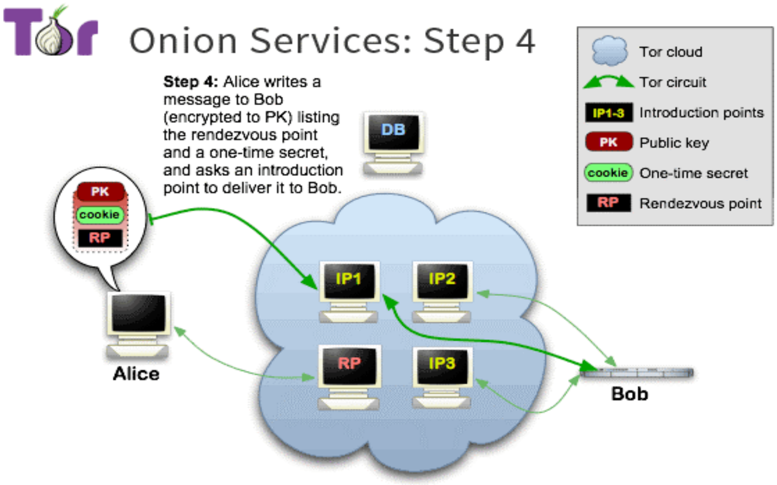

Tor
Onion Services: Step 4
Step 4: Alice writes a message to Bob (encrypted to PK) listing the rendezvous point and a one-time secret, and asks an introduction point to deliver it to Bob.
DB
PK
cookie
RP
IP1
IP2
RP
IP3
Alice
Bob
Tor cloud
Tor circuit
IP1-3 Introduction points
PK Public key
cookie One-time secret
RP Rendezvous point

Figure 4: Onion services: step 4 [1].

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

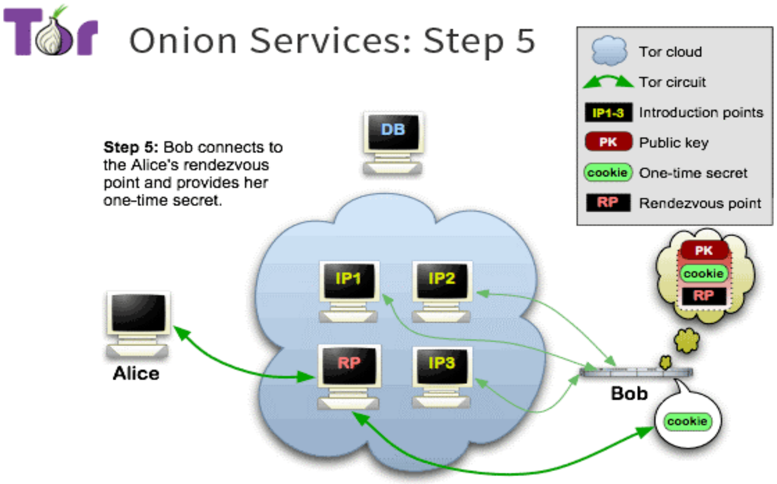


Figure 5: Onion services: step 5 [1].

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

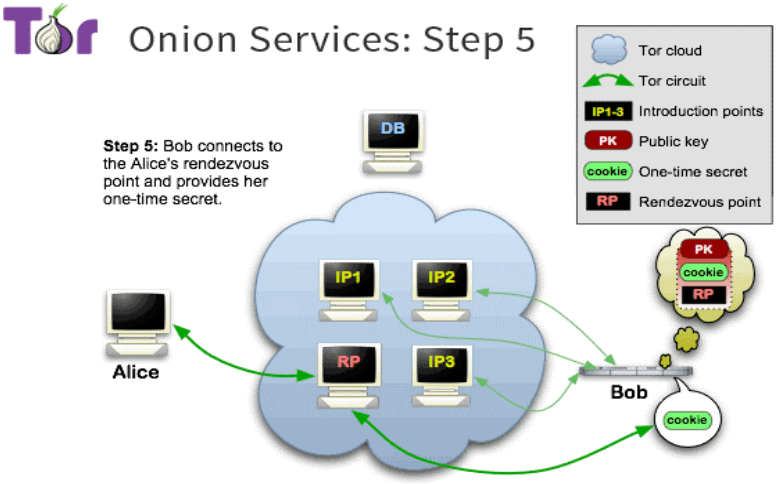


Figure 6: Onion services: step 5 [1].

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.

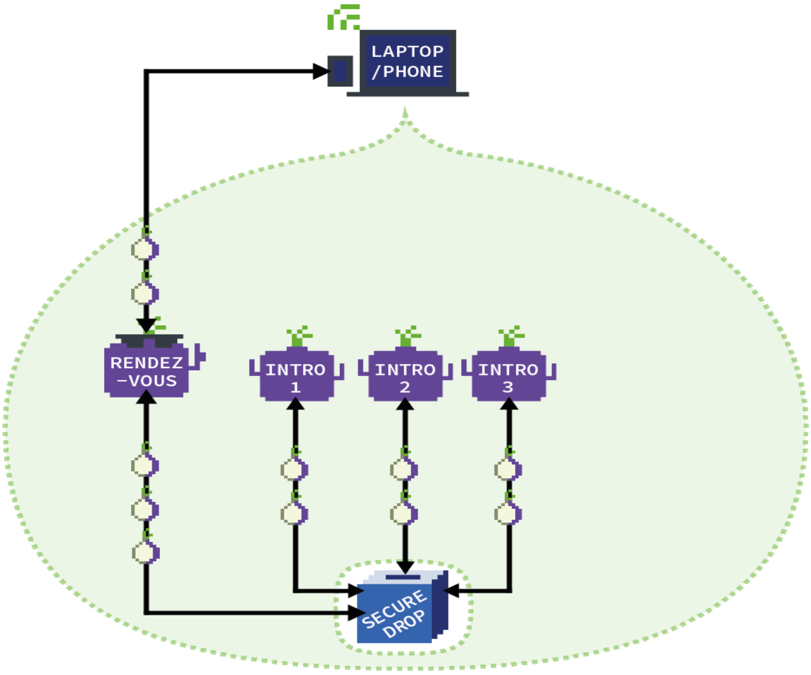


Figure 7: Full picture of onion routing [2].

3 Attacks on Tor Network

Though Tor network with the onion routing is considered to be relatively secure and hard to track the user activities, due to the pursue of low latency on communication, some attacks can take place.

In this section, we will look into some practical attacks on the Tor network, which reveals some problems of the construction.

In general, the main goal of an adversary attacking an anonymous communication system is to find out the originator or the accessing remote machine. Then in some specific ways they can pretend as they are the originator, and transmit false information to the receiver.

The following attacks are run against Tor in order to make the Tor project more secure and can improve anonymity to some extent, thus protecting users' profiles and data.

3.1 Traffic-analysis Attack

Tor network is assumed to hide the bit pattern of stream transmitted in the encrypted tunnel. Therefore, the information is of no use to the attackers to trace the stream. In this sense, traffic-analysis attack is about to work.

Traffic-analysis is the method to extract information from the network meta-data, containing the volumes and timings of network packets, along with original and destined network addresses [3].

3.1.1 Traditional

Traditional traffic analysis can be classified by the degree of granularity. The first type of attacks treats anonymous network as "black box", and focuses on the moments when users initiate connections, which are relayed out of Tor network. This kind of attacks can reveal repeated patterns of communications over Tor network. Another type of attacks operates on a smaller scale, they intercepts communications within the anonymous network. To trace an onion-routed stream in the Tor network, a global server can correlated the timing and volume of in and out streams [3].

However, these types of attacks require observing all nodes and network links, which is out of Tor's threat model and are of great difficulty. Therefore, we looks for a simpler attack which can be carried out much easier, as the following part gives.

3.1.2 Setup

Steven et el. [3] carried out the new-type traffic-analysis attack which enables attackers to track the path of connections between initiator and receiver with little capability.

It is first noted that since the load on Tor nodes can affect the latency of all streams transmitting on this node, so attackers can use all the connections routed through this node. Assume the attacker controls a corrupted node, which generates a connection that passes another Tor node, and the traffic load of this node is to be measured. Then a sequence of bursty data was input and the correlation between traffic volumes and data is going to be detected. The setup procedure is shown in the following figure.

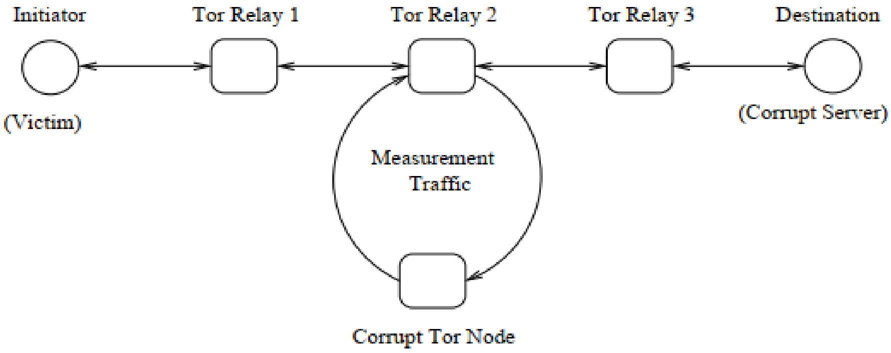


Figure 8: The attack setup [3].

3.1.3 Methodology

An attacker's aim is to figure out which nodes are carrying traffic with the pattern injected by the corrupt server using timing data from all nodes on the network. Two tests are run for each node: one in which the stream from the corrupt server passed through the target node, and one in which it did not. The correlation was fairly straightforward: the template generated by the corrupt server's modulated traffic was multiplied by the probe data, and the sum was calculated [3]. Define the function of the corrupted server $S(t)$ as:

$$S(t) = \begin{cases} 1, & \text{if server is sending at sample number } t \\ 0, & \text{otherwise} \end{cases}$$

The data from the probe is denoted as $L(t)$, which shows the latency of the target Tor node at sample t (in μs). The coefficient c, is the sum of the product of $S(t)$ and normalized $L(t)$, which is shown as $L'(t)$, divided by the number of samples the server is sending:

$$c = \frac{\sum S(t)L'(t)}{\sum S(t)}$$

A TCP server was used to imitate the corrupt server, which would send pseudorandomly produced data at the fastest speed allowed by Tor for a pseudorandom time period (between 10 and 25 seconds in the experiment), then stop sending for another period (between 30 and 75 seconds) [3].

3.1.4 Results

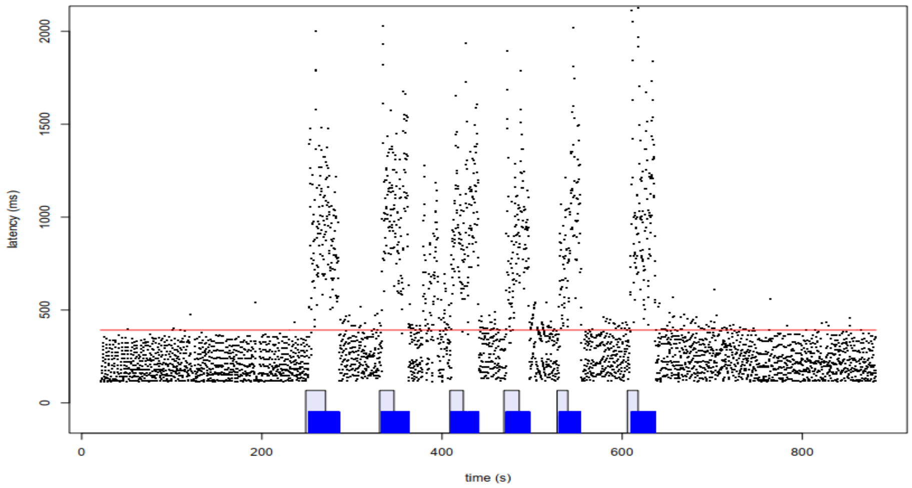


Figure 9: Probe results showing good correlation [3].

The dots represent the probes' latency, while the pattern of the victim stream transmitted is displayed at the bottom. The victim stream is overlaid to demonstrate how the network distorts the pattern [3]. What's more, the horizontal red line denotes the mean latency of sample period [3]. From Figure 9, we can see a good correlation between probe data in victim traffic.

In conclusion, since Tor's low latency requirement prevents it from shaping traffic in any manner, and concealing these features would necessitate a huge quantity of cover traffic. Therefore, attackers can have a discovery of the Tor server used to initiate the traffic stream, which is dangerous for users since attackers can forge their identities to communicate with the target remote client.

3.2 Bad Apple Attack

The name bad apple attack originates from the old saying "one bad apple spoils the brunch", which tells us that some insecure application can affect the anonymity provided by the Tor network.

Based on the fact that many Tor users use BitTorrent which is not secure, and Tor is not supposed to protect users from application-level attacks, it is possible for attackers to analysis the activities happened in BitTorrent using, and track the trace in Tor network, and get user data to some extent.

3.2.1 BitTorrent

BitTorrent is a popular Peer-to-peer (P2P) protocol for file replication [4]. To download a file, a BitTorrent client first use centralized trackers, a distributed tracker (DHT) and peer exchange (PEX) to look for peers sharing that file. The tracker receives an IP/port subscription from a peer in the case of a certain content identifier, and then send requests to peers who share the downloading of the file. Since the communication with DHT is done via UDP, the user datagram protocol, the IP address of a Tor user can be subscribed to the DHT tracker, thus causing severe information leak.

Finally, a Tor user using BitTorrent will be linked to more peers according to PEX, and the trace can be detected from the IP addresses.

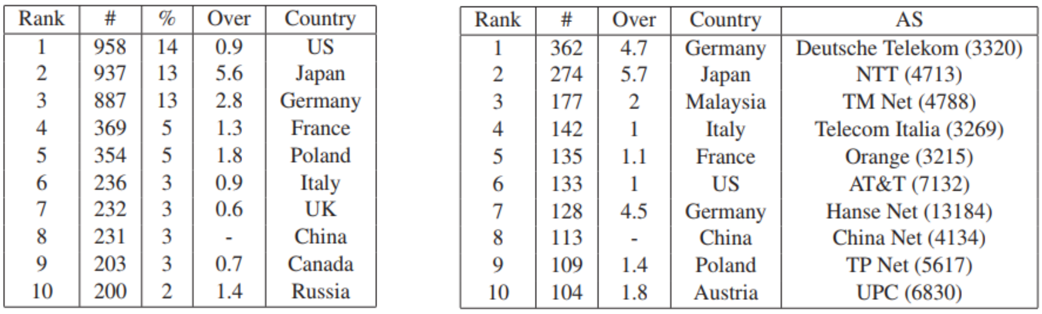
3.2.2 Setup

In Steven et el.'s work [4], they control and monitor several exit nodes of Tor, and conduct the first type of attack by hijacking tracker's responses, which inserts the IP or port of a virtual peer pretended by an attacker to the list of peers sent back by the tracker. If users use Tor to connect directly to the centralized tracker, then the IP address connected to the malicious peer will not be included in the exit nodes, namely direct connection. Thus it is more efficient to launch such type of attack, since attackers do not need to track the separated data in the Tor network, which is hard to analysis.

The second type of attack conducts a statistical exploitation of DHT. Stevens et el. [4] look into the BitTorrent subscription, using content identifier and listening port number to the extended handshake messages. When a message is received on an exit node, all the IP addresses of subscribers will be collected.

3.2.3 Results

It is estimated that 19% of all streams on Tor are BitTorrent streams based on the traffic relayed by the exit nodes. Furthermore, the successfully traced 9% of streams reveal 10,000 IP addresses of Tor users, along with the content these users attempt to download.



Rank	#	%	Over	Country
1	958	14	0.9	US
2	937	13	5.6	Japan
3	887	13	2.8	Germany
4	369	5	1.3	France
5	354	5	1.8	Poland
6	236	3	0.9	Italy
7	232	3	0.6	UK
8	231	3	-	China
9	203	3	0.7	Canada
10	200	2	1.4	Russia

Rank	#	Over	Country	AS
1	362	4.7	Germany	Deutsche Telekom (3320)
2	274	5.7	Japan	NTT (4713)
3	177	2	Malaysia	TM Net (4788)
4	142	1	Italy	Telecom Italia (3269)
5	135	1.1	France	Orange (3215)
6	133	1	US	AT&T (7132)
7	128	4.5	Germany	Hanse Net (13184)
8	113	-	China	China Net (4134)
9	109	1.4	Poland	TP Net (5617)
10	104	1.8	Austria	UPC (6830)

Figure 10: Popularity and over-representation of BitTorrent users on Tor per country [4].

Figure 10 shows the popularity rank of countries in which Tor is used by BitTorrent users to download contents. We can see that In Japan and Germany, the number of Tor BitTorrent users in Tor is multiples of those outside of Tor. We can see that more and more people emphasize on anonymity when surfing the internet. However, Tor network still has much room for improvement on protecting the anonymity of users.

4 Conclusion

In this project we first get an outline of Tor network and its features. Then we look into the most important basic construction, onion routing, to get a further understanding of how communication in the Tor network is realized, and how can it preserve users' anonymity by doing so. What's more, going from the fact that there are no 100% secure systems, we choose two attacks which have been carried out on the Tor network, the low-cost traffic-analysis attack and the bad apple attack, and demonstrate their realization procedure in detail.

The Tor network aims to make communication between people be more secure and remain anonymous, which is for the sake of human being to protect their privacy. Onion routing is an epoch-making implementation, which makes it quite hard for attackers to track users' information, so that normal people, who are in the main targeted user group, can keep their internet activities quite private from malicious websites, passive adversaries and cyber-spying. However, due to the pursue for low latency, the Tor network is constructed in a different way than mixing, which makes it vulnerable to some specific attacks. In the purpose of making Tor better, experimental attacks like the above two are conducted, revealing the weakness of Tor network, thus the technical staff can fix them and better protect users' privacy.

Although it is fantastic in protecting users' anonymity, Tor network is also used for illegal activities, like child abusing and drug selling. However, this is inevitable since criminals can do bad things no matter what they can use, and it is impossible to ensure that everyone is ethical and follows good manner. Therefore, Tor network is still a great invention aiming for the sake of people's privacy.

References

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

[2] “How do onion services work” , Tor Project, https://community.torproject.org/onion-services/overview/

[3] S. J. Murdoch, G. Danezis, “Low-Cost Traffic Analysis of Tor,” IEEE CS. IEEE Symposium on Security and Privacy, May 2007.

[4] S. L. Blond, P. Manils, A. Chaabane, K. M. Ali, C. Castelluccia, A. Legout, and W. Dabbous, “One Bad Apple Spoils the Bunch: Exploiting P2P Applications to Trace and Profile Tor Users,” 4th USENIX Workshop on Large-Scale Exploits and Emergent Threats (LEET ’11), National Institute for Research in Computer Science and Control, April 2011.