Appearing in Cyber Criminology, Hamid Jahankhani, ed., Springer 2018, https://link.springer.com/book/10.1007/978-3-319-97181-0

The Dark Web   Prepublication Draft, April 2018

Abstract

The dark web consists of those websites that cannot be accessed except through special anonymizing software, most commonly the Tor package. Web services hidden this way have proved extremely difficult for for authorities to track down. While there are many legitimate hidden sites, the dark web has also attracted a wide range of criminal enterprises, often enabled by the availability of anonymous cryptocurrency payments. While Tor has some theoretical weaknesses, most law-enforcement actions against hidden sites (the Silk Road, Playpen, etc) have succeeded only because of operational mistakes by the sites' administrators.

The **dark web** consists of those websites that cannot be accessed except through special **anonymizing** software. The most popular anonymizing system is Tor, originally an acronym for The Onion Router, but there are others, such as Freenet and I2P (below). While there are legitimate uses of dark websites (the New York Times has one, to allow sources to communicate confidentially), the dark web is perhaps best known for attracting criminal enterprises engaged in the sale of contraband. Products such as stolen credit-card data and child pornography are easily delivered via the Internet, but the dark web has also attracted merchants selling illegal drugs, armaments and other physical items.

Tor dark-web addresses end in the suffix ".onion", *eg* nytimes3xbfgragh.onion or facebookcorewwwi.onion. The challenge of anonymizing software is to figure out how to deliver traffic to such public addresses without allowing anyone to trace the traffic. Tor was designed to achieve anonymity for users and servers even from government-level attempts at unmasking. In the past two decades governments have gotten much better at monitoring the Internet; see the attacks outlined in "Traffic Correlation" below. However, most if not all hidden-site discoveries to date have relied on operational errors rather than any fundamental weaknesses in the Tor protocol.

All Internet traffic is transmitted via chunks of data called **packets** that are delivered to an attached **IP address**. Given a server IP address, the approximate location of the server is easy to discover using standard networking software; the exact location is straightforward for authorities to obtain. An immediate consequence is that a public dark-web address can never be associated with the site's IP address.

Virtual Private Networks

As a first step in describing anonymization, we consider VPNs. These provide limited anonymity for users, but none for servers. Suppose user Alice wishes to access server Bob, but not let Bob know it was her. To achieve this, she contracts with VPN provider Victoria. Alice prepares a packet addressed to Bob, and attaches to it an additional header sending the packet to Victoria. Victoria removes this additional header, rewrites the sender address to refer to Victoria rather than Alice, and sends it on to Bob. As far as Bob can tell, the packet came from Victoria; there is no evidence of Alice's participation.

Bob then sends the reply back to Victoria, which recognizes from context (specifically, from the "port numbers" associated with Alice's Internet TCP connection) that it must be forwarded on to Alice. Victoria does so, and Alice receives Bob's reply.

Nothing in this scheme provides any anonymity for Bob, whose real IP address must be available to Alice at the start. However, nothing in the packets seen by Bob identifies Alice. Alice is thus able to browse Bob's website anonymously.

Alice's identity can be easily unmasked by the authorities, however. Victoria's IP address was seen by Bob, so Victoria can be identified. If the authorities now show up at Victoria's with a subpoena, there is a good chance they will find log records showing that customer Alice, identified by her IP address if nothing else, had Victoria send packets on to Bob. If Victoria has kept no records, then Alice's original interaction with Bob is untraceable. However, the authorities can likely compel Victoria to record future connections to Bob; if Alice tries again, she is revealed.

Alice's identity can also be discovered by monitoring traffic at Victoria (perhaps from the vantage point of Victoria's Internet Service Provider), and looking for correlations between arriving and departing packets. If every packet arriving from Alice is followed by a packet sent on to Bob, and vice-versa, then Alice is unmasked. This approach has the advantage that Victoria need not be involved or even notified.

In the current-day Internet, some VPNs pride themselves on the anonymity they provide for their customers. Some advertise not keeping any logs, or at least not logging per-connection information. Some accept anonymous payment in cryptocurrencies such as Bitcoin. Some locate servers in jurisdictions outside the United States and Europe.

The Tor approach

The strategy used by Tor can be loosely described as a three-stage VPN, with the added element of encryption to prevent the VPN stages from learning more than the minimum necessary about one another. The VPN stages are known as **Tor nodes**, described below. The basic three-stage approach conceals only the user, not the server, but a variation allows anonymity for both endpoints.

The ideas behind Tor, and in particular the concept of “onion routing”, were developed by Paul Syverson, David Goldschlag and Michael Reed at the US Naval Research Laboratory; (Syverson *et al* 1997) is their survey paper describing their work. Their ideas were strongly influenced by (Chaum 1981). A stated early goal was the support of anonymous web surfing and anonymous emailing. In 1997, criminal misuse of anonymity was not widespread. Anonymous email services such as anon.penet.fi existed through much of the early development of Tor; the developers were aware of these services and their limitations.

The first “production” version of Tor was released as an open-source project in 2003 (with an alpha version the year before). In 2004, Roger Dingledine, Nick Mathewson and Paul Syverson published a description of the “second generation” Tor mechanism (Dingledine *et al* 2004). This remains generally current, though there have been technical updates.

In 2006 a nonprofit organization The Tor Project was formed; it continues to manage development of the Tor software. The primary funder of The Tor Project has been the US government (Levine 2014), with the stated goal of supporting democracy activists in authoritarian countries.

Tor Circuits and Nodes

The basic building block of Tor is the bidirectional **Tor circuit**, built around a chain of **Tor nodes**, usually of length three. One end of the circuit connects to Alice, and the other end connects to a public website. The Tor circuit will, like a VPN, prevent the website from identifying Alice.

As such a circuit must have a public IP address as its remote endpoint, it cannot by itself provide anonymity for servers. We will return to this point below, but the short answer is that to access anonymous servers, both the client and the server create a Tor circuit, and these meet somewhere in the middle.

The lifetime of a Tor circuit is on the order of 10 minutes. That is enough for one complete web connection and its immediate followups. A single Tor circuit may also be used to contact multiple websites. After 10 minutes or so (the exact time is chosen randomly), the client creates a new Tor circuit, though if a circuit is in continuous use as part of a large file transfer then it stays in place for as long as necessary.

A Tor client user can browse the web with little fear that the sites contacted will be able to determine the user's IP address and thus the user's identity (though see below at “Potential Attacks”). A Tor client user can browse sensitive information, or can upload leaked files to the press, or can send and receive email (through an ordinary free email account or through a special Tor-only email account), all with negligible risk of identification by any but the most committed adversaries. In these and other cases, we assume for the moment that the server end of the Tor connection is a public Internet website.

Tor nodes are usually run by volunteers who are concerned about Internet privacy. Tor nodes do not operate in secrecy; the list of all Tor nodes is of necessity public, as users must have this list to create their Tor circuits. In 2018, there were a little over six thousand Tor nodes. The limiting factor of a typical Tor node is how much bandwidth the administrator is willing to devote to Tor traffic.

If Alice wishes to connect using Tor to a public IP address, say owned by Bob, then Alice's first task is to pick three (for a Tor circuit of length three) Tor nodes, which we will call Tammy, Terrell and Tim. Alice picks these nodes by downloading the list of all Tor nodes and then choosing the three at random. (Alice may choose her nodes so they meet additional bandwidth and stability requirements, though that slightly reduces the randomness.) The first node on the list here, Tammy, is Alice's “guard” node; to reduce the effectiveness of some correlation attacks (below), Alice may wish to use the same small set of guard nodes for several weeks. The last node on the list, Tim, must be a Tor node that has agreed to serve as a Tor **exit node**, below.

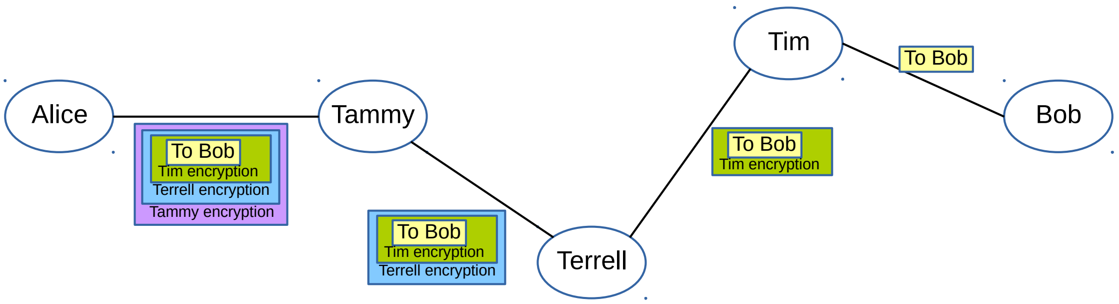

Diagram of a Tor circuit Alice—Tammy—Terrell—Tim,
with Internet TCP connection from Tim to Bob.
Packets, from Alice to Bob, are shown with layered Tor encryption,
with one layer stripped away by each Tor node of the circuit.
Additional layers of encryption do not make packets larger.

Once the circuit is built, Alice can send a packet to Bob by way of, in succession, Tammy, Terrell and Tim. Furthermore, none of the Tor nodes is aware of the IP address of any of the other non-adjacent nodes; that is, Tammy knows the IP address of Alice and of Terrell, Terrell knows the IP addresses of Tammy and Tim, and Tim knows the IP addresses only of Terrell and Bob.

As with a VPN, the authorities can likely identify Tim as having exchanged packets with Bob. However, the rules for Tor nodes prevent Tim from keeping any logs of its packet exchanges, and so the connection cannot be traced back to Terrell unless Tim has been subpoenaed or compromised. Even if this is the case, Terrell and Tammy would also have to have been compromised in order to trace the connection all the way back to Alice. This is unlikely, given Alice's random selection of Tammy, Terrell and Tim, though it does remain a theoretical risk. Another risk is a statistical attack, detailed below at “Traffic Correlation”, though as of today that too is mostly theoretical.

Bob can see Alice's connection as coming from Tim, and so can determine that the connection in question probably is using Tor (only probably, because host Tim might be used for non-Tor purposes as well). Some public websites do place some restrictions on what can be done via Tor connections; for example, Wikipedia limits editing over Tor, except in special circumstances.

Tor packets (often called cells) all have a fixed size; smaller messages are extended by padding, and larger messages are split over two or more Tor packets. Fixed-size messages make it harder to deanonymize users based on packet-size traffic analysis.

The length of a Tor circuit, normally three, can in principle be changed. However, this is seldom a straightforward configuration option; it usually requires recompiling the software, Increasing the circuit length may not result in material increases in privacy, on the theory that if the first and last nodes of the Tor circuit are compromised then traffic-correlation attacks have a reasonable probability of success regardless of the number of intermediate nodes.

Sending Packets

To set up the Tor circuit, Alice first contacts Tammy, and negotiates an appropriate cryptographic session key (as opposed to Tammy's public key), using Diffie-Hellman-Merkle key exchange. Alice then tells Tammy that the next hop is Terrell, and Tammy forwards packets from Alice on to Terrell.

Alice now repeats the key negotiation with Terrell. At no point is Terrell aware that the start of the Tor circuit is Alice; the key negotiation between Terrell and Alice is conducted via Tammy as an intermediary. Once the Alice—Terrell key is negotiated, Alice tells Terrell, over an encrypted channel unreadable by Tammy, that Tim is the next hop.

The final step is for Alice to negotiate a session key with Tim. Again, Tim knows only that the communications are coming from Terrell; Tim has no idea of Alice's real identity. Similarly, Tammy knows nothing about Tim.

At the head of each Tor packet is a two-byte “circuit identifier”, which is sent unencrypted. When Alice sends a packet to Tammy, she prefixes it with the circuit ID she used to set up the initial contact with Tammy. Tammy uses this circuit ID to look up the appropriate encryption key for this leg, and to look up the next hop in the circuit, Terrell. Tammy then sends the packet on to Terrell, updating the circuit

ID to the value Tammy negotiated with Terrell. The circuit ID is examined and updated by each Tor node until the end of the circuit.

Alice is now ready to send a packet to Bob. She includes Bob's address (but not her own), and encrypts the packet with the key she shares with Tim. She then re-encrypts everything with the key she shares with Terrell. Finally, she encrypts a third time with the key she shares with Tammy. After this last encryption, she attaches the circuit ID agreed on with Tammy. This layered encryption is known as onion encryption, after the layers of an onion.

The packet is then sent to Tammy, who decrypts it with the key shared with Alice. Tammy sees, from the circuit ID, that the next hop is Terrell, so Tammy sends it on. Terrell receives the packet and decrypts it with the Terrell-Alice key; the packet is still encrypted with the Tim-Alice key. Terrell sees from the circuit ID that the next hop is Tim, and sends it on. Tim removes the final layer of encryption, and sees that the final destination is Bob. Tim then sends this packet on to Bob.

Alice's packet to Tim is effectively wrapped in three layers of encryption. These layers are stripped away, one by one. This layering is notionally like the layering of an onion, which gives rise to the name Onion Routing. The layered encryption prevents any one of the Tor nodes from finding out more than they need to know of the others. For example, even if Tammy and Tim are both compromised, they cannot together trace the connection definitively back to Alice, because the Tim—Bob packets do not match the Alice—Tammy packets due to the differing layers of encryption. (However, an attacker now does have a good chance of unmasking Alice probabilistically due to traffic correlations between transmissions from Tammy to Terrell and those from Terrell to Tim; see below at “Traffic Correlation”.) In general, Alice's anonymity is has at least some protection as long as at least one of the Tor nodes on her circuit is not compromised.

Because each Tor packet appears completely different on each of the Alice—Tammy, Tammy—Terrell and Terrell—Tim links, due to the different number of layers of encryption, an outsider has no hope of correlating packets based on content, and thus tracing the connection back to Alice. There *is* a potential risk that such a correlation can be achieved by looking closely at traffic patterns and volumes; see below. In addition to the Tor encryption, Tor traffic on each circuit link is often also protected by Transport Layer Security (TLS) encryption, the standard Internet-connection encryption mechanism.

Actual connections over Tor must be made using the TCP protocol, used for the vast majority of Internet traffic. When Alice wishes to open a new TCP connection to Bob, she sends to Tim – via the circuit – a special Tor setup packet known as Relay Begin. This packet contains Bob's site address and the desired TCP port number (*eg* port 80 for web traffic). It is Tim that opens the TCP connection. Afterwards, Alice sends data to Bob (and vice-versa) through Relay Data packets. Tim converts Relay Data packets from Alice to the Tim—Bob TCP connection, and vice-versa.

Tim has access to the plaintext of the Relay Begin packet. Tim may have access to the plaintext of later packets, but it is common that Alice will negotiate an encrypted connection with Bob (using TLS), so that not even Tim can read the actual contents of further Alice—Bob exchanges.

When Bob wants to reply to Alice's request, Bob sends his data to Tim. Tim encrypts it with the key Tim shares with Alice, and sends it on to Terrell. Terrell adds another layer to the onion, re-encrypting the packet with the key Terrell shares with Alice, and sends it on to Tammy. Tammy adds a third layer, and sends the packet on to Alice, who is able to decrypt everything in the order Tammy, Terrell, Tim.

Tor exit nodes

The last node in the circuit, Tim, is in effect the public face of the circuit. If Alice is participating in the online sharing of a copyrighted work, or if Alice is accessing illegal content, then it is Tim that appears to be the guilty party. Managers of Tor exit nodes are routinely served copyright-related subpoenas, and are occasionally raided by the authorities. This is why there are, as of 2018, about 6,000 Tor nodes but only about 800 exit nodes.

The Tor Project maintains a standard exit-node notice, which many exit nodes post online (The Tor Project, Exit Notice). It contains an explanation of Tor, and various disclaimers:

> This router maintains no logs of any of the Tor traffic....
> Attempts to seize this router will accomplish nothing....
> If you are a representative of a company who feels that this router is being used to violate the DMCA, please be aware that this machine does not host or contain any illegal content.

A Tor exit node can have a complex *exit policy*, designed to limit the sites accessible via the node. An overall bandwidth limitation is nearly universal; an exit node can also enforce a lower per-connection bandwidth limitation. Exit policies may also block certain protocols and certain IP address ranges. While most nodes allow web access, via TCP port 80, it is not uncommon to block the sending of bulk email by denying Tor connections to TCP port 25. Exit-node policies are available online, allowing a Tor user to choose an exit node compatible with his or her objectives.

It is the bandwidth limitations put in place by Tor-node administrators – of both exit nodes and internal nodes – that are the primary cause of Tor's slowness. The triple-forwarding of each packet contributes a fixed delay, of perhaps a couple hundred milliseconds, but delays from this source do not accumulate proportionally as download sizes and web-page sizes increase. However, if a Tor node caps a connection's bandwidth at 100 KB/s, then a 2 MB page will take 20 seconds to load. The bandwidth-limitation issue has the greatest impact on those sharing large files, such as full-length movies.

Anonymous servers

As described so far, Tor only supports anonymous clients. It is also possible, and central to the idea of the Dark Web, to support anonymous *servers* (The Tor Project). The basic strategy is that the server and the client each create Tor circuits to an agreed-upon *introduction point*, and communicate via that point. Although the introduction point may be public, the server cannot be traced back from it any more than the client can.

Anonymous servers were originally called *hidden services*, though the less-perjorative term *onion services* has recently become popular.

Suppose Bob wishes to create an onion service. The first step is to create a public/private keypair, using RSA. The .onion service name is then based on an 80-bit secure hash of the public key. When users contact the .onion site, they will receive the public key, and can verify that it matches the site name.

For a first try, Bob might generate q4wgkcm22kdafxgb.onion as his public key. This is not very memorable, and so most sites try to generate large numbers of keys until they get one that begins with something human-readable. Normally, to get, say, 4 specific characters takes on order $32^4 \simeq 1{,}000{,}000$ tries. The .onion name for the New York Times, nytimes3xbfgragh.onion, likely took something like 30 billion tries. Facebook's onion address is facebookcorewwwi.onion; the first eight characters are intentional and the second eight just happened to end up as something that makes superficial sense in English. Facebook has claimed this was due to luck, but, given the computational resources available to them, their idea of luck may differ from that of normal-sized entities. (The New York Times has an onion address to support the leaking of documents; their onion server would remain hidden even if the authorities immediately demanded access to all their non-hidden servers. It is not clear why Facebook needs an .onion address, as users must login and so are not anonymous.)

After Bob generates his keys, and configures his web server, it is time for him to get his site out there. To do this he picks a set of Tor nodes – not necessarily exit nodes – that are known as *introduction points*. Bob builds Tor circuits to each of them. Bob then uploads to a public distribution service (built into Tor) the list of these introduction points, together with Bob's public key. Bob signs this list with his private key.

Alice obtains this information, and sets up her circuit Tammy to Terrell to Tim as before. This time Tim is asked to serve as a *rendezvous point*, but Tim need not be an exit node as Alice's traffic leaving Tim will not in any way be publicly visible.

Alice then sends Tim a secret password, and, via a second Tor circuit, contacts one of Bob's introduction points and sends the rendezvous point and the secret password. Both are encrypted with Bob's public key.

Bob now creates a circuit to the rendezvous point, Tim. In setting up this circuit, Tim is the ultimate destination. We will suppose Bob's circuit is Ty to Tula to Toni. After Bob verifies that Tim knows the secret password, and thus is legitimately the far end of Alice's part of the circuit, Alice and Bob can begin communicating via the combined circuit Tammy—Terrell—Tim—Toni—Tula—Ty.

Bob and Alice do not use one of Bob's introduction points in their combined circuit because the introduction points are publicly known. Avoiding them is more cryptographically sound, and also means that Bob's introduction points cannot be accused of carrying Bob's content. Tor exit nodes, by comparison, are accused of this sort of thing regularly.

Anonymity and browsers

All Tor provides is connection anonymity. It is possible, however, that Tor user Alice may be identified by her browser. First, the browser may send cookies to Alice's computer. Second, browser *fingerprinting* techniques, perhaps based on the unique set of fonts and plugins Alice has installed, may allow the browser to be identified uniquely to Bob (see below at Application Attacks). If Alice uses the same browser to browse non-anonymously, and if Bob shares the fingerprint information with other sites Alice has visited, then Alice's real IP address may be revealed.

To avoid this, Tor users typically use a browser that has been specially configured to resist common and not-so-common identification attacks. Fingerprinting is likely to be blocked, and all browser cookies should be deleted at the end of the session. So-called “private browsing” is often made the default. Secondarily, using a different browser with Tor than for “public” browsing makes it very hard to connect the Tor use to the public use.

The browser most commonly bundled with the Tor package is based on the Firefox Gecko browser engine, which supports a broad range of strong privacy features (many of which are not enabled by default in the standard version of Firefox). On Apple systems the Gecko engine is not available, and so either an alternative browser is used, or the user runs Tor in a virtual machine.

Tor itself simply creates network connections; ultimately, the end-user can use whatever applications he or she trusts. Users can, for example, with a little technical knowledge use Tor with any browser. This way the Tor system does not require anyone to trust their application software.

Legitimate uses of Tor

There is a long history of governments defending citizen surveillance with the argument “if you have nothing to hide then you have nothing to fear”. Some government agencies have long been suspicious of any use of Tor. There are, however, many everyday situations in which users might be more comfortable using Tor than a conventional web browser. For some of these, a VPN might serve as well, but Tor is free while VPNs are not.

Legitimate uses of Tor start with ordinary browsing for information that may be quite personal or sensitive. Someone searching for information about “HIV” or “addiction” might be very averse to public discovery or tracking.

Victims of stalking can use Tor to avoid revealing their IP address, and thus their location.

Ordinary people browsing non-sensitive topics might also want to use Tor if they simply wish to avoid relentless tracking by advertisers and large Internet companies. Some of this can be achieved by using an ordinary browser in “private” or “incognito” mode, but not all; in particular, private-mode browsing still reveals the client's IP address.

Persons engaged in political activism that is legal but that nonetheless attracts untoward government scrutiny – members of an antiwar group, for example, or Occupy Wall Street – might use Tor to read online manifestos and to communicate with fellow activists.

Tor is the tool of choice for those leaking government information, including whistle-blowers. Those leaking non-governmental information would probably be safe simply with a temporary email account, but Tor is sometimes used along with that for additional security. While leaking government information is sometimes against the law, such leaking is frequently viewed as a public good. The SecureDrop system, designed to support anonymous communications to the press and supported by many newspapers, is based on Tor.

Law-enforcement officers often use Tor so that their Internet use does not appear to be coming from an IP-address block assigned to the police. Citizens sometimes use Tor to report tips to the police anonymously.

Tor provides a lifeline for pro-democracy activists living under authoritarian regimes; this is in fact the US government's usual official argument in favor of continued Tor funding. Activists can keep in touch with one another and can disseminate news and images without risk of arrest.

There are frequent claims that US agents, and foreigners recruited by them for spying, use Tor to communicate. It is difficult to evaluate the volume of such traffic. It seems likely, though, that if the US government does make significant use of Tor for this and related purposes, then it would be likely to encourage other, non-espionage uses in order to provide a significant volume of cover traffic. The benefits of such cover traffic remain even if those other uses are of questionable legality.

Tor is often used for copyright infringement; for example, to allow someone to access online content via a peer-to-peer service without revealing their IP address. (Configuring bittorrent to use Tor securely in this matter is quite difficult, and is not recommended; bittorrent clients are notorious for leaking real IP addresses. Tor's bandwidth limitations also make it problematic for large file downloads.) While much of this might not be considered a “legitimate use”, it is usually not criminal, and in some cases may be defensible on Fair Use grounds.

There are also some legitimate uses of server-side anonymity. News organizations, for example, often use Tor onion servers for submissions from whistle-blowers and leakers so as to largely eliminate the risk of document seizure by the authorities.

As another example, consider a website supporting online discussion of sensitive topics, such as addiction or even ordinary medical issues. Such a site might allow users to log in; keeping the site anonymous will eliminate any risk that the authorities will demand information about site users. This may encourage users to participate, and to open up about their experiences. A site catering to stalking victims might maintain an onion server to enforce the privacy of its users.

The use of Sci-Hub to obtain scientific papers, while clearly copyright infringement, is sometimes justified on the grounds that such infringement has no negative effect on the incentives for content creation. Sci-Hub has lost most of its traditional domain names to governmental seizure, but its onion service remains available worldwide.

Anonymity and Crime

Tor's browser anonymity, without onion services, enables a variety of antisocial actions. Things that an isolated Tor user can achieve, without anonymous confederates, include the harassment of others with impunity, the posting of embarrassing revenge content, or the infringement of copyrights through peer-to-peer networks. With confederates, Tor users can exchange illegal content such as child pornography with one another.

Adding onion servers enables additional illegal actions; for example, user-to-server copyright infringement (though this has been widespread even with publicly identifiable servers) and large-scale free sharing of illegal content. The existence of onion servers also enables a wide range of “political” crimes; that is, offending various governments. Terrorists can use Tor for recruitment and training.

However, most traditional criminal activities usually involve the exchange of money, and for these Tor alone is of limited use. However, the development of Bitcoin in 2009, providing a mostly anonymous form of currency, has enabled the rise of Tor-based e-commerce sites that sell illegal products and services. The most common illegal item sold appears to be drugs, but weapons, stolen credit cards, hacking services and thugs for hire are also available.

At least one illegal Tor-based online marketplace, known as The Farmer's Marketplace, did attempt to use conventional payment systems such as Paypal. Despite attempts to obfuscate the delivery of funds, the US Drug Enforcement Agency was able to trace the flow of money and shut down the site.

Alternatives to Tor

The Invisible Internet Project, or I2P, is an alternative to Tor. It was founded in 2004. I2P is designed for hidden services only, not for anonymous browsing of public websites. I2P's circuits are all one-way; every endpoint creates at least one *in-tunnel* (or circuit) and one *out-tunnel*. If Alice and Bob wish to communicate, Alice's out-tunnel connects to Bob's in-tunnel and vice-versa. This makes traffic-correlation attacks much harder, as a typical observer will see information flowing in one direction only.

I2P also supports the connection of a single out-tunnel to the in-tunnels of multiple destinations. If Alice wants to communicate to Bob, Charlie and Debra, she can consolidate all her outbound traffic to any of the three into a single out-tunnel. The end point of that tunnel will forward the packets to their correct destination. This technique is called *garlic routing*, the idea being that Alice's bundled packets represent a head of garlic, broken into individual cloves at the exit of Alice's out-tunnel.

The Freenet system, first released in 2000, is another alternative to Tor. Like Tor and I2P, it relies on a cloud of Freenet nodes to handle routing. With Freenet, however, hidden data is also stored in that cloud. The data is distributed throughout the cloud; any one file may be split up over multiple nodes. Popular data is likely to be cached by multiple Freenet nodes.

By default, Freenet looks for hidden data anywhere in the Freenet network. Freenet also has a “darknet” mode, in which data is retrieved only from nodes on a manually generated list of trusted nodes.

Potential Attacks

The goal of DarkNet attacks is to breach one endpoint's anonymity, but not necessarily to be able to read the encrypted traffic. Even relatively weak evidence may be useful. For example, if, after collecting online evidence, the authorities believe there is a 10% chance Alice might be one of the persons connecting regularly to an online narcotics marketplace, they might then monitor what is being delivered to Alice's home, or carefully examine discarded wrappings. For onion services, the goal is to identify the physical location of the server involved, or the identity of one or more of its administrators. (In all the cases described below, the onion server was discovered first, which eventually led to the unmasking of the administrators.)

Traffic Correlation

The biggest deanonymization risk to most Tor users relies on traffic correlation; that is, by looking for transmission patterns at one point in the network that are repeated very soon after at another point, thus suggesting, over time, that the traffic is connected. Traffic correlation attacks tend to be easier when the two points in question are close to one another, but this is not essential if the attacker has sufficient resources. Some of these attacks require sufficiently high levels of resources and network access that they can only be executed by a government-level actor, but that may be small comfort. See (Syverson *et al*, 2000) and (Johnson *et al*, 2013).

Perhaps the simplest attack is discovery of the circuits passing through a single Tor node T. The attacker monitors all traffic entering and leaving T, recording the source IP address of each arriving packet and the destination address of each departing packet. If the attacker notices, over time, that whenever a packet arrives from A, another packet departs for B within 50 milliseconds, and vice-versa, that is very suggestive evidence that there is a Tor circuit through A, T and B. It may help if there are patterns to the traffic; for example, perhaps A regularly ends five packets and receives back eight, followed 200 ms later by another fourteen. If this (5,8,14) pattern shows up for only one of the other addresses T communicates with, it is likely that this other address represents B.

If A is user Alice, then the attacker has identified the first and second Tor nodes of Alice's circuit. If A and B are other Tor nodes, the attacker has identified an entire three-node Tor path, but not the user endpoints.

The information garnered by this attack is comparable to what would be obtained if the attacker had completely compromised node T, or was actually running node T. However, in isolation, discovery of the circuit neighbors at a single Tor node does not deanonymize any user. The real risk, below, is if this attack is perpetrated simultaneously against other Tor nodes.

Correlation-based circuit-neighbor discovery is not a sure thing. The Tor node T likely has many simultaneous circuits; based on data from (The Tor Project, Metrics), an order-of-magnitude estimate is 100. As each circuit lasts only ten minutes, there might not be time to deanonymize all the circuits before they expire.

The Internet Service Provider of node T is easily able to carry out this kind of correlation attack, possibly at the request of the ISP's government. If an ISP is induced launch this surveillance attack against one Tor node in its domain, it is likely to attack all of them. The attacker may also run some Tor nodes directly, gaining the same circuit-neighbor information. If the first and last nodes of Alice's circuit from earlier, Tammy and Tim, are surveilled or compromised, then Alice is deanonymized. If Alice has been accessing an onion service, it might take attacks on four such nodes to reveal Alice's connection to that service.

It is also potentially possible, though harder, to launch a larger-scale traffic-correlation attack that monitors Alice's traffic to and from the Tor node she is connected to, and also monitors a large number of exit nodes for matching traffic. For the latter, cooperation of a number of ISPs would likely be required. Initially, Alice's contribution to the exit-node traffic will be lost in the noise. However, over time, some correlations may appear. Again, specific traffic patterns may help. Although this attack is less certain, and generally more expensive, a success means that Alice is completely deanonymized.

To make this job easier, the ISP of a Tor node might even apply “traffic shaping” to that node's outbound traffic, to create recognizable patterns. For example, traffic from Tammy to other Tor nodes might be saved up and sent in batches a few tens of milliseconds apart. If this burst signature is then seen at another Tor node, that is evidence of a Tor circuit to that second node through Tammy.

Exit nodes may be monitored by their ISPs. It is also, however, straightforward for a committed adversary to set up a large number of exit nodes. This may have, in fact, been done, by various governmental agencies.

This kind of larger-scale attack might not even need exit-node monitoring. Websites can be profiled by the number of packets they send and receive (Hintz, 2003). Suppose a connection to a particular public site involves one packet sent to the site, seven sent back, three sent to, and then seventeen sent back. That (1,7,3,17) signature would likely still be apparent even if the site were accessed via Tor; the only question would be how many other sites have the same signature. Extending the length of the signature, or including timing information on the delays between packet exchanges, may make this kind of signature significantly more trustworthy. Building a database of signatures for a large number of websites is quite straightforward. While this attack is largely hypothetical today, work continues on making it effective.

Earlier, we claimed that if the first and last nodes of Alice's Tor circuit, Tammy and Tim, are compromised, then Alice can still not be definitively deanonymized, because the traffic Tim sends to Terrell cannot be matched with certainty to the traffic Terrell sends to Tammy due to the layer of encryption added by Terrell. However, if Tammy and Tim *are* compromised, traffic correlation is likely to unmask Alice with a high degree of probability. Investigators will look for bursts of packets sent by Tammy to some other node X, followed soon after by a very similar burst from X to Tim. At this point it is usually quite easy to infer that X is Terrell.

If a number of Tor nodes are, collectively, connecting to a host leased in a cloud datacenter that does not run any public services, that might lend support to the hypothesis that the host in question is hosting a Tor onion service. This situation can readily be monitored by the datacenter itself.

In theory, correlation attacks are straightforward to prevent, by having Tor nodes introduce random delays when forwarding packets, and by having the nodes also send considerable volumes of “fake” traffic. Neither of these approaches is practical, however; the first increases the delays experienced by Tor users to unacceptable levels, and the second uses up Tor-node bandwidth that is already in short supply.

In none of the specific examples described below in “Tor Identity Breaches” do correlation attacks appear to have played a role.

DNS leaks

Suppose Alice wishes to connect to, say, hackforums.net. The site's name must be looked up, using the Domain Name System, to determine its IP address. The correct way to do this is for Alice to set up her Tammy—Terrell—Tim circuit, as before, and then have her exit node, Tim, do the DNS lookup. Alice sends to Tim a Relay Begin message containing the string form of the site name, “hackforums.net”. If Alice's software is configured incorrectly, though, it is possible that Alice will send the DNS query directly to her local ISP, which will return the IP address (“B”, 2014). Her local ISP will likely keep a log record of this request, and Alice's attempt to access the site is revealed.

The browser bundled with most Tor software distributions is correctly configured to do remote-end DNS lookups, but Tor is also used with other, non-web protocols, such as email clients, Usenet news readers and the secure shell (ssh) login client. Configuring these so that DNS lookup works safely with Tor can be complex.

Relatedly, if at the beginning of Alice's Tor session she uses a conventional browser to search for “nytimes onion address”, an observer might suspect that Alice went to nytimes3xbfgragh.onion to leak something. This is especially true if it is already known that someone has leaked documents that were available only to Alice's work department of a dozen persons.

Attacks exist that monitor the DNS names looked up by a set of Tor exit nodes, through eavesdropping, but with this approach it is somewhat harder to tie a given request to Alice.

Application attacks

If Alice is browsing the Internet using Tor, her browser is probably the one packaged with Tor: a derivative of Firefox. Web browsers in general are notorious for having vulnerabilities. If Bob is running an onion service, odds are it involves Apache, MySQL and PHP. All three of those introduce vulnerabilities of the sort that Tor provides no protection against.

On the client side, many browser plugins leak information. Tor browsing should not use insecure plugins. Sometimes, though, Tor users have been talked into installing deanonymization plugins using the ruse that the plugin is a “security enhancement”.

There are a large number of techniques for fingerprinting browsers. Most of these techniques are heavily used in the normal-browsing world, by advertisers and their allies. A server may extract the lists of fonts and plugins; many browsers are uniquely determined by this. Another fingerprinting technique involves drawing an image on an offscreen “canvas” and checking for subtle rendering details. Yet another technique involves precise timing measurements of mouse movements, which serves to fingerprint the client human user, not the client system. None of these fingerprint techniques reveal the identity of the user by themselves, but if the same user generates the same fingerprint via public browsing, the jig is up. The Tor browser attempts to block most known fingerprinting techniques, though the user is often asked if the blocking should continue, and it is easy to click “no” by mistake.

The onion server, if compromised, may be able to serve Javascript to the Tor clients that extracts information about the clients, such as their public IP address. A Javascript program can be downloaded which instructs the user machine to contact a designated server via the machine's public IP address. The usual Tor browser configuration includes settings to block most Javascript, but these settings can be changed. Though it is used less commonly than in the past, the Adobe Flash plugin will also run Javascript.

Web servers, regardless of whether they are used with Tor, are subject to a wide range of attacks. SQL injection can lead to database compromise. If the database includes usernames, order history and shipping addresses, a great many client users are exposed.

A common approach to expose the server itself is to find a flaw or misconfiguration that exposes the public IP address of the server. This is likely what happened in the Silk Road, Playpen and Hansa Market cases below. Ironically, onion servers do not actually need public IP addresses; they can be behind a network-address-translation firewall, and be assigned only a private IP address that is useless for tracking.

Metadata

Suppose someone uses Tor to upload an image anonymously. Now suppose that the EXIF image metadata was left attached to that image, and that it contains the GPS coordinates of where the picture was taken, and the name of the owner of the camera. Anonymity is lost, through no fault of Tor.

Tor Identity Breaches

Eldo Kim was an undergraduate at Harvard University. During finals week in December 2013, someone used Tor to email a bomb threat to the Harvard authorities. At the time the email was sent, Harvard's network logs showed that Kim's laptop was the only campus device that had connected to any Tor node. That pretty much pinpointed Kim, who confessed when confronted by the authorities. Note that Kim would likely not have been unmasked had Tor been more popular on campus. (Brandom, 2013)

The Silk Road

The Silk Road, silkroad6ownowfk.onion, was the first contraband marketplace to exist on the Dark Web. The site was launched in early 2011, the identity of the owner was eventually discovered to be Ross Ulbricht. Ulbricht ran the site using the alias Dread Pirate Roberts, or “dpr”. The site primarily sold drugs, and also stolen credit-card numbers and online account credentials; sales of child pornography or of violent services were not allowed.

In its early months, the Silk Road faced a problem common with anonymous transactions: the seller may fail to deliver. To avoid this, the Silk Road instituted both a seller-review system and an escrow system. Under the escrow system, a purchaser who did not receive their ordered merchandise had some chance of obtaining a refund. Sellers on the Silk Road had to pay a fee to participate.

The Silk Road was known to the US Federal Bureau of Investigation and Drug Enforcement Administration early on, but they had no way to find out who ran it or where the server was. DEA agent Carl Force was, however, able to become an online confidante of Ulbricht (as dpr), under the alias “Nob”. Force had no idea, of course, about Ulbricht's real identity, or where the server was located.

Eventually the DEA was able to identify Curtis Green as a Silk Road employee, or at least as a customer, perhaps through seized packages. In January 2013 the FBI raided Green's home. After Green contacted Ulbricht about the arrest, Ulbricht allegedly tried to hire his online confederate Nob – actually Force – $80,000 to murder Green (Greenberg, 2015). The DEA organized fake photos of Green's death, and Ulbricht paid up. Ulbricht was never tried for this allegation. (After Ulbricht's trial, Force was convicted of stealing from the government some of the Bitcoins that were seized during the investigation.)

Over time, Ulbricht made a series of errors in *operational security*; these are issues that did not involve fundamental weaknesses in the Tor protocol. In March 2013, he posted a technical question about Tor on the StackOver.com site, specifically about how to connect to a Tor onion service using the cURL software package. He used an alias, “altoid”, but gave his email address as rossulbricht@gmail.com (Ulbricht, 2013). The email address, and the alias, were changed very soon after.

By June 2013, the FBI had figured out the location of the Silk Road Tor server, below; this is believed to be the result of a server configuration error (below). Ulbricht had been careful to pay for the server using Bitcoin and fake identification, so the discovery of the server did not lead quickly back to him.

Also in June 2013, Gary Alford, an Internal Revenue Service agent attached to the DEA, went searching for Internet posts touting the Silk Road when it was first getting started. He found one by a user with alias “altoid”, and connected that to the StackExchange.com post above. As of this point, Ulbricht was on the list of suspects (Popper 2015).

In July 2013, a package of nine fake IDs was intercepted at the US border (Hern, 2013). When investigators went to the address they had been shipped to, Ulbricht was there. Worse, a picture resembling Ulbricht was on the IDs. This strengthened the DEA's suspicions about Ulbricht; he was not arrested.

Another serious error was that on isolated occasions Ulbricht logged into the Silk Road server without using Tor. Some of these logins were from an Internet café a few blocks from Ulbricht's home. Once the FBI located the server, below, they were able to log these connections, and use them to determine Ulbricht's general location.

On October 1, 2013, Ulbricht was arrested in a public library in San Francisco. Two FBI agents created a distraction, while others grabbed Ulbricht's laptop, which was configured to encrypt itself had Ulbricht had enough time to close the lid. The Silk Road server was also seized and shut down at this time.

Ulbricht was convicted of narcotics trafficking and related offenses in February 2015. Numerous site dealers, and probably some site customers, were also eventually arrested and convicted.

From a technical perspective, the most interesting question is how the FBI was able to locate the Silk Road's server, which is what led eventually to Ulbricht. The FBI's official explanation, presented in an affidavit by Chris Tarbell (Tarbel 2014), was that the login page contained a misconfigured CAPTCHA software widget; these components ask the user to, for example, type in the letters appearing in a distorted image, and are intended to prevent automatic logins. Tarbell stated that he tried sending a variety of data combinations to the login page (a technique known as “fuzzing”), and at some point one of the replies contained an IP address that did not belong to a Tor node. When he attempted to connect directly to that IP address, a CAPTCHA identical to the Silk Road's came up.

While misconfigured systems can do odd things, CAPTCHA widgets do not normally return IP addresses at all. This has lead to suspicions that the FBI may not have been telling the full story (Krebs 2014). One possibility is that they were able to install some form of malware on the server; another possibility is that some input error (perhaps not on the login page at all) forced the site's PHP programming language to dump all its state as an error message, including the public IP address. It is also possible that Ulbricht made modifications to the CAPTCHA widget that did not quite work as planned.

At Ulbricht's trial, his legal team tried to argue, among other things, that he had simply set up a web server, and wasn't responsible for what was sold on it. However, the prosecution presented detailed message logs indicating that Ulbricht had a close hand in managing the site. He was convicted in February 2015, and sentenced by Judge Katherine Forrest to life in prison without parole. Ulbricht's legal team has claimed that the severity of the sentence was based in part on the uncharged allegation that Ulbricht had conspired to have Green murdered; Judge Forrest did mention the alleged murder-for-hire plot at the sentencing hearing (Judge Forrest 2015).

Ulbricht's team had also argued that the FBI raided the server, located in Iceland, without a warrant. Counterarguments include the fact that the server was not under US jurisdiction, that the raid was led by Icelandic authorities, that the server was leased (from a cloud provider) and operating contrary to the provider's terms of service, and that Ulbricht has never claimed he had an ownership interest in the server.

Running the Silk Road took considerable technical skill; Ulbricht would not have wanted for traditional, legal employment. It does not appear that Ulbricht created his site so that he would be able to obtain illegal drugs for himself more easily, though (Bearman and Hanuka, 2015) reports that in his youth he was a moderately heavy user of cannabis. One motivating factor, surely, was the promise of considerable wealth; Ulbricht's total earnings amounted to tens of millions of dollars, at a minimum.

However, Ulbricht was also a committed libertarian. On his linkedin page (Ulbricht, 2015) he wrote

> I want to use economic theory as a means to abolish the use of coercion and agression amongst mankind.... The most widespread and systemic use of force is amongst institutions and governments, so this is my current point of effort. … I am creating an economic simulation to give people a first-hand experience of what it would be like to live in a world without the systemic use of force.

Ulbricht's “economic simulation” was a marketplace in which buyers might purchase drugs free of governmental “coercion” and “systemic use of force”.

Silk Road 2

A month after Silk Road's seizure, the site reopened as Silk Road 2.0. The reopened site claimed to be under control of former Silk Road administrators. Three Silk Road 2.0 administrators were arrested the following month, probably traced by their activity on the original Silk Road. In November 2014, as part of Operation Onymous, Blake Benthall was arrested as the alleged owner of Silk Road 2.0, and the site closed (Wikipedia, Operation Onymous; O'Neill 2014). Several other onion-service marketplaces were also closed, though not the two largest marketplaces, Agora and Evolution. The seizure of Silk Road 2.0 is believed to have been enabled by operational-security errors by Benthall, and perhaps also by over-reliance on the only partial anonymity provided by Bitcoin transactions.

In 2015, the Evolution marketplace was closed by its owners as part of an “exit scam”: the owners walked away with an estimated $12 million in Bitcoin held in the site's buyer-escrow fund (Brandom 2015; Krebs 2015). The Agora marketplace closed voluntarily a few months later, with the owners citing increased security concerns about Tor itself.

Playpen

The onion service known as Playpen started in August 2014 as a marketplace for child pornography. An FBI investigation began soon after, and received a significant boost in December 2014 when a source reported to the FBI that under some conditions the site leaked its real IP address (Rumold 2016). The FBI was then able to track the site to a data center in Virginia, and, from there, was able to identify the site's owner, Steven Chase.

The site was seized on February 20, 2015, and Chase was arrested. However, the FBI kept the site running until March 4 in order to collect information about the users.

Playpen's customers would have had no reason to supply a shipping address (*cf* users of Hansa Market, below), so the FBI tried a different approach. They took advantage of a vulnerability in the version of the Firefox-based browser then bundled with Tor, and were able to obtain IP addresses of about 1300 users. No information about the details of the vulnerability have been released, but it seems likely that it involved execution of code remotely installed on users' computers by the Tor server.

The FBI obtained a search warrant in the Virginia district where the server was found, signed by Magistrate Judge Theresa Buchanan. The warrant allowed for the deployment of a “network investigative technique”, or NIT, against any computer that logged into the Playpen server (Crocker 2016).

The warrant was controversial in that it did not specify the locations of the user computers to be searched, and most of them turned out to be outside the Virginia district in question. Rule 41 of the Federal Rules for Criminal Procedure specified at the time that warrants could be issued “to search for and seize a person or property located *within* the district.”

Rule 41 was amended at the end of 2016 specifically to allow the use of tools like the one used by the FBI in this case; a warrant may now be issued for the search of computers in any location if “the district where the [computer] is located has been concealed through technological means” (Federal Rules of Criminal Procedure).

The courts have not yet fully resolved the Fourth Amendment issues at stake here, or whether the amended Rule 41 passes Constitutional muster.

In March 2017, in the Playpen case *United States v. Jay Michaud*, the government dropped the charges when the FBI was ordered to reveal the precise technical details of the “network investigative technique”. However, the indictment was dismissed without prejudice, allowing the government to re-file the charges at a later date, presumably after the point when the browser vulnerability is patched and therefore of no further use (Newman, 2017).

Stephen Chase, unlike Ross Ulbricht, has issued no philosophical manifesto justifying his site. The forfeiture order following his conviction (Judge Voorhees, 2017) lists no cash or cryptocurrency assets, indicating that Chase earned little if any money from the site. Several other child-pornography websites have been run on a free-exchange basis, suggesting their founders shared the paraphilia of their customers.

AlphaBay and Hansa Market

AlphaBay and Hansa Market were two competing Tor-based online e-commerce sites. Both primarily sold illegal drugs.

On July 4, 2017, the an international law-enforcement operation led by the FBI seized AlphaBay's servers, operating in Canada and the Netherlands, and arrested the site's owner, Alexandre Cazes, in Thailand (Greenberg 2018). Exactly how the authorities found the servers has not been released, but operational-security errors appear to have played a large role. An email address used by AlphaBay, for example, had been used previously by Cazes for a legitimate business, and the pseudonym Cazes used on the site had also been used elsewhere by him previously.

Cazes was found dead in his cell a week after his arrest, apparently by suicide.

At the time of the shutdown, AlphaBay had 350,000 product listings, according to the FBI, versus about 14,000 for the Silk Road at the point it was shut down (FBI 2017).

AlphaBay customers went scrambling for new sources. Most ended up at Hansa Market, which had been the second-largest online drug marketplace before the AlphaBay closure. But on July 20, 2017, Hansa Market too shut down. Worse, for buyers and sellers, it turned out that Hansa Market had been operating under complete control by the Dutch police since June 20, as part of a Dutch-German-American operation. The Dutch team had also figured out how to identify large numbers of buyers and sellers.

According to Dutch police, sometime in late 2016 an independent computer-security firm first got wind of the possible location of a Hansa Market development server, used for testing new software before it was migrated to the production servers, and notified the Dutch authorities (Greenberg 2018). How this discovery was made has not been released, but the development server is believed to have accepted non-Tor connections and thus would have been vulnerable to IP-address scanning.

When the Dutch authorities began monitoring the server, in a Dutch data center, they soon discovered one of Hansa Market's production servers in the same data center, and other Hansa servers in Germany. Searching those servers revealed references to two administrators' real identities.

Shortly after their discovery, these Hansa Market servers went dark, as Hansa Market itself was migrated to different servers. However, in April 2017 the police were able to track a Bitcoin payment, via blockchain analysis, from the suspected administrators to a data center in Lithuania. That data center turned out to be hosting the new Hansa Market servers.

When the servers were taken over in June, the police configured them to save all messages sent through the site. The site continued to strip EXIF metadata from images uploaded by dealers, but now began logging it first; this data often included GPS coordinates. Sellers were sent cryptographic-key files in Excel format; when opened, these files contained a macro that contacted the authorities (Dutch National Police Corps 2017).

Hansa Market continued to operate normally, to all appearances, though the police did ban the online sale of the exceptionally dangerous drug fentanyl. That decision, however, was initially proposed by existing Hansa Market moderators (Krebs 2017, Popper 2017).

At one point after the AlphaBay seizure, Hansa Market was getting so many new registration requests that the police had to temporarily disallow new registrations.

At the time Hansa Market was finally shut down, the police had information on tens of thousands of customers, and hundreds of dealers. Information on customers came from the messages they sent dealers; for about 10,000 customers, one of their messages included a shipping address. It is not expected, however, that more than a fraction of those customers will actually face prosecution.

Bibliography

"B", David (2014) Common Darknet Weaknesses 3: DNS Leaks and Application Level Problems, Privacy PC, Jan 29, 2014. Available at http://privacy-pc.com/articles/common-darknet-weaknesses-3-dns-leaks-and-application-level-problems.html, accessed March 2018.

Bearman J and Hanuka T (2015) The Rise and Fall of Silk Road, Wired Magazine, April 2015, available at https://www.wired.com/2015/04/silk-road-1/

Brandom, Russell (2015), Feds found Silk Road 2 servers after a six-month attack on Tor, Jan 21, 2015, in: The Verge, available at https://www.theverge.com/2015/1/21/7867471/fbi-found-silk-road-2-tor-anonymity-hack. Accessed February 2018.

Brandom, Russell (2013) FBI agents tracked Harvard bomb threats despite Tor, Dec 18, 2013, in: The Verge, available at https://www.theverge.com/2013/12/18/5224130/fbi-agents-tracked-harvard-bomb-threats-across-tor. Accessed February 2018.

Chaum, D (1981) Untraceable electronic mail, return addresses, and digital pseudonyms, Communications of the ACM, Volume 24 Issue 2, February 1981.

Crocker, Andrew (2016) Why the Warrant to Hack in the Playpen Case Was an Unconstitutional General Warrant. Electronic Frontier Foundation, Sept 28, 2016. Available at https://www.eff.org/deeplinks/2016/09/why-warrant-hack-playpen-case-was-unconstitutional-general-warrant, accessed Feb 2018.

Dingledine R, Nick Mathewson N and Syverson P (2004) Tor: The Second-Generation Onion Router, Proceedings of the 13th USENIX Security Symposium, San Diego, California, August 2004.

Dutch National Police Corps (2017) Underground Hansa Market taken over and shut down, July 20, 2017. Available at https://www.politie.nl/en/news/2017/july/20/underground-hansa-market-taken-over-and-shut-down.html, accessed March 2018.

FBI (2017) Darknet Takedown: Authorities Shutter Online Criminal Market AlphaBay, Federal Bureau of Investigation announcement, July 20, 2017. Available at https://www.fbi.gov/news/stories/alphabay-takedown. Accessed March 2018.

Federal Rules of Criminal Procedure, Rule 41: Search and Seizure. Available at https://www.law.cornell.edu/rules/frcrmp/rule_41. Accessed March 2018.

Judge Katherine Forrest (2015) United States of America v Ross Ulbricht, 14 Cr. 68 (KBF) (sentencing hearing), May 29, 2015, available at https://freeross.org/wp-content/uploads/2015/05/Sentencing_2015-May-29.pdf, accessed February 2018.

Greenberg, A (2015), Silk Road Boss' First Murder-For-Hire Was His Mentor's Idea, Wired Magazine, April 2015. Available at https://www.wired.com/2015/04/silk-road-boss-first-murder-attempt-mentors-idea, accessed March 2018.

Greenberg, A (2018), Operation Bayonet: Inside the Sting That Hijacked an Entire Dark Web Drug Market, Wired Magazine, March 2018. Available at https://www.wired.com/story/hansa-dutch-police-sting-operation, accessed March 2018.

Hern, Alex (2013) Five stupid things Dread Pirate Roberts did go get arrested, The Guardian, October 3, 2013, available at https://www.theguardian.com/technology/2013/oct/03/five-stupid-things-dread-pirate-roberts-did-to-get-arrested. Accessed February 2018.

Hintz, Andrew (2013) Fingerprinting Websites Using Traffic Analysis, Proceedings of the 2nd International Conference on Privacy Enhancing Technologies (PET), 2003.

Johnson A, Wacek Chris, Rob Jansen, Mica Sherr, Paul Syverson (2013) Users get routed: traffic correlation on tor by realistic adversaries, Proceedings of the 2013 ACM SIGSAC conference on Computer & communications security, November 2013.

Krebs, B (2014) Silk Road Lawyers Poke Holes in FBI's Story, KrebsonSecurity. Available at https://krebsonsecurity.com/2014/10/silk-road-lawyers-poke-holes-in-fbis-story, accessed March 2018.

Krebs, B (2015) Dark Web's “Evolution Market” Vanishes, KrebsonSecurity. Available at https://krebsonsecurity.com/2015/03/dark-webs-evolution-market-vanishes, accessed March 2018.

Krebs, B (2017) Exclusive: Dutch Cops on AlphaBay “Refuges”, KrebsonSecurity. Available at https://krebsonsecurity.com/2017/07/exclusive-dutch-cops-on-alphabay-refugees, accessed March 2018.

Levine, Yasha (2014) Almost everyone involved in developing Tor was (or is) funded by the US government. Pando.com, July 16, 2014. Available at https://pando.com/2014/07/16/tor-spooks, accessed March 2018.

Newman, LH (2017) The Feds would rather drop a child porn case than give up a Tor exploit, Wired Magazine, March 7, 2017. Available at https://www.wired.com/2017/03/feds-rather-drop-child-porn-case-give-exploit; accessed March 2018.

O'Neill, PH (2014) The police campaign to scare everyone off Tor, The Daily Dot, Nov 7, 2014, available at https://www.dailydot.com/layer8/tor-crisis-of-confidence/, accessed March 2018.

Popper, Nathaniel (2015) The Tax Sleuth Who Took Down a Drug Lord, The New York Times, Dec 25, 2015, available at https://www.nytimes.com/2015/12/27/business/dealbook/the-unsung-tax-agent-who-put-a-face-on-the-silk-road.html, accessed March 2018.

Popper, N (2017) Hansa Market, a Dark Web Marketplace, Bans the Sale of Fentanyl, The New York Times, July 18, 2017, available at https://www.nytimes.com/2017/07/18/business/dealbook/hansa-market-a-dark-web-marketplace-bans-the-sale-of-fentanyl.html, accessed March 2018.

Rumold, Mark (2016) Playpen: The Story of the FBI’s Unprecedented and Illegal Hacking Operation. Electronic Frontier Foundation. Sept 15, 2016. Available at https://www.eff.org/deeplinks/2016/09/playpen-story-fbis-unprecedented-and-illegal-hacking-operation, accessed February 2018.

Syverson P, David Goldschlag D and Michael Reed M (1997) Anonymous Connections and Onion Routing”, Proceedings of the 1997 IEEE Symposium on Security and Privacy, May 1997.

Syverson P, Tsudik G, Reed M and Landwehr C (2000) Towards an Analysis of Onion Routing Security, International workshop on Designing privacy enhancing technologies: design issues in anonymity and unobservability, Berkeley California, July 2000, Springer-Verlag New York, Inc, pp 96-114.

Tarbell, C (2014) Declaration of Christopher Tarbell, United States v Ross Ulbricht, US District Court, Southern District of New York, S1 14 Cr. 68 (KBF), September 5, 2014, available at https://freeross.org/wp-content/uploads/2018/01/140905-Tarbell-Declaration.pdf, accessed March 2018

The Tor Project, Inc. (undated) Tor: Onion Service Protocol. https://www.torproject.org/docs/onion-services.html.en, accessed February 2018.

The Tor Project, Exit Notice (undated) This is a Tor Exit Router. https://gitweb.torproject.org/tor.git/plain/contrib/operator-tools/tor-exit-notice.html, accessed February 2018.

The Tor Project, Metrics (2018) Welcome to Tor Metrics. https://metrics.torproject.org, accessed April 2018.

Judge Richard Voorhees (2017) U.S. v. Chase, Amended Preliminary Order of Forfeiture, Docket No. 5:15cr15, April 19, 2017. Available at https://www.leagle.com/decision/infdco20170420c76. Accessed April 2018.

Ulbricht, Ross (by assumption) (2013) in: https://stackoverflow.com/questions/15445285/how-can-i-connect-to-a-tor-hidden-service-using-curl-in-php, March 2013. Accessed March 2018.

Ulbricht, Ross (2015 [estimated]), in: https://www.linkedin.com/in/rossulbricht. Accessed March 2018.

Wikipedia (undated) Operation Onymous. https://en.wikipedia.org/wiki/Operation_Onymous, accessed February 2018.