Fun with onion routing

Michael Owen, senior security consultant, Information Risk Management plc

While the internet has been seen by many as an opportunity to spread information and opinions, censorship and access monitoring have been seen to be gradually increasing in a number of nations. From the great firewall of China to attempts at online interception such as the FBI's Carnivore system, it has seemed that freedom of information on the internet is under threat. In recent years, one particular mechanism has been gaining in popularity both for those escaping censorship and those who resent continual monitoring and a presumption of guilt. This mechanism is onion routing, particularly as used in Tor, a proxy and router-based tool computer users can use on their desktops to achieve uncontrolled, anonymous connectivity to the outside world.

Anonymity and its attackers

Online anonymity is a difficult problem. System designers must devise a communications system which can run on a wide range of hardware of unknown levels of trust, with attackers who could in theory have access to any level of network traffic, watching for matches and patterns. In a worst case scenario, attackers might have control of almost all servers, and be able to view all traffic passing in, through, and out of the system.

Given unlimited resources and time, a wide range of mechanisms can be deployed to create highly anonymous network traffic. Unfortunately, real-world constraints on network resources and user patience for rescheduled data have limits, and as such, lines have had to be drawn in the establishment of workable solutions.

Tor has chosen a pragmatic threat model, the main threat being "an adversary who can observe some fraction of the network traffic; who can generate, modify, delete, or delay traffic; who can operate onion routers of his own and who can compromise some fraction of the onion routers."[1]

Thus, Tor does not intend to protect users from a traffic matching situation where an attacker can simply monitor a user and all exit nodes. Instead, Tor attempts to protect the overall traffic of the routing system from analysis which would identify sections for further, more detailed analysis.

Anonymity 101 - traffic analysis

As the fundamental consideration of an anonymous network, traffic analysis should be explored before looking at Tor in too much detail. Traffic analysis is the examination of network traffic flows to establish the identities of parties involved. This is done through examination of the volume and timings of the traffic being seen at various points throughout the system. This analysis is done independently of the network traffic contents, which are assumed to be hidden, as non-hidden traffic provides all the information traffic analysis wishes to uncover, and more.

Clearly the level of access to traffic an attacker has makes a critical difference. An attacker with access to all network traffic in a system could, for example, easily match connections of particular sizes to both sides of a conversation if no countermeasures are in place. If an attacker is limited to only monitoring a small number of connections at a time, he will be limited to targeting suspected communicators and anticipating their communications targets.

There are a number of mechanisms which can be used to protect against traffic analysis, and most have been explored in the precursors to Tor which we examine below. These include traffic normalisation, which uses a combination of splitting and padding to ensure that all network communications takes place using blocks of a constant size. In another technique, sequence alteration messages or parts of messages have their ordering altered to prevent basic matching of input to output on a one-to-one basis. All of these mechanisms can be used to improve anonymity, but each comes at a cost.

Onion routing precursors

Looking back, the first step into anonymising systems was taken by American cryptographer David Chaum, who published his paper on what became known as mix networks[2]. These were networks for anonymous and pseudoanonymous message delivery, often across what was known as a cascade of mix servers. Chaum's mix networks laid the path for many further anonymity tools with its encryption layering scheme for the movement of mail across a cascade of mix network servers.

After the first implementation of a version of mix networks in the form of Cypherpunk remailers, remixing, message padding, and dummy traffic were incorporated into the design of Mixmaster remailers, and eventually the Mixmaster II anonmymous remailers[3].

Work on email anonymity continues with the Mixminion system[4], but email anonymity has had the opportunity to move in a different direction from the onion routers this article discusses. These early systems focused on high latency communications, such as email and usenet postings. High latency systems have the advantage of being tolerant of delays which would be completely unacceptable in low-latency communications such as web browsing and IRC chats. With the rise of the web, there was clearly a need for technologies to anonymise low-latency communications.

"High latency systems have the advantage of being tolerant of delays which would be completely unacceptable in low-latency communications such as web browsing and IRC chats."

The first treatments of anonymity for these low-latency communications came in the mid-nineties, when the principles of mix nets and some novel techniques were applied to network models in Wei Dai's Pipenet model[5]. The Pipenet model focused on strong anonymity, and once again raised the issue of anonymity versus practicalities. Pipenet anonymity is very strong, but a large amount of processing and padding was involved in the movement of traffic across a Pipenet network. These techniques boost anonymity at the expense of bandwidth, which has been a significant cost.

The Pipenet model was the inspiration for much of the design of the Freedom network, a product of then Zero-Knowledge systems. Freedom was a major commercial enterprise, offering pseudoanonymity commercially, with all mix servers owned and maintained by a corporate entity.

The Freedom network was short-lived, but did discover two points for future anonymity designers to keep in mind. Firstly, while people appreciate anonymity, they may not appreciate it not enough to pay the costs a strongly anonymising system can incur. Secondly, some mechanisms which add to anonymity such as link padding can add a considerable cost to a commercial system[6].

Sprouting onions

The first iteration of onion routing was proposed in 1996, in a paper published through the Naval Research Laboratory's (NRL) Center for High Assurance Computer Systems. Funded by the Office of Naval Research (ONR), it proposed a system architecture specifically designed to limit a network's vulnerability to traffic analysis[7].

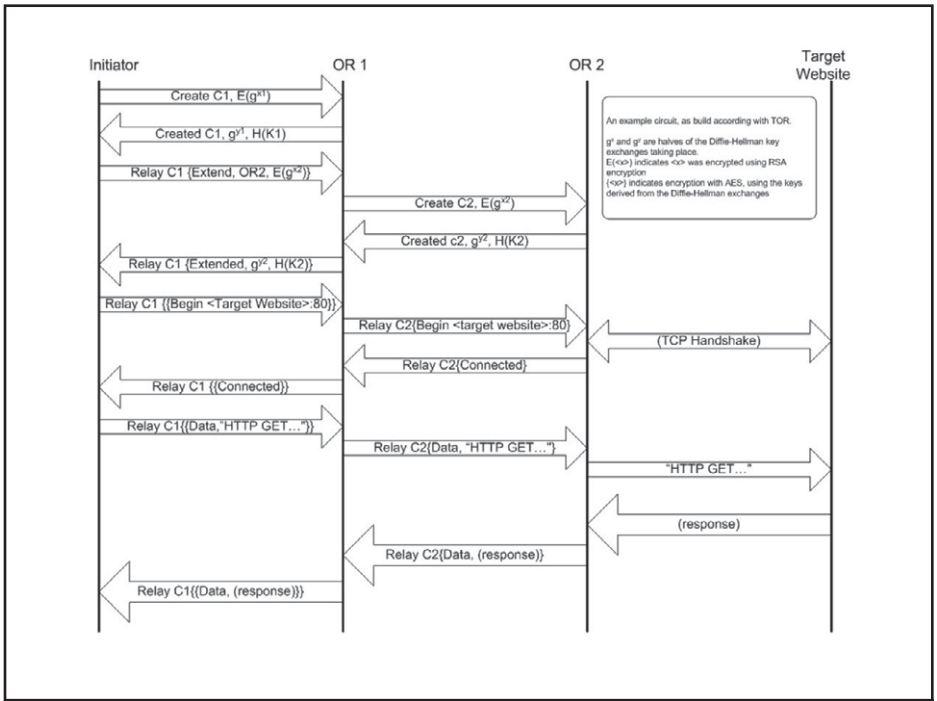

Figure 1: Data flows required for circuit building

It is worth noting that mitigating traffic analysis does not necessarily equate to anonymous connectivity. The goal of onion routing was to protect communications from outside identification via traffic analysis. This does not preclude identification within the chain of communication, which may range from completely anonymous to mutually authenticated.

It was in these first generations of onion routing that the name was coined. The establishment of communication within an OR network takes place by passing layered units of key and routing information. These packages of encrypted connection information are layered like the layers in an onion, with each server, known as an onion router (OR) in the chain stripping off its layer of the onion, exposing the next layer for the next server in the circuit.

The OR design progressed rapidly to a next generation onion router with a variety of improvements, including the splitting of the system into defined subcomponents. This opened the door for systems to run their own onion routing proxies without running servers. It was an important development for clients, and one leveraged by Tor to make onion routing more accessible to home users.

Tor: The third generation

The Tor design was first seen in 2004, effectively becoming the third generation of onion router. The name 'onion routing' persists despite the fact that Tor does not use onions for the establishment of connections between each OR in a virtual circuit. The onions came to be recognised as a shortcoming; their design was flawed, and could potentially lead to compromise due to a lack of perfect forward secrecy.

When onions are used, it is theoretically possible for a hostile OR to record traffic and then compromise successive ORs in a circuit to force them to decrypt the data through the re-generation of derived keys. Tor has had to devise a mechanism for link traffic encryption which provides the perfect forward secrecy needed to ensure that in the event of a server compromise, all downstream ORs are not also compromised.

The creation of the virtual circuit route is performed by the onion proxy (OP) the client device connects to,

however the mechanism used in TOR is an incremental one. The OP, which has obtained a listing of all available ORs, assembles a planned circuit for traffic routing based on criteria including OR uptime and bandwidth. Once a circuit has been planned by the OP, it sends a create cell off to the first OR in the chosen circuit, and uses a Diffie-Hellman (DH) key exchange to agree keys through a response cell known as a created cell.

For each OR step past the first step, the initiating OP simply sends out a relay extend cell specifying the next OR, once again with the appropriate payload for a DH exchange to take place with the new OR in the circuit. The OR returns a relay extended cell, including its half of the DH handshake. This continues for as many iterations as are necessary to complete the circuit (see figure 1).

Encryption of traffic takes place at two levels. Firstly, all ORs maintain links with all other operational ORs, and use TLS to encrypt all inter-OR traffic, as well as traffic between ORs and OPs. Secondly, the DH-derived encryption keys mentioned above are used for AES encryption of data between ORs and the OP, with each OR taking incoming encrypted data, and either stripping off the encryption layer associated with it, in the case of traffic passing away from the OP, or adding another layer of encryption in the case of traffic passing back towards the OP (see figure 2).

Once a circuit has been established as described, the end user is free to create and send off relay cells.

Data movement in a live circuit

Relay cells are the data units of Tor connections, passing through the circuits built by an OP. Once a circuit has been established, communications can begin through the use of a relay begin cell being sent from the OP, telling the circuit to open a connection to a particular hostname or IP address. Once the OP has received a follow-up relay connected cell from the border OR, relay data cells are used until a relay teardown cell from the OP initiates the destruction of that particular circuit.

> "Interestingly, Tor does not make use of any of the traffic padding or mixing techniques which have been discussed for other systems."

Relay cells are handled in a similar way to onions and data cells in onion routers, with layered encryption being applied by the OP for outgoing cells, and layered decryption for cells arriving at the OP. When a cell is received by an OR, the OR looks up the circuit in its internal tables, and decrypts the cell with the appropriate keys. The OR then determines if the decrypted cell has a valid digest (as created by the OP during cell creation). If a valid digest is found, the OR then acts as the border OR for that cell, relaying it to the outside world as appropriate. If the digest is not valid, the OR looks up the circuit and OR for the next step in the circuit, sets the circuit as appropriate in the cell, and forwards the cell on to the next OR.

By setting the digest to be valid for an OR partway through a defined circuit, the OP is able to force variations in the terminating OR. This capability is known as leaky pipe functionality, and is a low-impact mechanism for adding complexity to a traffic analysis attack against Tor.

Padding and mixing in Tor

Interestingly, Tor does not make use of any of the traffic padding or mixing techniques which have been discussed for other systems. This comes from a move within the Tor project to assemble an anonymity system which provides a pragmatic level of security.

Previous experience with projects such as the Freedom Network[6] have

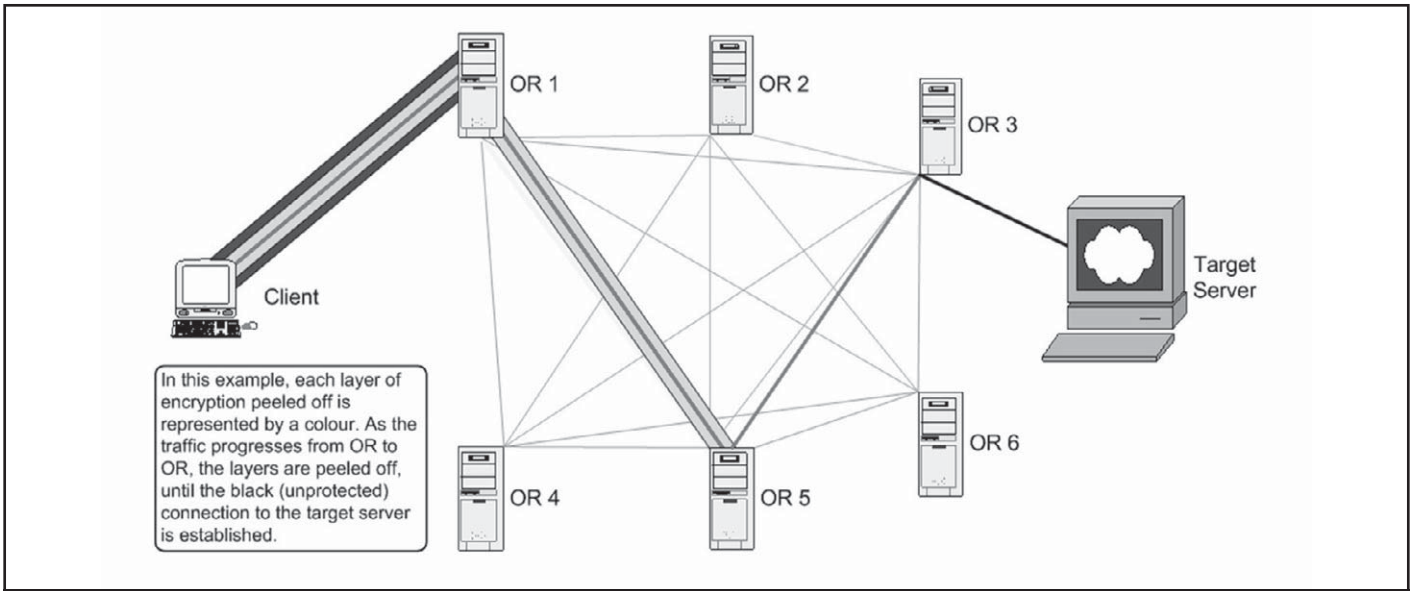

Figure 2: AES encryption wrappers in Tor networking

shown that padding and traffic shaping can be very expensive mechanisms to deploy on a wide scale, and the benefits of such mechanisms have been difficult to quantify. The Tor project has left the door open for such mechanisms to be incorporated in the future, however designs for such mechanisms must be "proven and convenient."[1]

Location hidden services

Tor provides a service for establishing location hidden services, which allow users to offer services via Tor without exposing their IP address or physical location. This facility makes use of rendezvous points, which serve as prearranged meeting points for service providers to meet up with service users.

When a service provider (Bob) decides to establish a hidden server, he generates a public key-pair to identify his service, and selects a number of ORs to act as what are known as introduction points (IPs). Bob must then build circuits to each of his IPs, configuring them to wait for requests to access his hidden service. These IPs are then advertised by Bob on the Tor lookup service, with advertisements created by Bob and signed using previously generated keys.

When a service user (Alice) decides to connect to Bob's service through whatever mechanisms he has employed to alert others to his service, Alice then looks up Bob's service in the Tor lookup service. Alice then selects an OR as her rendezvous point (RP), builds a circuit to this point, and then relays the details of the rendezvous point to Bob via one of his introduction points, including the first half of a DH key exchange.

If Bob elects to respond to this request, he builds a circuit to Alice's RP, completing the DH handshake via the exchange of a rendezvous cookie. The RP then associates its circuit with Alice to its circuit with Bob, and becomes a normal OR in an Alice to Bob circuit.

Alice is then free to send a relay begin cell across the circuit to Bob's OP, which connects to Bob's hidden server. Alice is now free to communicate across the circuit as a normal Tor circuit, without having any knowledge of where the hidden server is located.

Potential and real users of Tor

When confronted with the possibilities of Tor, the standard response tends to be 'who could want this?'. In recent years, there has been a stigma attached to attempts to escape monitoring, with people often declaring that those with nothing to hide should be happy to share everything about themselves with all and sundry. It should be said at the outset that there is no denial that a tool such as Tor could be used by criminals to evade identification in some situations, just as many other enabling technologies can be used for both negative and positive purposes. What must be considered before condemning Tor as a criminal tool are the users who benefit from Tor, including those who are benefiting from it right now.

A large potential user base for tools such as Tor is found in any nation whose government strives to restrict access to one or more subjects online, particularly though the use of firewalls employing blacklisting to block sites such as news sites, government sites, or sites relating to human rights.

At present, Tor is partially able to circumvent these controls, having been used to some success to circumvent controls such as the great firewall of China. While this capability has been suffering of late, Tor are working to improve capabilities in this area.

For boosting the flow of information from a restricted nation, location hidden services provide an ideal mechanism for information dispersal. Such servers are ideal not only for spreading news from a repressed area to the rest of the work, but also for spreading government-suppressed information in a country.

A major user of Tor is one which might not occur to the casual examiner of Tor functionality: western governments. It is no accident that led to the Naval Research Laboratory's funding Tor usage. The capability to anonymously search foreign websites for information is valuable, and is directly mentioned in a 2004 presentation by the NRV to the National Science Foundation.

Tor also provides an excellent tool for establishing communication channels which are protected from network traffic analysis that would link them back to a major western nation. Resistance groups requiring intelligence assistance from foreign powers, intelligence agents operating in foreign areas, and journalists wishing to bypass censors will all find Tor services useful.

Recent attacks

A paper has been released from the University of Colorado which leverages some of the routing optimisation mechanisms in Tor to route traffic towards malicious ORs[8]. Tor OPs are by design intended to make heavier use of ORs with higher bandwidth availability and high uptimes, and these statistics have been targeted by researchers for attack. It has been found that with a relatively low number of servers in a Tor group, it is possible to greatly increase the likelihood of malicious servers being used.

This attack is possible either through the use of malicious servers which are truly highly-available with good bandwidth, or via servers which simply claim to have these attributes. Once this is done, it is generally possible to link data flows through malicious servers to identify initiators and their targets, compromising the anonymity of the system. The level of malicious servers required in tests was 5-10%, which would mean a rather large number of servers in the current Tor infrastructure.

This attack is not entirely unknown, however there are few if any planned countermeasures for it at present. To date, the Tor community has concluded that such an attack would be apparent. Instances of suspicious numbers of servers appearing on the Tor network have been observed and queried in the past.

A further attack which is based on known issues with Tor has also recently been in the news. HD Moore, of Metasploit fame, has recently developed an altered Tor server known as

Torment, which monitors traffic from circuits terminating on the altered OR, and attempts to obtain the IP address of the client when certain keyword matching takes place in the outbound and inbound traffic. This attack is notionally intended to target criminals using Tor, such as paedophiles.

The client is located by injecting of HTML into returning traffic that points the client at a Java applet which has been engineered to leak this information through DNS lookups. It is known, and indeed advertised by Tor, that active web code could compromise anonymity. The Tor website advises users to disable active content in web browsers.

Alternative technologies for anonymity

A number of anonymising services are available online today. Web sites such as Anonymiser and FindNot offer a level of anonymity, generally acting as one-step proxies with no mechanisms in place to confound traffic analysis techniques. These systems work as advertised, but the security offered is limited, as there is only one set of hosts to monitor, and no mixing of traffic between hosts.

These tools effectively have selected a threat model which precludes any attacker having visibility of both incoming and outgoing traffic for any server, and assumes that no hosts are either hijacked or run by malicious entities. More importantly, they are fixed domains, and thus easily blocked by network blocks such as the great firewall of China.

Apart from these limited tools, we have found no complete alternatives to Tor. Systems are often proposed, but have generally remained theoretical. One system to make the move from theory to implementation is FreeNet, an anonymous information sharing tool[9].

Readers interested in other designs which have appeared in the past few years should examine Crowds[10] and Hordes[11]. Similar systems can be found in a number of research papers listed in areas such as the Anonymity Bibliography[12].

Conclusion

Tor is a well designed tool for the provisioning of anonymous information access and sharing across networks. An increasingly popular tool, it is run on a range of servers and clients across the world, providing the means for circumvention of online censorship, and the power to evade or complicate internet access monitoring tools.

While it attracts its share of attackers and criticism, it continues to be the best tool for anonymity and free information access on the internet. As such, it can be expected to be around for some time to come.

References

1. R. Dingledine, N. Mathewson, P. Syverson, "Tor: The Second-Generation Onion Router." Tor: Anonymity Online, 2004, Electronic Frontier Foundation, March 2007, < http://tor.eff.org/tor-design.pdf>
2. D. Chaum, "Untraceable Electronic Mail, Return Addresses, and Digital Pseudonyms", CACM 4(2), February 1981.
3. U. Moller, L. Cottrell, P. Palfrader, L. Sassaman, "Mixmaster Protocol, Version 2." July 2003, Abditum.com, March 19th 2007, <www.abditum.com/mixmaster-spec.txt>
4. G. Danezis, R. Dingledine, N. Mathewson, "Mixminion: Design of a Type III Anonymous Remailer Protocol." Proceedings of the 2003 IEEE Symposium on Security and Privacy, 2003
5. W. Dai, "Pipenet", April 15, 2007, <www.weidai.com/pipenet.txt>
6. Goldberg, I, "Privacy-enhancing technologies for the Internet II, Five Years Later." Proceedings of the Workshop on Privacy Enhancing Technologies 2002, 2002
7. D. Goldshlag, M. Reed, P. Syverson, "Hiding Routing Information." Information Hiding, 1996, pp. 137-150
8. K. Bauer, D. McCoy, D. Grunwald, T. Kohno, D. Sicker, "Low-Resource Routing Attacks Against Anonymous Systems." Technical Report CU-CS-1025-07, University of Colorado at Boulder, February 2007
9. I. Clark, O. Sandberg, B. Wiley, T. Hong, "Freenet: A Distributed Anonymous Information Storage and Retrieval System." Lecture Notes in Computer Science volume 2009, 2001, pp 46
10. M. Reiter, A. Rubin, "Crowds: Anonymity for Web Transactions." ACM Transactions on Information and System Security," 1998, pp. 66-92
11. B. Levine, C. Shields, "Hordes: A multicast-based protocol for anonymity." Journal of Computer Security, Volume 10, 2003, pp 213-242
12. Anonymity Bibliography, Freehaven, March 2007, <http://freehaven.net/anonbib/>

Resources

- Fu, Xinwen, B. Graham, R. Bettati, Wei Zhao, "On Effectiveness of Link Padding for Statistical Traffic Analysis Attacks." Proceedings of the 23rd International Conference on Distributed Computing Systems, 2003, pp. 340-347
- Sun, Qixiang, Daniel R. Simon, Yi-Min Wang, Wilf Russell, Venkata N. Padmanabhan, Lili Qiu, "Statistical Identification of Encrypted Web Browsing Traffic." Proceedings of the 2002 IEEE Symposium on Security and Privacy, May 2002, pp. 19
- Tor: anonymity online, March 18, 2007, Electronic Frontier Foundation, March 19th, 2007, <http://tor.eff.org>

About the author

Michael Owen is a senior security consultant for Information Risk Management plc, a security consultancy focusing on risk analysis and management as well as overall IT security management. Michael began his security work looking at military messaging systems and Common Criteria evaluations, and has since done work with online banks, online gambling institutions, and a range of transport and logistics firms. Anonymity and privacy have been major interests of his since he began working in IT and IT security in 1998.