Simulation of Anonymity Network Model

Omkar Ghaisas, Irfan Siddavatam, Ashwini Dalvi, Faruk Kazi

Abstract*: In order to protect privacy a user might want to be anonymous over internet. Tor is most used network when it comes to achieving internet anonymity. But Tor network itself is not secure. If tor network gets compromised then anonymity of Tor client will get compromised which means ip address of user's computer will be noted at Website interface. Hence we have proposed a model which gives advance anonymity to the Tor client. The model consists of three components Local Host, Remote Terminal and Target Website. Due to inclusion of Remote Terminal even though Tor gets compromised only ip address of Remote Terminal gets noted at the website interface. Hence user's ip address is never noted at web interface or user is anonymous. This paper provides simulation tests on anonymity model using shadow. In Simulation Results we have graphs which show different relationships between Benchmark parameter which are throughput, goodput, ticks, control overhead and retransmission overhead.

***Index Terms*: Internet Anonymity, Tor, Shadow, Privacy.**

I. INTRODUCTION

Anonymity can be required by lot of people. Normal people may use anonymity to protect their privacy from marketers and ISPs who sell their records. They may also use when searching sensitive topics such as wealth related topic. Tor network is also used to circumvent government censorship.

Anonymity network is also required by activist and whistle blowers in order to stay anonymous. Government officials and journalists as well as their audience. Anonymity network has extensive need in military operations.

A. Anonymity Networks

Normally when a client directly access a website or web server through normal network then their ip gets noted at the server side. But if client wishes to connect the server anonymously then the ip address of the client needs to be spoofed. In order to achieve anonymity following methods are available

- VPN- VPN or virtual private network allows users to tunnel their website request through another computer. But this technology has a drawback that our IP address is noted at the VPN server.
- Proxychain- Proxy chain allow us to connect to the chain of proxy servers. Hence our request will go through one proxy to another and finally to the website or web server. Again our ip address is noted at the Proxy server.
- TOR network- TOR or the onion router uses onion routing in which request is sent in layers of encryption and each layer is peeled at tor node. Tor is an Internet networking protocol designed to anonymize the data relayed across it. Currently TOR is most popular method to stay anonymous over internet.
- I2P network - I2P or internet invisible proxy uses garlic routing. It is a decentralised network. I2P is still in developing stage.

Here Tor is the most commonly used network for achieving anonymity. The model proposed in this paper is also based of Tor network.

B. Tor Network

Tor is anonymity and censorship circumvention tool. Tor nodes create a circuit which gives anonymity between website and client. Tor clients connect to relays which are intermediary computers which can be used to create circuits for anyone who needs it. Anonymity in Tor is achieved by using onion routing protocol.

In Onion Routing the internet traffic is wrapped up in multiple layers of encryption. As seen in Fig. 1. at each end or node one layer of encryption is peeled off as traffic is sent across the network.

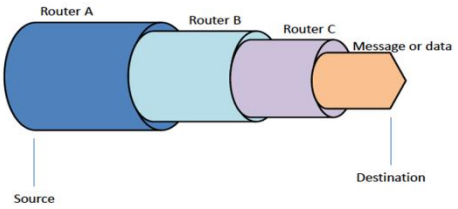


Fig 1. Onion Routing Architecture.

C. Tor network working

When a client wants to connect to Tor network it gets the list of Relays from the Directory Authority. In Tor network there is a Directory authority which contains a list of relays or Tor nodes. Tor relays are computers over the internet which allows client traffic to route through Tor network.

Then the client goes through three hops across these relays. It includes entry hop which is called transit node, it takes another hop in Tor network which is also called transit node and takes final hop which is called exit node.

Revised Manuscript Received on May 10, 2019

Omkar Ghaisas, Information Technology Department, K J Somaiya College of Engineering, Mumbai, India.
Dr. Irfan Siddavatam, Information Technology Department, K J Somaiya College of Engineering, Mumbai, India.
Prof. Ashwini Dalvi, Information Technology Department, K J Somaiya College of Engineering, Mumbai, India.
Dr. Faruk Kazi, Electronics dept, Veermata Jijabai Technological Institute, Mumbai, India.

At Tor client side the traffic is encrypted with Three layers of encryption. A layer is peeled off at each hop. After exit node Tor traffic is sent to web server in unencrypted form.

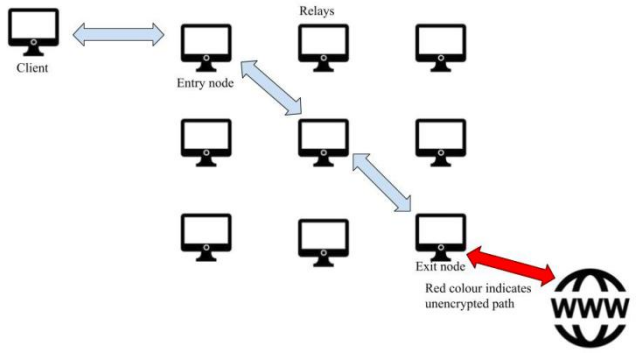


Fig 2. Tor Network Architecture.

The working is given in Fig. 2. where the red arrow represents unencrypted traffic.

Tor itself is not secure. Tor network can be compromised which leads to compromise in users anonymity. Drawbacks of Tor network are shown in Literature Survey. In further sections we have proposed a model which can provide advance anonymity by overcoming Tor drawbacks along with Simulation Tests performed on that model using shadow.

II. LITERATURE SURVEY

Most popular choice for anonymity is Tor network. Tor network provides anonymity to a client which can surf over the internet while staying anonymous to the browser[1]. This is done by routing traffic through Tor relays and exits through tor exit nodes. But if aggregate frequency of usage in 2015 for each exit node is calculated then we get that 20.65% exit nodes used are from USA, 14.44% are from Germany and 13.20% are from switzerland. This shows that few countries are dominant when it comes to exit nodes of Tor network.[2] This poses the risk of traffic analysis by organizations and ISPs of these countries. This limits significantly the set of Tor exit nodes which can be used with confidence. The role of Carnegie Mellon University and the FBI in the 2014 Tor attacks using compromised exit-node selection shows the significance of the compromised Tor node.[3]

After Edward Snowden's disclosure we know now that NSA is actively working against privacy advocates and interested in getting as much information as possible. By using compromised exit nodes NSA can target a user if another node is in control of NSA as well. That means if two Tor Relays of Tor are compromised then anonymity of user is compromised. It is not sure if NSA is ahead of academic world. We can be protected by Tor from mass surveillance, but if NSA directly targets a user then it is not sure if Tor alone will be able to protect users anonymity.[4]

As http and https application have more routers available in Tor network it is least susceptible to path compromise by traffic analysis. For http there are 625 whereas for https there 629. The other applications over Tor are used much less as compared to http and https and hence they are much susceptible to path compromise by traffic analysis. Some example would be bitTorrent and eDonkey which have 23 and 20 routers respectively.[5] This attack comes under scalability attack where attacker creates a malicious tor node and runs it on unpopular post like port 25 or port 119 instead of popular ports 80 and 443. Then the attacker will create a tor node which will get the traffic from popular port and forward it to unpopular port. This attack can compromise anonymity of 50% tor users.[6]

Tor has now spread to 75 countries. This has increased the rate of adding malicious nodes in the network. Tor relays are selected on the basis of relationship of security and bandwidth. Initially bandwidth assigned to a relay was 0.1 gbps. Now there are relays which offer huge bandwidths upto 4.5 gbps. If a relay with huge bandwidth and low security then Tor will still select that compromised relay.[7] From simulations it is derived that catch probability or probability of getting compromised on Tor network is independent of geographical location. But it is highly dependent on bandwidth and increases when bandwidth increases from 10 mbps to 50 mbps but significantly increase if we add more number of 10 mbps nodes.[8]

Other attacks which can compromise anonymity ae plugin based attacks, torben attack where a attacker can exploit low latency feature of tor to get its surfing details[9], induced torguard selection where malicious entry node is advertised, exploiting BGP protocol by exploiting natural churn or by BJP hijacking. [10] There is a whole set of attacks called Raptor or Routing Attacks on Privacy in Tor. These attacks includes asymmetric traffic analysis and BGP churns.[11]Active website fingerprinting attack is possible by actively delaying http requests with 95% accuracy and 3% false positives[12].

III. ANONYMITY MODEL

In normal Tor configuration user goes through Tor network and connects to Target Website or the website user wants to access. This is shown in Fig. 3.

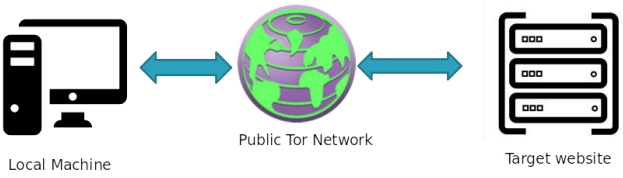


Fig 3. Tor Browsing.

As we have seen in Literature Survey that there are various factors which can lead to compromise in our anonymity through Tor network. So we proposed advance anonymity model to achieve further anonymity.. Proposed model has three Machines which are local machine, Remote Terminal and Target website. Advance Configuration is shown in the Fig. 4.

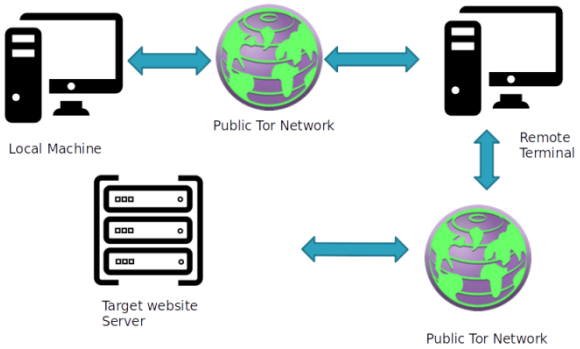


Fig 4. Advance Anonymity Configuration.

Here the client which is termed as local host goes through Tor network and connects to a Remote Terminal. From Remote Terminal traffic is redirected to target website server over Tor network again. We can explain working of this model in Algorithm as follows

Algorithm 1: Advance anonymity configuration.
1: start tor proxy
2: if tor proxy==open
3: start VNC Remote connection
4: if VNC connection to Remote Terminal==open
5: start Tor Browser
6: if Tor Browser==open
7: surf Target Website
8: Download necessary data
9: exit Tor browser
10: Upload data to Local Host
11: exit VNC Remote Connection
12:exit Tor Proxy

IV. SIMULATION

A. Simulation Tool

We have used shadow simulator to simulate our anonymity model. Shadow is a discrete event simulator that can run real applications as plug-ins while requiring minimal modifications to the application. Shadow is used to simulate Tor network. Shadow plugin is used to simulate Tor network topologies using latest Tor software. Plug-ins containing applications link to Shadow libraries and Shadow dynamically loads and natively executes the application code while simulating the network communication layer.[13]

B. Simulation Model

In shadow we have simulated our anonymity model for running simulated tests. Here we have created two servers namely server and host. We have created a node called remote. The node called remote downloads 100 mB of data from server. It does this in non bulk fashion by taking pauses. Later it uploads all data to host server. This is done using bulk fashion or at a time all data is uploaded. The topology can be shown in Fig. 5.

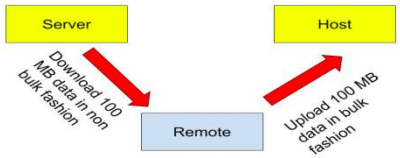


Fig 5. Shadow Configuration.

Here in Fig. 5 we can see blue box represents node whereas yellow box represents server. Downloading of data is done in non bulk fashion which means it was downloaded with taking pauses. After each 100KB download of data there was 60 second pause and again next download took place. The upload was done in bulk fashion as in there were no pauses. This is because while browsing a user might surf and take time hence we have simulated download in nonbulk fashion. But while uploading user would take complete dump directly from remote hence there are no pauses during upload.

C. Simulation Results

Here the simulation was executed twice while keeping tcp window as 1 packet and then as 1000 packet. Graphs are generated which compare results of both tcp windows. After running simulation of above configuration in shadow simulator we got multiple graphs as output. These graph determine relationship between different parameters.

In our anonymity model user goes through Remote Terminal and opens Tor Browser. Here user can browse over internet and start surfing. Surfing over internet is done in nonbulk fashion and hence in simulation we have simulated download from Server to Remote in nonbulk fashion. Here Remote downloads 100 MB from Server in nonbulk way which includes downloading 100 KB and taking pause for 60 seconds. After pause again 100 KB will be downloaded.

First graph which is seen in fig. 6 is throughput vs ticks. Here throughput is the amount of data moved from one tor node to another tor node. Ticks is number of CPU cycles done from the first epoch. They are represented in seconds and hence we can also consider ticks as time.Fig. 6 shows relationship between throughput and ticks for downloading data from Server to Remote Terminal.

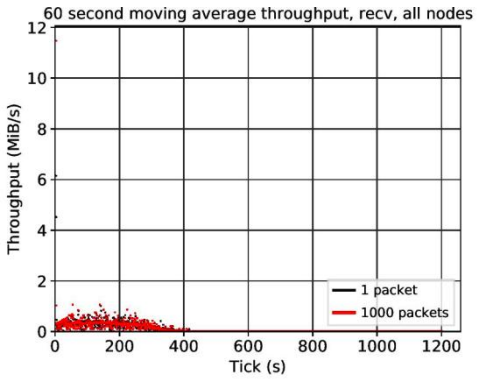


Fig 6. Relationship between Throughput and Ticks for download.

This shows that for downloading data throughput is 0.5MiB/s. Hence there is no impact of TCP window on nonbulk operation. We can also see that it takes 400s to complete download. Fig. 8 gives relationship between Goodput and Ticks for download of data. Goodput is the amount of useful data moved from one tor node to another tor node.

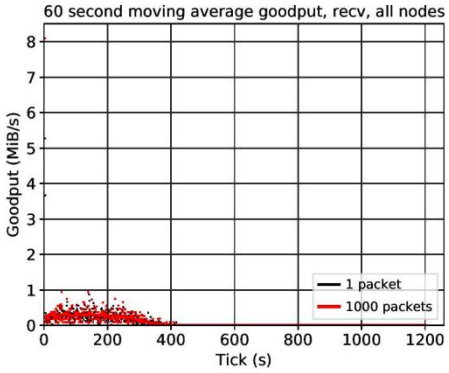


Fig 7. Relationship between Goodput and Ticks for download.

Here we can see that the Goodput is 0.4 MiB/s for downloading of data and also for Tcp window 1 and 1000. Fig. 8 shows relationship between Control Overhead and Ticks for download. Control overhead is excess bandwidth required for downloading or uploading data.

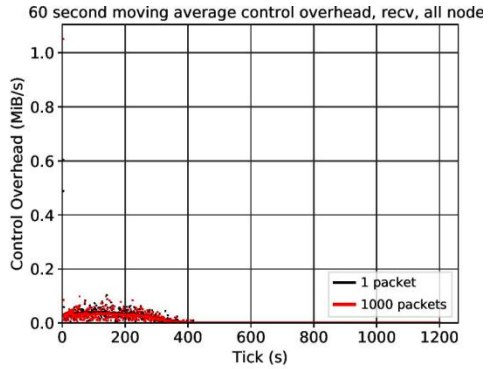


Fig 8. Relationship between Control Overhead and Ticks for download.

Control Overhead can be seen as 0.075 MiB/s for downloading of data. Fig 9 shows relationship between retransmission overhead for download.

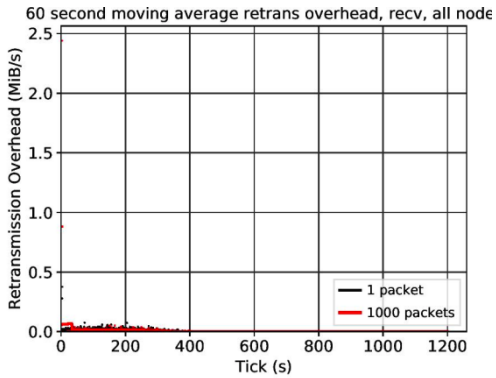


Fig 9. Relationship between Retransmission Overhead and Ticks for download.

In Fig. 9 Retransmission Overhead for downloading of data is negligible. Download time required for downloading 100 KB data can be shown in Fig. 10. Here we can see that most downloads took place between 0.4 seconds and 0.45 seconds. Minimum reading is 0.1 seconds whereas maximum reading is till 1.3 seconds.

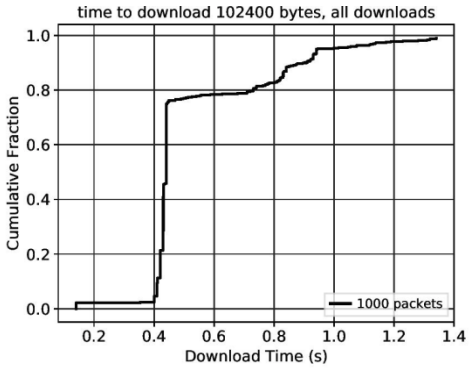


Fig 10. Time to download 100 KB.

Fig 15 shows time taken to download first byte or we can call it Time to First Byte(TTFB).

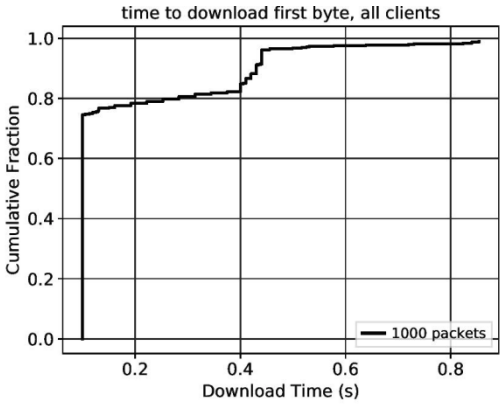


Fig 11. Time to First Byte of all nodes.

TTFB is mostly 0.7 seconds for most of the nodes but also increases till 0.5 seconds. Maximum reading goes till 0.9 seconds.

In the Simulation output for download it is clear that there is no impact of tcp window for nonbulk operation. We can conclude that this is similar to normal Tor Browsing as shown in Fig 3. Now at this step the downloads and other necessary data got from Browsing are stored in Remote Terminal. Here the data stored has to be brought back to Local Host. Hence in simulation we upload 100 MB of data from Remote to Host. As user is no longer browsing the data is uploaded in bulk fashion. There are no pauses taken during upload. This is the extra time or latency introduced in our model as this part would not exist in normal Tor browsing as shown in Fig. 3.

Fig. 12 shows relationship between throughput and ticks for downloading data from Remote to host.

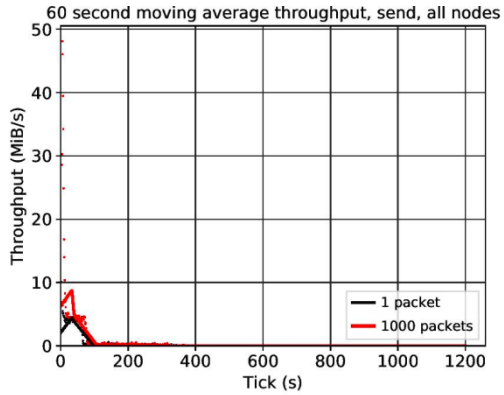


Fig 12. Relationship between Throughput and Ticks for upload.

This shows that we have throughput for upload is around 8 MiB/s for upload when tcp window is 1000 and 5 MiB/s when tcp window is 1. Hence there is impact of tcp window during upload of data.

Fig. 13 shows relationship between Goodput and Ticks for uploading of data.

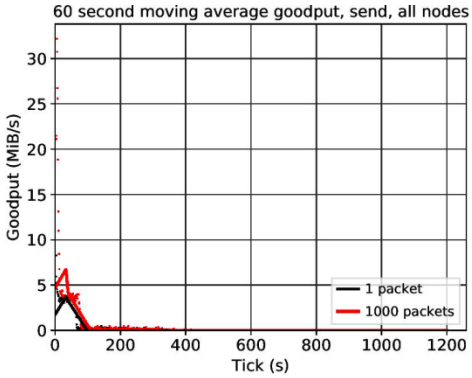
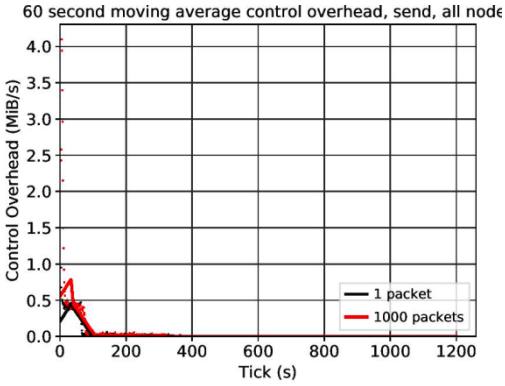


Fig 13. Relationship between Goodput and Ticks for upload.

Here we can see the goodput is around 6 MiB/s for upload for Tcp window of 1000 and 4 MiB/s for Tcp window of 1. This shows that TCP Window has Significant impact when it comes to Bulk upload. Fig. 13 shows relationship between goodput and ticks for uploading data from Remote to host. Fig. 14 shows Control Overhead for uploading of data and relationship with Ticks.



.

Fig 14. Relationship between Control Overhead and Ticks for upload.

In Fig. 14 Control Overhead for uploading of data is 0.8 MiB/s for tcp window 1000 and 0.4 MiB/s for Tcp Window 1. In Fig. 15 we can see the Retransmission overhead for uploading data.

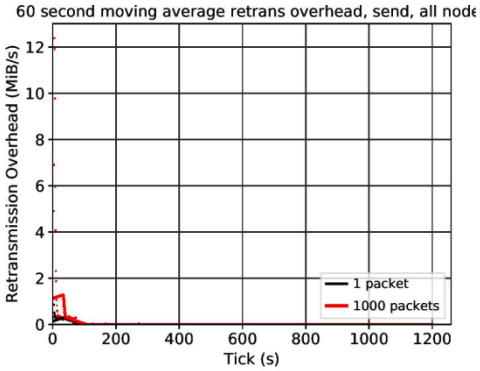


Fig 15. Relationship between Retransmission Overhead and Ticks for upload.

Fig. 15 we can see the Retransmission overhead for uploading data is 1 MiB/s for upload for tcp window 1000 and 0.3 MiB/s for tcp window 1.

We can see here that time taken for upload is 100s. This is the extra time that our model needs which normal Tor Browsing would not need. This is added delay which is tradeoff for extra anonymity.

V. CONCLUSION

Tor is the most used network when it comes to anonymity. But tor network itself is not secure. Anonymity of the user can be compromised due to multiple reasons. If anonymity of user gets compromised then user's ip address will get noted at the Server interface.We have proposed a model for advanced anonymity in this paper. The model has a Remote Terminal between Local Host and Target website or web server. Due to presence of Remote Terminal in between Webserver and local host even if Tor network is compromised ip address of Local Host is not noted at the Webserver. Only ip address of Remote terminal will be noted on users interface. We have simulated this model in Shadow Simulator in order to get Benchmark Parameters. In Simulation we have three components which are host, server and remote. Here data is downloaded from server to remote and uploaded to host. In simulation results it is seen that downloading of data in a nonbulk fashion took 400s. This time will also be required while downloading directly from Tor Browser. We have found different Benchmark parameters in simulation which include throughput, goodput, Control Overhead and Retransmission overhead. We can also conclude that tcp window doesn't affect Benchmark parameters when we download a data in non bulk fashion. We got total goodput of 0.4 MiB/s for downloading data. As the user will download data while surfing the throughput is less. The extra step in our anonymity model is uploading of data from remote to host. Here uploading of data in bulk fashion took 100s. We got benchmark parameters which includes throughput, goodput, Control Overhead and Retransmission Overhead. This is the delay is tradeoff for extra anonymity in our model. But tcp window affects Benchmark parameters significantly when we upload data in bulk fashion. Here goodput of 6MiB/s for tcp window 1 packet and 4 MiB/s for tcp window 1000 packets.

REFERENCES

1. R. Dingledine, N. Mathewson, and P. Syverson, “Tor : the second-generation onion router,” in Proceedings of the 13th Usenix Security Symposium, August 2014.
2. R. Koch, M. Golling and G. D. Rodosek, "How Anonymous Is the Tor Network? A Long-Term Black-Box Investigation," in Computer , vol. 49, no. 3, pp. 42-49, Mar. 2016.
3. “Did the FBI Pay a University to Attack Tor Users?,” Tor Blog , 11-Nov-2018. [Online]. Available: https://blog.torproject.org/blog/did-fbi-pay-university-attack-tor-users.
4. Etzioni, A. (2014). NSA: National Security vs. Individual Rights. Intelligence and National Security , 30(1), 2017, pp.100-136.
5. K. Bauer, D. Grunwald and D. Sicker, "Predicting Tor path compromise by exit port," 2009 IEEE 28th International Performance Computing and Communications Conference , Scottsdale, AZ, 2009, pp.
6. M. A. Sulaiman and S. Zhioua, "Attacking Tor through Unpopular Ports," 2013 IEEE 33rd International Conference on Distributed Computing Systems Workshops , Philadelphia, PA, 2013, pp. 33-38.
7. M. Khan et al ., "The effect of malicious nodes on Tor security," 2015 International Conference on Applied Research in Computer Science and Engineering (ICAR) , Beirut, 2015, pp. 1-5.
8. S. Dahal, Junghee Lee, Jungmin Kang and Seokjoo Shin, "Analysis on end-to-end node selection probability in Tor network," 2015 International Conference on Information Networking (ICOIN) , Cambodia, 2015, pp. 46-50.
9. T. Sameeh, "Research: Classification of attacks on Tor clients and Tor hidden services", Deep Dot Web , 2019. [Online]. Available: https://www.deepdotweb.com/2019/02/20/research-classification-of-attacks-on-tor-clients-and-tor-hidden-services/. [Accessed: 26- Feb- 2019].
10. Vanbever, Laurent & Li, Oscar & Rexford, Jennifer & Mittal, Prateek. (2014). Anonymity on QuickSand: Using BGP to compromise tor. Proceedings of the 13th ACM Workshop on Hot Topics in Networks, HotNets 2014.
11. Sun, Yixin & Edmundson, Anne & Vanbever, Laurent & Li, Oscar & Rexford, Jennifer & Chiang, Mung & Mittal, Prateek. (2015). RAPTOR: Routing Attacks on Privacy in Tor. Proceedings of the 24th USENIX Security Symposium.
12. M. Yang, X. Gu, Z. Ling, C. Yin and J. Luo, "An active de-anonymizing attack against tor web traffic," in Tsinghua Science and Technology , vol. 22, no. 6, pp. 702-713, December 2017.
13. Jansen, Rob, and Nicholas Hooper. *Shadow: Running Tor in a box for accurate and efficient experimentation*. No. TR-11-020. MINNESOTA UNIV MINNEAPOLIS DEPT OF COMPUTER SCIENCE AND ENGINEERING, 2011