TOR AND ONION SERVICES

(Rushit Shah)

Attacks can be divided into seven categories according to their strategy and objective -

* Correlation Attacks End-to-end - Passive Attack
* Congestion Attacks End-to-end - Active Attack
* Timing Attacks End-to-end - Active Attack
* Fingerprinting Attacks Single-end - Passive Attack [14]
* Denial of Service Attacks Single-end - Active Attack
* Supportive Attacks
* Revealing Hidden Services Attacks

## Correlation attacks

The correlation attack uses the packet timing to link the network flows. The packet flows, packet sizes and its timing help to correlate the network flows which can help in breaking the anonymity in anonymous communication. As a result, the attacker can conduct attacks over TOR as he finds the correlation between the entry and exit nodes to confirm that a communication is taking place between the client and the server.

[28] helps to clearly understand the way in which attackers use the flow correlation method. If we consider a Tor network with M ingress rows and N egress rows, the relation between them cannot be determined because of the use of encryption and onion routing. This helps a user to stay anonymous and carry out its activities. But to de-anonymize the network’s activity, attacker inspects the packet contents and identifies the associated flow pairs. This is done by comparing the characteristics of the traffic. After linking the associated network flows, the adversary can surpass the anonymity and exploit the network.

A statistical correlation metrics is used currently for the flow correlation attacks. Previously, the correlation metrics that were used by flow correlation algorithms are-

* 1. Mutual information

This metric works on comparing the dependency of two random variables. For instance, to use this metrics, the traffic features of an egress Tor flow are considered which is dependent on its corresponding ingress flow. The traffic features of target flows is studied which is later reconstructed and compared. To make a reliable decision and outcome by following this method, it requires a long vector of features.

* 1. Pearson Correlation

One of the advantages of this metric as compared to the previous one is that it does not require an empirical distribution of variables. Hence, the metric can be worked with a shorter length of data. The Pearson correlation is a linear correlation between the random variables.

* 1. Cosine Similarity

In this metric, the angular similarity of two random variables is measured. In terms of the requirement of data, the cosine similarity is like Pearson correlation as it does not require a creation of empirical distribution of variables and can instead be directly applied on two random variables.

* 1. Spearman Correlation

This metric measures statistical dependence between the rankings of two variables.

The main aim of the attacker is to increase the fraction of Tor connections which is being intercepted. There are two popular ways which are used by the attacker to get the desired result. One is to run many Tor relays. When these relays are run on the network, the traffic features of the Tor connection are recorded. If the adversary has the access to malicious relays, it increases the chance of intercepting both the ends in a Tor connection. The other method is by taking into account the autonomous systems or internet exchange points. A specific number of autonomous systems and IXPs intercept a significant fraction of Tor traffic. This helps in performing the correlation attack on Tor.

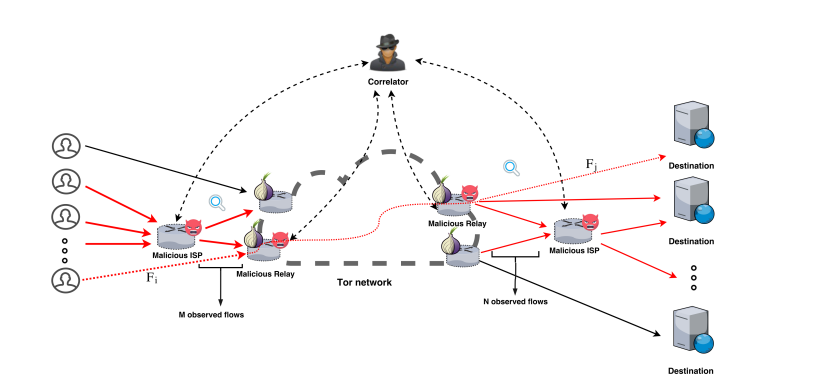


Fig 2: The flow correlation attack setting on Tor [28]

* HTTP Based Application-Level Attack-

The attack is based on the problem of low latency applications based on TCP streams and not on specific web browsing on Tor. The assumption of this attack is that the attacker can control multiple routers as well as the entry and exit routers of the circuit. The following is possible because the tor operates in a voluntary manner. The HTTP’s vulnerability is used for man-in-the -middle attacks.

The client has a greater chance to select their entry and exit routers in the circuit as the resource claims of their routers are exaggerated. A forged web page or a targeted web page attack can then be carried out successfully if the client issues a HTTP request. The overall idea is to let the client’s browser initiate malicious connections. This helps in creating a distinctive traffic pattern. The client’s identity is exposed by the detection of the entry router. The entry router is not required to carry out the attack if the adversary is capable of sniffing the packets that are transmitted via the link between the client and the entry router.

To avoid this attack, it is necessary to minimize the chance of choosing malicious routers in the circuit. This can be done by increasing the total number of Tor routers. The circuit construction algorithm is also evolved to select only fully trusted and dedicated routers through strict authorization and authentication. Additional ways to avoid the attacks is by using HTTPS and web browser plugins.

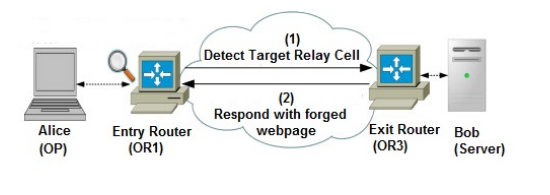


Fig 3: Forged webpage injection attack [29]

* Bad Apple Attack-

The Bad Apple Attack is a kind of application-level attack that requires the installation of a malicious program on the client's computer in order to obtain the client's IP address and use Tor for communication. Since it is not Tor-specific, the method used to retrieve the IP address is not covered in depth in the paper. After the malicious application has the IP address, it can use the Tor network to relay it to a malicious server. The attacker has to have access to the client's circuits' exit nodes in order to be able to correlate traffic from the malicious application with other traffic. The exit node can correlate communication from the malicious application with other network traffic because Tor combines numerous streams, possibly from various apps, in a single circuit. For instance, the attacker can link the client to a specific website if the malicious exit node first detects traffic from the malicious program and then detects traffic from that website on the same circuit.

* Replay Attack-

The replay attack is carried out by the attacker by creating a duplicate cell by selecting a cell in the entry node. This cell is then sent to the second node in the circuit. Once the circuit has been established, the duplicated cell is selected, making it a relay cell. The encryption of Tor layer with the Advanced Encryption Standard can be decrypted by using these relay cells. The cells are detected at the exit nodes and notifies the attacker about the entry nodes. The decryption error takes place as the decryption and encryption counter goes out of sync by duplicated cells. The attack is confirmed by the adversary by matching the timing of error generated and duplicated cells sent. Once confirmed, the communication between the client and server can be accessed by the attacker.

* Cell based Counter Attack-

The cell-based counterattack works on the method of manipulating the timing of relay cells and the exit and entry counter cells. The attacker obtains the right to embed a signal in the client or server traffic. With the help of this attack, it can be confirmed that there lies a communication between the client and the server and further attacks can thus be implemented by the adversary. The attacker needs to be really careful while sending each symbol and needs to calculate the timings accordingly because, if the timing is not appropriate, the attacker can lose the connection between the client and the server. If the timing for the symbols to be sent is too short, the cells can get combined by other relays in the circuit. On the other hand, if attacker waits too long to send the symbols, it increases the latency which might force the user to create a new circuit. Along with this, the attacker also needs to choose the number of cells which needs to be sent in such a way that the combined cells can be recognized as the symbols. This is necessary because natural congestion or any delay that is caused in the network, results in the insertion of cell pattern at the middle onion routers.

## Congestion Attacks

The congestion attack focuses on monitoring the connection between two nodes as well as creating new paths through other nodes through which all the available capacity is consumed. When the attacker clogs one of the target paths, a change in the observed speed of the victim’s connection is observed.

An attack carried out in 2005, with the help of congestion as well as timing analysis, was able to reveal all the routers that were involved in the Tor circuit. This was carried out by measuring the load of each node in the network and then congesting the nodes. By this, they were able to discover which nodes were responsible for the participation in a particular circuit.

[1] conducted the congestion attack on Tor. To carry out this attack, three features were necessary. First, when routing requests, tor routers do not introduce any artificial delays. This makes it easy for an adversary to observe changes that take place in request latency. Second, the Tor router addresses are officially known and thereby easily obtained from the directory servers. Third, the Tor server implementation that was used during this attack did not restrict the arbitrary length parts.

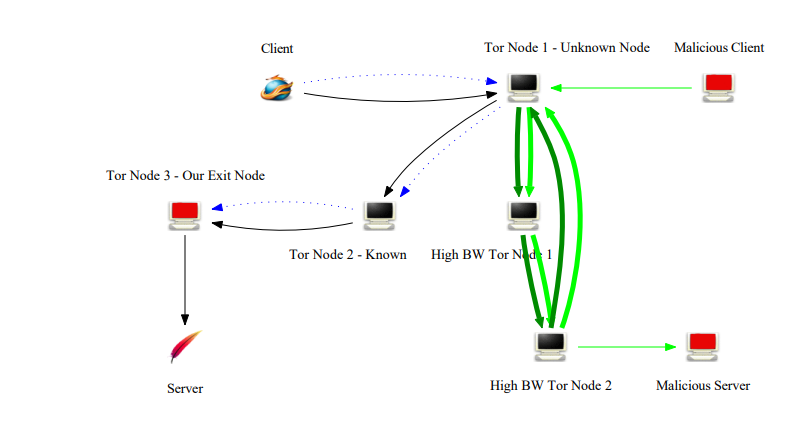


Fig 4: Congestion attack setup [25]

The above figure provides details about the steps of the attack that was conducted. The adversary ensures that a repeated request performance is carried out by the initiator at known intervals. Then, the adversary observes the patterns of arrival times for these requests. Finally, the adversary performs the clogging attack, thereby changing the pattern. With the help of this attack, the adversary determines the entry node.

To carry out the congestion attack, the attacker uses the method of looping back to the circuit. For this, the attacker creates a long circuit which consists of the inclusion of that router repeatedly on the path. The router will have no idea that the circuit is looped back to itself. As a result, during the time of packet scheduling, Tor will treat this long circuit attack as many different circuits. The attacker then carries out the attack and transmits the data when the circuit is long enough. The main hindrance for the attacker is time. For an attack to be carried out successfully, it is necessary to have a sufficient long circuit length. The length of the circuit created also depends upon the delay and bandwidth because if the delay is large and the bandwidth is small, it will take a longer time for the attacker to create the circuit which is of sufficient length to serve the purpose. To check if the victim’s circuit latency depends on the attack, the attacker can vary the strength of the attack. However, the attacker should not make the congestion attack too powerful especially for the targets which have low bandwidth as the routers in this case will be overwhelmed by the amount of traffic it is encountered resulting in the routers being knocked out. If the routers stop working, the attacker will stop receiving any requests.

* Attack by modulating traffic-

The modulating traffic attack consists of a corrupt server and one or more corrupted Onion Routers. To execute this attack, the attacker needs to make the client connect with the corrupted server. The attacker only has the partial view of the Tor network so in order to start the attack, the adversary uses unencrypted HTTP traffic. The corrupted ORs are then connected with legitimate ORs. These corrupted ORs fill the connection with probe traffic to check if they are on a path from target client to corrupt server and record the latency of the connection. If the latency pattern matches the pattern that the corrupt server sends to the client, it is assumed that there are chances of the OR to be a part of the circuit from the client to the corrupt server. This technique is utilized to discover all the ORs that are on the path from client to server. This attack is also used to check if the origination of two different circuits arises from the same client. This attack works better when the length of circuits is greater or a larger pool of ORs are used in the Tor network.

## Timing Attacks

This attack employs traffic analysis techniques to connect otherwise unrelated data streams back to the initiator. The attacker can connect to specific Tor nodes and measure message latencies. The attacker can analyze the data using traffic-analysis techniques by estimating the traffic load of a Tor node against known traffic patterns. A malicious server sends data to the victim in a pattern, which is then observed by creating a connection through candidate onion routers and performing traffic analysis in a variant of this attack.

* A cell counter-based attack against Tor-

In a normal case of the attack, the attacker requires both ends of the circuit to carry out this attack, but a variation of this attack allows the attacker to carry out this attack by just knowing the exit onion router. In this case, the attacker carries out a man in the middle attack.

The attack works by hiding a secret signal in the traffic's cell counter, which is then recognized by another malicious node in the network to confirm the communicating parties. The secret signal could be a bit sequence. The signal injection can be performed on either the exit or the entry onion router. The injection operates by varying the number of queued relay cells on each onion router. For example, the attacker could specify three relay cells for '1' and one relay cell for '0'. Some variation may occur during the transfer of these cells from one onion router to another due to network congestion and latencies.

* Browser based attacks-

It describes a time-based attack that takes advantage of web browser behavior when it interacts with tampered HTTP traffic. This attack requires two malicious onion routers: an entry onion router and an exit onion router. The entry onion router looks for time-based patterns in the user's traffic, whereas the exit onion router modifies the HTTP traffic to include HTML or JavaScript code that generates calls to a malicious web server in recognizable time patterns.

This attack does not require the malicious entry and exit onion routers to be in the same circuit. If a user leaves their browser open after receiving code from a malicious exit onion router, the attacker will have control of an entry onion router in a new circuit created by the onion proxy at some point in the future. This allows the attacker to connect the user's new circuit activities to their previous activities on the compromised circuit, potentially jeopardizing their anonymity.

* Indirect Rate Reduction Attack-

The main motive of this attack is to extract the information about the clients which are communicating with a predefined server through the Tor network. To carry out this attack, the communication with the entry node of the circuit of the clients has to be intercepted. Therefore, the clients chosen by the adversary needs to be monitored beforehand by the attacker. The attacker uses a technique of congestion control behavior of TCP. The congestion control behavior states that the congestion window of the device scales down on receiving three same ACK packets. The attacker identifies the exit nodes that have the chances to connect to the server as well as lot of different ports. These ports receive three packets by the adversary. The packets have wrong sequence number, but the IP address of these packets are spoofed so it seems that they have come from the server. The exit node will send three ACK packets which will scale down the congestion window of the server.

The adversary must make sure that enough time is given to the server congestion window to recover between the iterations of attack. It is possible for the adversary to clearly determine the clients that are communicating with the server after carrying out the attack multiple times. The adversary has to wait several minutes before trying to repeat the attack by sending fake packets as some time is required for the execution of iterations that are sufficient enough for the connections to last long.

## Fingerprinting Attacks on Tor

A website fingerprinting attack can identify a website even if its traffic is encrypted using Tor or a VPN. This is accomplished by analyzing packet information such as packet length, packet count, and timing. A website can be identified using this information without having to inspect the encrypted payload.

In Tor, there are two ways to collect traffic data. The first method entails an attacker establishing an entry node and capturing traffic that passes through it. However, because Tor chooses nodes at random, a victim is unlikely to connect to the attacker's node. The second method involves a network operator, such as an ISP, as the attacker. This attacker can intercept traffic packets between a victim and a Tor entry node.

* Closed and Open world tests-

Two different schemes can be used to evaluate fingerprinting attacks. The first type of test is a closed-world test in which the victim can only access a limited number of websites that the attacker attempts to detect. For example, the attacker could choose 100 websites and research their distinguishing features. In this scheme, the victim can only access these 100 websites, and the attacker wants to know which ones they are.

The second method of evaluation is an open-world test. In this case, the victim can access any website on the Internet, and the attacker must determine whether the website is one of the monitored sites. If the website is monitored, the attacker must also determine which of the 100 monitored websites it is. This scheme is more difficult to implement because the attacker must identify a potentially unknown website among a large number of unmonitored sites.

* Website fingerprinting-

In this attack, the adversary determines the size of the file which is being downloaded by counting the total size of packets on each port. When a user visits a webpage, the information is displayed by the browser by sending a request to download different files. These files are downloaded through a separate TCP connection using different ports. Each webpage requires a different file, and every file is of different size. The information related to size of file creates a unique fingerprint that is used to identify the webpage. To extract the information, the attacker builds a collection of fingerprints to monitor the user’s browsing behavior. This collection is built by comparing recorded fingerprints against their collection. The attackers use the data related to the number of incoming and outgoing packets to estimate the file size as it is difficult for them to detect the precise size due to the fixed data cells of 512 bytes which is used by Tor. The fingerprint is represented by a vector that counts the occurrences of subsequent incoming packets.

## DoS attacks on Tor

The attacker carries out a DOS attack on TOR by flooding it with traffic or requests which prevents legitimate users from accessing the tor network.

Various research papers have conducted experiments to understand the DoS attacks which are taken place on the Tor and the repercussions to it.

Earlier during the lack of upper bound on the length of tor circuits in older tor enabled the attacker to perform the bandwidth amplification DoS attack. This attack was carried out by creating cyclic, arbitrary length circuits through high bandwidth tor relays. This led to a congestion which had an effect on the latency of legitimate circuits which can be used to determine the guard relay on circuit.

A DoS attack was carried out by asymmetric, amplification packet-spinning. The goal was to keep the legitimate relays busy with expensive cryptographic operations which results in the legitimate clients choosing attacker’s relays. To carry out this attack, the attacker makes use of malicious relays to create a circular Tor circuit that starts and ends at a malicious relay.

A selective DoS attack can take place by the attacker which increases the probability to choose the attacker’s relays as guard and exit relays. This helps the attacker to de-anonymize a large fraction of Tor circuits.

Tor's end-to-end reliable data transport can be exploited through which the memory is consumed by filling up the application layer packet queues by implementing the sniper attack which is a memory-based DoS attack. By this method, the attacker is capable to sequentially disable 20 exit relays in a span of 29 minutes, making the Tor network unusable while remaining undetected.

From an attacker’s perspective, the goal is to either disrupt the tor network entirely or disrupt a portion of it that affects the entire subpopulation of Tor users. To disrupt a portion of Tor, the attacks are carried out against the bridge infrastructure and the set of unpublished relays of Tor that permits the users to participate who are otherwise prevented to access the Tor network directly. An attack is considered successful if it entirely prevents the users to access Tor or if the performance is degraded to such an extent that the anonymity service becomes too burdensome to use. In a current scenario, even the degrading performance is seen as an impediment as the delay in the performance as compared to the current levels can lead to a lot of customers to abandon the network.

* Unbalancing Load-

A bandwidth DoS attacks can be used over TorFlow to disrupt Tor services. The TorFlow is used to scan Tor relays to measure their relative performance. The relay weights which are produced by TorFlow can be disrupted by an adversary. The attacker launches a bandwidth DoS attack that clogs the TorFlow scanner’s link. The TorFlow scanners are identified by the adversary through their IP address. Once the scanner’s link is clogged, the latency and packet loss increase. As a result, the scanners take more time to download files through Tor relays. The scanner will then term the performance of relay as bad thereby reducing the accuracy generated for the relay weights. This process leads to a disruption in process of loading balance.

One of the easiest methods used by the adversaries to carry out this attack is to brute force the TorFlow scanner by flooding it with bandwidth at constant rate.

To increase the impact on the final set of relay weights generated, the attacker can use another method where a certain amount of bandwidth is used to flood the victim and later pausing the attack for some time. This process is repeated simultaneously which will produce an output where the scanner records normal download times for some relays and reduced times for other set of relays when attack is paused.

The adversary also uses another set of technique by determining the best set of performing relays and targeting them at a specific set of time. The adversary measures the fastest relays and targets them with bandwidth DoS attacks which creates a greater impact in performance.

## Supportive Attacks

The attacks here do not aim to directly de-anonymize the Tor users but are helpful in a way to carry out the attack of de-anonymization at a later point of time.

* Sybil Attack-

The number of active Tor relays, which are nodes that assist in directing internet traffic through the Tor network to safeguard users' privacy and anonymity, suddenly increased in June 2010. Later, it was found that someone had installed a large number of Tor relays on PlanetLab equipment. Although it may appear innocent, this can be exploited to attack the Tor network. The attacker can enhance their consensus weight and potentially gain control of a sizable chunk of the network by setting up a lot of relays. They can now monitor and intercept traffic, jeopardizing the security and privacy of Tor users.

A Sybil attack is when an attacker generates numerous virtual identities or nodes to control a network more effectively than they could with just one. The success of many attacks against Tor depends on the volume of traffic an attacker can see, also referred to as their consensus weight. An attacker's consensus weight increases with the number of nodes they control, making it simpler for them to carry out attacks like fingerprinting and correlation attacks. In essence, a Sybil attack facilitates information gathering and compromises user security and privacy on the network.

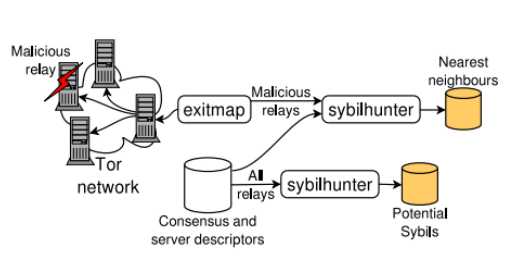


Fig 5 : Sybil Attack setup [29]

The Sybil attack not only facilitates other attacks but also jeopardizes users' access to and privacy within the Tor network. The efficiency of the Tor network depends on the relays' dependability because failing relays can decrease user experience and limit anonymity. If users experience problems brought on by faulty relays, they may stop using Tor completely, which would reduce the number of users and the level of anonymity. By purposefully affecting the dependability of anonymous communications and raising the possibility of user anonymity being compromised, attackers can exploit this by installing their own malicious relays. The Tor network's functionality and security are thus seriously compromised by the attack.

* Packet Size Analysis Attack-

A technique was published in 2011 that distinguished the Tor traffic from non-Tor traffic just by intercepting and analyzing the traffic passively. On examination of this Tor traffic, researchers found out that the third packet transmitted is around 140 bytes and the size of 5th packet is around 920 bytes. With the help of these simple patterns, it is possible to classify around 98% of the actual Tor traffic. Additionally, many packet size are larger than 512 bytes which is the size of Tor cell. This experiment was conducted in a controlled condition but can also be used in the real world by the attackers to exploit the Tor network. Just by analyzing the traffic it was possible for the adversary to differentiate between Tor and non-Tor traffic.

* Tor Authentication Protocol

A paper outlining the attack on Tor Authentication Protocol (TAP) was published by a researcher named Y Zhang in 2009. The attack was carried out when a user runs multiple concurrent sessions of TAP. The TAP plays a crucial role in Tor’s security as it negotiates the session keys between user and onion routers (ORs) in a circuit. The vulnerability takes place when the OR A is malicious and the user connects to it and negotiates a session key, then negotiates a session key with OR B which is non-malicious. The attacker can now interleave the messages from both the sessions to create a different session key between user and OR B. This attack does not directly harm the user, but it goes against the original purpose of TAP protocol.

## Revealing Hidden Services Attack

* First Node Attack

When the attacker’s relay tries to connect to the hidden service’s server, it would immediately reveal the location of the hidden service to that node.

To become the first node in the circuit from a hidden server, there is a requirement of a malicious node and a client by the attacker that connects to the hidden service. A timing pattern is sent by the client in the communication. If the malicious node is on the circuit, it can detect the timing pattern. By knowing the IP addresses of all nodes until the rendezvous point, the malicious node can determine if it is present on the circuit or not. The timing analysis can reveal the location of a malicious node if it is present in the first or second node after the hidden service. If a malicious node is not in the correct position, the attack is carried out again until the malicious node becomes the circuit’s first node. The anonymity of the hidden service is compromised and can reveal its location with this attack.

* Clock Skew Attack

Another way of identifying the location of hidden services is by clock skew attack. In this attack, a list of potential servers is selected. By repeatedly making requests to the hidden servers, the temperature of the server increases which results in its clock to turn at a slightly different rate. The timestamps which are received by the hidden service can be analyzed and its clock skew can be determined. By comparing this clock skew with the skew of the candidate servers, it is possible to locate the hidden service.

## Entry and exit onion router selection attack

* Compromising Anonymity using Packet Spinning

The following attack uses looping circuits and malicious onion routers. It describes a Tor anonymity attack that makes use of looping circuits and malicious onion routers. The attack creates circuit loops to prevent other onion routers from being selected, resulting in a denial-of-service attack. This increases the likelihood of malicious onion routers being selected in circuits, allowing for additional attacks. The attack is based on the assumption that circular circuits are not detectable and that legitimate onion routers spend time performing cryptographic calculations.

* Low resource routing attacks

This attack consists of an adversary which exploits flaws in the routing mechanisms of the Tor network to compromise the anonymity of users with limited resources. These attacks typically involve supplying false resource information to directories or exploiting the victim's circuit creation process in order to identify patterns in the network's routing algorithm. The goal is to increase the likelihood of the adversary being chosen as the entry and exit nodes in the user's circuit, thereby jeopardizing their anonymity. These attacks are especially dangerous for users with limited resources because they may not have access to the network's more secure options.