Network Security

Professor Gaurav S. Kasbekar
Department of Electrical Engineering
Indian Institute of Technology, Bombay

Week - 10

Lecture - 59

Tor: The Onion Router: Part 1

Hello, in this lecture and the next lecture, we'll discuss anonymous connections. What is meant by an anonymous connection? Suppose a user, Alice, wants to connect with the user Bob over the Internet. But these users, Alice and Bob, don't want other users to know who is communicating with whom. That's an anonymous connection.

A popular method for achieving anonymous connections is Tor: The Onion Router. We'll discuss Tor: The Onion Router in this lecture and the next lecture. To illustrate anonymous connections, let's start with an example. Suppose Alice wants to visit a controversial website, possibly a political activist website. Also, Alice wants to achieve the following three objectives.

One is that Alice does not want to reveal her IP address to the website. That is, because if Alice's IP address is known to the website, then the website might try to contact her, which Alice may not want. So, Alice does not want to reveal her IP address to the website. Alice does not want her local ISP—for example, her home or office ISP—to know that she is visiting the website. That's because it's a controversial website, and she does not want others to know that she is visiting it.

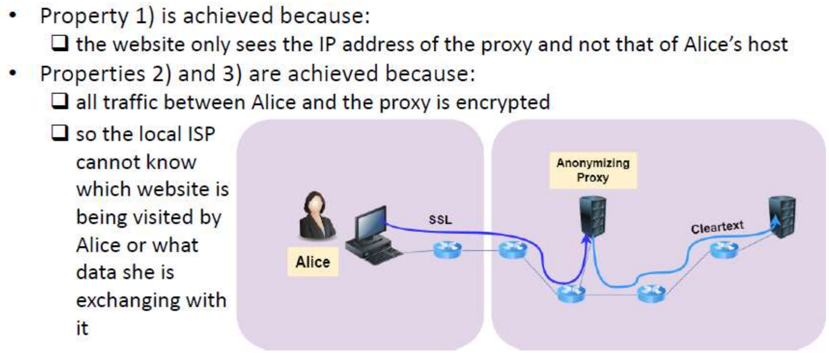
Also, she does not want her local ISP to see the data that she is exchanging with the website. So, the information that she exchanges with the website should be confidential. Others should not know what information she is exchanging. So, how can Alice achieve such a connection to the controversial website? Let's try to use the mechanisms that we have learned to achieve such a connection.

The most basic possibility is that Alice just connects using an ordinary TCP connection without TLS to the website. Does this achieve any of these three objectives? So, Alice does not achieve any of the three required properties. First, consider packets that are sent from

Alice to the website. The source IP address is present in the packets that are sent from Alice to the website.

Hence, the IP address of Alice is revealed to the website. So, the first property is not achieved. If a local ISP observes packets that are flowing from Alice to the website, then by looking at the source IP address and the destination IP address in the packets, the local ISP will come to know that Alice is visiting that particular website. So, this property 2 is not achieved.

Since TCP does not encrypt data, the local ISP can see the data that she is exchanging with the website. So, the third property is also not achieved. Now, if Alice connects with the website using TLS, in that case, Alice's data is encrypted and the data from the website is also encrypted. So, property 3 is achieved. The local ISP cannot see the data that Alice is exchanging with the website.



- Property 1) is achieved because:
  - ❑ the website only sees the IP address of the proxy and not that of Alice's host
- Properties 2) and 3) are achieved because:
  - ❑ all traffic between Alice and the proxy is encrypted
  - ❑ so the local ISP cannot know which website is being visited by Alice or what data she is exchanging with it

Hence, property 3 is achieved. But you can easily check that properties 1 and 2 are not achieved, even if Alice connects using TLS. So, her source IP address is presented to the website in every packet she sends. So, Alice's IP address is revealed to the website, and the destination address of every packet that she sends can be sniffed by the local ISP. Hence, property 2 is also not achieved.

So, TLS also does not help us achieve these three objectives. Hence, we need an alternative approach. So, one possibility is that Alice can use a combination of a trusted proxy server and TLS. So, that is shown in this picture. Alice establishes a TLS connection with a host known as an anonymizing proxy.

This is an ordinary TLS connection between Alice and the anonymizing proxy. This is the controversial website that Alice wants to visit. But Alice does not directly establish a connection with the controversial website. Instead, she establishes a TLS connection with a third party known as an anonymizing proxy. This third party extracts Alice's packets and sends them in clear text form to the political activist website.

So, this anonymizing proxy acts as a relay between Alice and the website. The proxy does not disclose Alice's IP address to this website. So that way this website does not know Alice's source IP address. And it also achieves the other two properties. Alice first makes a TLS connection to this trusted proxy.

That's a third party. And she then sends into this TLS connection an HTTP request for a web page at the desired website. So, when this proxy receives the TLS encrypted HTTP request, it decrypts the request and forwards the clear text HTTP request to the website. So that transmission from the proxy to the website is shown here by this flow. The website then responds to the proxy that is shown by this flow and the proxy then forwards the response of the website to Alice over TLS.

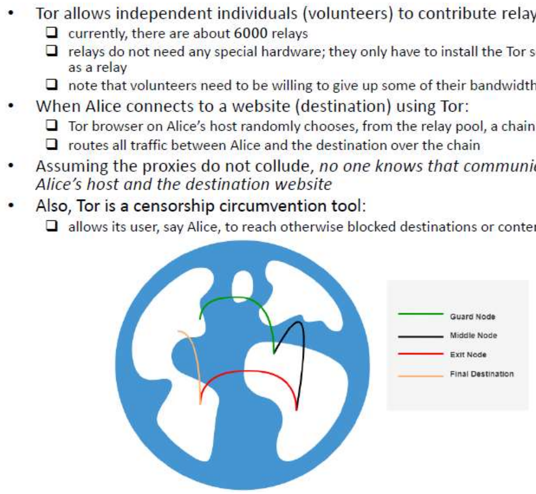
So, property 1 is achieved because the website only sees the IP address of the proxy. Hence, the website does not know Alice's IP address. Hence, property 1 is achieved. Properties 2 and 3 are achieved because all traffic between Alice and the proxy is encrypted. Notice that in the encrypted packets, Alice informs the proxy that she wants to connect with this particular website.

Hence, an ISP who is intercepting packets sent by Alice does not come to know that Alice is connecting with this particular website. The local ISP cannot know which website is being visited by Alice or what data she is exchanging with the website. That's because all traffic between Alice and the proxy is encrypted. So, this solution achieves all three properties that we required. Several companies make such proxy services available.

For example, "proxify.com" is a service that provides such a proxy service. It provides an anonymizing proxy service. But there's a disadvantage of this solution, that is, the proxy knows Alice's IP address as well as the IP address of the website she's visiting and also the proxy can see all the traffic being exchanged by Alice and the website. Hence, this solution fails if the proxy is dishonest. So, there is a single point of failure.

If this proxy knows everything and if the proxy turns out to be dishonest, then this solution fails. So, we want a more robust solution in which, if a single party turns out to be dishonest, the solution should still not fail. So, we want a robust solution which does not rely on the legitimacy of a single party. So, there should be multiple parties involved in achieving these three properties. Even if one of them turns out to be dishonest, the solution will not fail.

Tor: The Onion Router is a more robust solution. It routes traffic through three or more relays. So, instead of only one anonymizing proxy, Tor uses three or more relays. So, a Tor connection is shown here. The source of the flow is over here.



- Tor allows independent individuals (volunteers) to contribute relays to its relay pool
  - ❑ currently, there are about 6000 relays
  - ❑ relays do not need any special hardware; they only have to install the Tor software and configure it to act as a relay
  - ❑ note that volunteers need to be willing to give up some of their bandwidth
- When Alice connects to a website (destination) using Tor:
  - ❑ Tor browser on Alice's host randomly chooses, from the relay pool, a chain of three relays
  - ❑ routes all traffic between Alice and the destination over the chain
- Assuming the proxies do not collude, *no one knows that communication took place between Alice's host and the destination website*
- Also, Tor is a censorship circumvention tool:
  - ❑ allows its user, say Alice, to reach otherwise blocked destinations or content

In the previous example, Alice is the source of the flow, who is situated here. And to establish a connection with the website—political activist website, which is here. She does not directly contact the political activist website. Instead, she uses three relays. There can be more relays as well.

This is the first relay, this is the second relay, and this is the third relay. So, Alice establishes a connection with the first relay. That connection is shown here by the green curve. Then, this is the connection between the first relay and the second relay, shown by the black curve. The connection between the second relay and the third relay is shown by the red curve, and finally, this is the connection between the third relay and the political activist website, shown by the yellow curve.

In this way, the packets are forwarded through three relays. So, throughout this discussion, we will assume that three relays are used, but potentially more relays could also be used. For example, 10 relays or 15 relays could also potentially be used. And the more relays there are, the more secure the solution becomes. But the overhead is larger because the traffic has to flow through more relays.

So, what are these relays? Tor allows independent individuals or volunteers like us to contribute relays to its relay pool. Currently, there are about 6,000 relays. So, anyone can contribute their computer to this pool, and then that computer starts acting as a relay. Relays don’t need any special hardware.

They only have to install the Tor software and configure it to act as a relay. So, anyone with a computer can contribute a relay to the pool. So, they just have to install the Tor software. Note that volunteers need to be willing to give up some of their bandwidth because a relay has to accept traffic from the source or from the earlier relay on the path and then forward those packets to the next relay on the path or to the destination. So that consumes bandwidth, hence volunteers need to be willing to give up some of the bandwidth.

When Alice connects to a website, which is the destination using Tor, the Tor browser on Alice’s host randomly chooses from the relay pool of around 6000 relays, a chain of three relays. So, these three relays are randomly chosen. And then the Tor browser on Alice’s host routes all traffic between Alice and the destination over the chain of the three selected relays. Assuming that the proxies do not collude, no one knows that communication took place between Alice’s host and the destination website. Consider these three relays, 1, 2, and 3.

Under the protocol of Tor, this relay should not tell this relay which is the source from which this relay is forwarding packets to this relay. So, this relay should not disclose the identity of this source to this relay. Similarly, this relay should not disclose the identity of this relay to this relay and this relay should not disclose the identity of the destination to this relay and so on. So, each relay should only know which is the next hop and which is the previous hop on the path. It should not know the identities of the other nodes on the path.

But if they collude; suppose this relay and this relay collude, and this relay informs this relay about the identity of the source, and also this relay and this relay also collude and this relay informs the identity of the source to this relay. In the face of such collision, this solution fails. But these relays correspond to contributions from independent volunteers. So, it is unlikely that the different relays will collude. Also Tor is a censorship circumvention tool. That is, it allows its users, say Alice, to reach otherwise blocked destinations or content.

For example, Facebook or YouTube may be blocked in a particular country. In that case, users in that country can still use Tor to reach the blocked content via the use of several intermediate relays, for example, these relays. Consider Alice who wants to establish a connection with Bob. In the first step, Alice's Tor client obtains a list of Tor nodes from a directory server. So, this Dave is a directory server in this example.

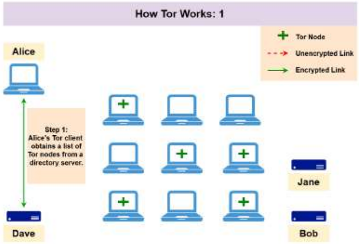
Alice's host contacts Dave and obtains the list of the relays who are available. For example, the list of 6,000 relays will be passed on from Dave to Alice. Then, in the next step, Alice's Tor client picks a random path to the destination server, and this path consists of three intermediate relays. So, Alice uses this particular path in this example. Alice to this relay, this relay to this relay, this relay to this relay and this relay to the destination.

And these green links are encrypted, and the red link is unencrypted. So, Alice establishes a connection with the required destination, Bob, through these three randomly selected relays in this example. And later on, suppose Alice wants to establish a connection with another destination, say Jane. In that case, three relays are again selected randomly, so most likely they'll be different from the relays selected earlier. In this case, the relays happen to be these.

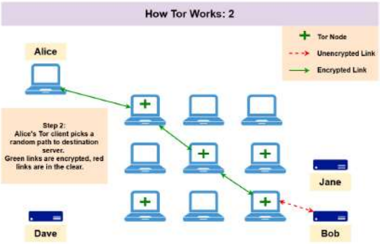
This is the first relay, then this is the second relay, and this is the third relay. So, the traffic is routed through these three selected relays in this example. So, this is how Tor works. Each time a particular source, Alice, wants to communicate with some destination, three relays are selected at random, and traffic is routed through those three relays. So, what are the applications of Tor?

Individuals use Tor to prevent websites from tracking them and their family members since the website does not know the IP address of the individual browsing it. Another example is that individuals use Tor to connect to news sites, political activist sites, sites like YouTube or Facebook, and so on, when these are blocked by local internet providers or the countries they live in. So, these agencies that block connections do not know who is communicating with whom when Tor is used. Hence, individuals can use Tor to connect to such sites. Journalists use Tor to communicate more safely with whistle-blowers and dissidents.

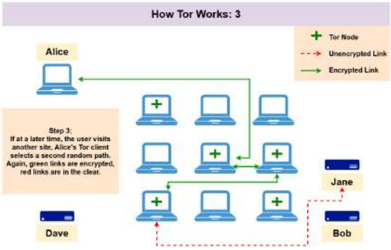
Obviously, journalists don't want any agencies that intercept the connections to know who they are communicating with. Hence, they can use Tor to make their connections anonymous. Non-governmental organizations, or NGOs, use Tor to allow their workers to connect to their home website when they are in a foreign country. They don't notify anyone nearby that they are working with that organization. So, this is another application of Tor.



Step 1:
Alice's Tor client obtains a list of Tor nodes from a directory server.



Step 2:
Alice's Tor client picks a random path to destination server.
Green links are encrypted, red links are in the clear.



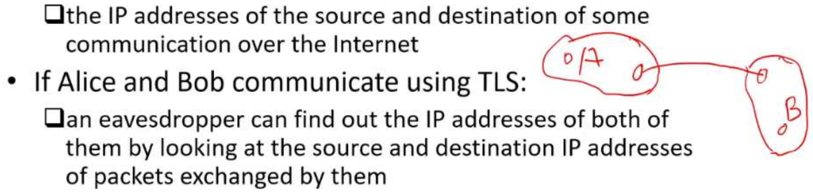
Step 3:
If at a later time, the user visits another site, Alice's Tor client selects a second random path. Again, green links are encrypted, red links are in the clear.

Business executives use Tor to keep their strategies confidential. For example, an investment bank may not want competitors to be able to track what websites their analysts are watching. Again, here the concept of anonymous connections helps. The military uses Tor. For example, consider some military agents operating in some hostile territory.

In that case, insurgents may monitor internet traffic and discover all the hotels and other locations from which people are connecting to known military servers. So, these people are the field agents, the agents who are connecting to their servers. These insurgents should not know who is connecting with the known military servers. So, to make these connections anonymous, they can use Tor. Military field agents deployed away from home use Tor to mask the sites they are visiting.

We can see that Tor has a variety of applications. And in each case, these connections are sensitive connections in which anonymity is required. No one should know who is connecting with whom. We now discuss the limitations of TLS and VPN. So, in all these applications, we need to prevent intruders from finding out the IP addresses of the source and destination of some communication over the internet.

If Alice and Bob communicate using TLS, then an eavesdropper who is intercepting the packets between Alice and Bob can find out the IP addresses of Alice and Bob by looking at the source and destination IP addresses of packets exchanged by them. Hence, TLS does not achieve anonymity. Suppose Alice and Bob communicate using a VPN. In that case, an eavesdropper can still find out that communication is taking place between some user in Alice's network and some user in Bob's network. Recall that the typical way that a VPN is connected is the following.



❑the IP addresses of the source and destination of some communication over the Internet

- If Alice and Bob communicate using TLS:

❑an eavesdropper can find out the IP addresses of both of them by looking at the source and destination IP addresses of packets exchanged by them

There is some organization, possibly a head office, and there is a gateway router in that organization, and there is another organization, and it has its own gateway router. So, a tunnel is created between these gateway routers, and Alice is somewhere here, and this is Bob. So, a tunnel is created between the gateway router of this organization and the gateway router of this organization. So, over this tunnel the traffic is transmitted. So, all the traffic from this organization is aggregated over this tunnel and similarly all the traffic from this organization is aggregated over this tunnel.

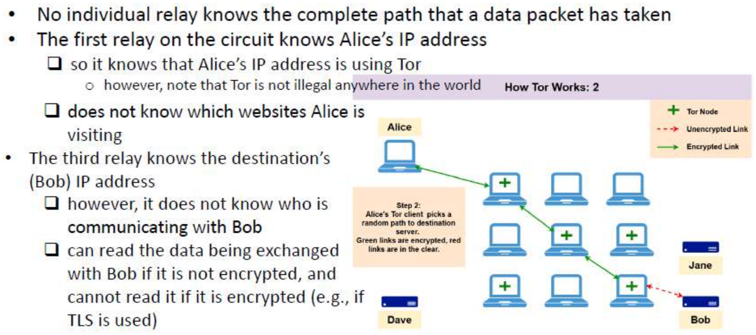
So, an eavesdropper cannot know who exactly in this organization's network is communicating with whom in this organization's network. But they still know that someone here is communicating with someone here. That is, they know that two companies are collaborating. So, in that case, this may not be acceptable. The leakage of the fact that someone in this organization is connecting with someone in this organization.

Hence, it does not provide the required degree of anonymity. So, these are the limitations of TLS and VPN, which are overcome by Tor. We will now discuss the operation of Tor in detail. The objective of Tor is to make it difficult for attackers to find out that a given source is communicating with a given destination. So, to create a private network pathway with Tor, the client's browser incrementally builds a circuit of encrypted connections through relays on the network.

Alice first creates a connection with the first relay on the path, then, with the help of this relay, Alice extends this circuit to the second relay, and then, with the help of this second relay, Alice extends this circuit to the third relay, and finally, using the third relay, Alice extends the connection to the destination, Bob. So, we will see later in detail how the client's browser incrementally builds a circuit of encrypted connections through relays on the network. The circuit is extended one hop at a time, as we just mentioned. Each relay along the way knows only which node gave it data and which node it is giving data to. For example, this relay only knows the identity of this relay and the identity of this relay.

This relay does not know who is the source of the connection and who is the destination of the connection. So, this is a crucial property. Each relay only knows which node gave it data and which node it is giving data to. Similarly, this relay only knows the identity of Alice and the identity of this relay. But this relay does not know the identity of this relay or the identity of Bob.

No individual relay knows the complete path that a data packet has taken. The first relay on the circuit knows Alice's IP address. So, the first relay knows that Alice's IP address is using Tor, but that's okay because Tor is not illegal anywhere in the world. The first relay does not know which websites Alice is visiting, so we have already mentioned that this first relay only knows the identity of Alice and the identity of this relay, but this relay does not know the identities of the destination. The third relay knows the destination's IP address, that is Bob's IP address in this example.



- No individual relay knows the complete path that a data packet has taken
- The first relay on the circuit knows Alice's IP address
  - ❑ so it knows that Alice's IP address is using Tor
    - ○ however, note that Tor is not illegal anywhere in the world
  - ❑ does not know which websites Alice is visiting
- The third relay knows the destination's (Bob) IP address
  - ❑ however, it does not know who is communicating with Bob
  - ❑ can read the data being exchanged with Bob if it is not encrypted, and cannot read it if it is encrypted (e.g., if TLS is used)

How Tor Works: 2
Tor Node
Unencrypted Link
Encrypted Link
Alice
Step 2:
Alice's Tor client picks a random path to destination server.
Green links are encrypted, red links are in the clear.
Jane
Dave
Bob

So, this relay knows the identity of Bob but it does not know who is communicating with Bob. So, the third relay can read the data being exchanged with Bob by Alice if it is not encrypted and cannot read the data if it is encrypted. For example, if TLS is used in this connection. So, if the data is sensitive then it can always be encrypted using a TLS connection. In that case, even this link will be encrypted.

This is an option that is available. If the data is not confidential, it can be sent in the clear over this link. If it is confidential, then even this link can be encrypted using TLS. So, in summary, we discussed the concept of anonymous connections. We discussed an example where anonymity was required, and then we discussed several possible solutions, such as TLS, VPN, and anonymizing proxies. One solution that overcomes many of the limitations of these other options is Tor, and we began our discussion of Tor.

We discussed how Tor uses three intermediate relays to establish connections between a source and a destination. We will continue our discussion of Tor in the next lecture. Thank you.