Ten things to look for in a circumvention tool

Roger Dingledine
The Tor Project

Originally prepared for the March 2010 “Index on Censorship”,
then adapted for the July 2010 “China Rights Forum”

Contents

Introduction

As more countries crack down on Internet use, people around the world are turning to anti-censorship software that lets them reach blocked websites. Many types of software, also known as circumvention tools, have been created to answer the threat to freedom online. These tools provide different features and levels of security, and it's important for users to understand the tradeoffs.

This article lays out ten features you should consider when evaluating a circumvention tool. The goal isn't to advocate for any specific tool, but to point out what kind of tools are useful for different situations. I've chosen the order of features based on ease of presentation; so you shouldn't conclude the first feature is the most critical.

Internet-based circumvention software consists of two components: a *relaying* component and a *discovery* component. The relaying component is what establishes a connection to some server or proxy, handles encryption, and sends traffic back and forth. The discovery component is the step before that – the process of finding one or more reachable addresses.

Some tools have a simple relaying component. For example, if you're using an open proxy, the process of using the proxy is straightforward: you configure your web browser or other application to use the proxy. The big challenge for open proxy users is finding an open proxy that's reliable and fast. On the other hand, some tools have much more sophisticated relaying components, made up of multiple proxies, multiple layers of encryption, and so on.

One caveat to start out: I'm an inventor and developer of a tool called Tor that is used both for privacy and for circumvention. While my bias for more secure tools like Tor shows through here based on which features I've picked (meaning I raise issues that highlight Tor's strengths and that some other tool developers may not care about), I have also tried to include features that other tool developers consider important.

1 Has a diverse set of users

One of the simplest questions you can ask when looking at a circumvention tool is who else uses it. A wide variety of users means that if somebody finds out you are using the software, they can't conclude much about why you're using it. A privacy tool like Tor has many different classes of users around the world (ranging from ordinary people and human rights activists to corporations, law enforcement, and militaries) so the fact that you have Tor installed doesn't give people much additional information about who you are or what sorts of sites you might visit. On the other hand, imagine a group of Iranian bloggers using a circumvention tool created just for them. If anybody discovers that one of them is using it, they can easily guess why.

Beyond technical features that make a given tool useful to a few people in one country or people all over the world, marketing plays a big role in which users show up. A lot of tools spread through word of mouth, so if the first few users are in Vietnam and they find it useful, the next users will tend to be from Vietnam too. Whether a tool is translated into some languages but not others can also direct (or hamper) which users it will attract.

2 Works in your country

The next question to consider is whether the tool operator artificially restricts which countries can use it. For several years, the commercial Anonymizer.com made its service free to people in Iran. Thus connections coming from Anonymizer's servers were either paying customers (mostly in America) or people in Iran trying to get around their country's filters. For more recent examples, Your Freedom restricts free usage to a few countries like Burma, while at times systems like Freegate and Ultrasurf outright block connections from all but the few countries that they care to serve (China and, in the case of Ultrasurf recently, Iran). On the one hand, this strategy makes sense in terms of limiting the bandwidth costs. But on the other hand, if you're in Saudi Arabia and need a circumvention tool, some otherwise useful tools are not an option for you.

3 Has a sustainable network and software development strategy

If you're going to invest the time to figure out how to use a given tool, you want to make sure it's going to be around for a while. There are several ways that different tools ensure their long-term existence. The main three approaches are the use of volunteers, making a profit, and getting funds from sponsors.

Networks like Tor rely on volunteers to provide the relays that make up the network. Thousands of people around the world have computers with good network connections and want to help make the world a better place. By joining them into one big network, Tor ensures that the network is independent from the organization writing the software; so the network will be around down the road even if The Tor Project as an entity ceases to exist. Psiphon takes the second approach: collecting money for service. They reason that if they can create a profitable company, then that company will be able to fund the network on an ongoing basis. The third approach is to rely on sponsors to pay for the bandwidth costs. The Java Anon Proxy or JAP project relied on government grants to fund its bandwidth; now that the grant has finished they're investigating the for-profit approach. Ultrareach and Freegate use the "sponsor" model to good effect, though they are constantly hunting for more sponsors to keep their network operational.

After asking about the long-term survival of the network, the next question to ask is about sustainability of the software itself. The same three approaches apply here, but the examples change. While Tor's network is operated by volunteers, Tor relies on sponsors (governments and NGOs) to fund new features and software maintenance. Ultrareach and Freegate, on the other hand, are in a more sustainable position with respect to software updates: they have a team of individuals around the world, mostly volunteers, devoted to making sure the tools are one step ahead of censors.

Each of the three approaches can work, but understanding the approach a tool uses can help you predict what problems it might encounter in the future.

4 Has an open design

The first step to transparency and reusability of the tool's software and design is to distribute the software (not just the client-side software, but also the server-side software) under an open source license. Open source licenses mean that you can examine the software to see how it really operates, and you have the right to modify the program. Even if not every user takes advantage of this opportunity (many people just want to use the tool as-is), the fact that some users can makes it much more likely that the tool will remain safe and useful. Without this option, you are forced to trust that a small number of developers have thought of and addressed every possible problem.

Just having an open source license is not enough. Trustworthy circumvention tools need to provide clear, complete documentation for other security experts – not just how it's built but what features and goals its developers aimed for. Do they intend for it to provide privacy? What kind and against what attackers? In what way does it use encryption? Do they intend for it to stand up to attacks from censors? What kind of attacks do they expect to resist and why will their tool resist them? Without seeing the source code *and* knowing what the developers meant for it to do, it's harder to decide whether there are security problems in the tool, or to evaluate whether it will reach its goals.

In the field of cryptography, Kerckhoffs' principle explains that you should design your system so the amount you need to keep secret is as small and well-understood as possible. That's why crypto algorithms have keys (the secret part) and the rest can be explained in public to anybody. Historically, any crypto design that has a lot of secret parts has turned out to be less safe than its designers thought. Similarly, in the case of secret designs for circumvention tools, the only groups examining the tool are its original developers and its attackers; other developers and users who could help to make it better and more sustainable are left out.

Ideas from one project could be reusable beyond that project's lifetime. Too many circumvention tools keep their designs secret, hoping that government censors will have a harder time figuring out how the system works, but the result is that few projects can learn from other projects and the field of circumvention development as a whole moves forward too slowly.

5 Has a decentralized architecture

Another feature to look for in a circumvention tool is whether its network is centralized or decentralized. A centralized tool puts all of its users' requests through one or a few servers that the tool operator controls. A decentralized design like Tor or JAP sends the traffic through multiple different locations, so there is no single location or entity that gets to watch what websites each user is accessing.

Another way to look at this division is based on whether the *trust* is centralized or decentralized. If you have to put all your trust in one entity, then the best you can hope for is "privacy by policy" – meaning they have all your data and they promise not to look at it, lose it, or sell it. The alternative is "privacy by design" or "privacy by architecture" – meaning the design of the system itself ensures that users get their privacy. The openness of the design in turn lets everybody evaluate the level of privacy provided.

This concern isn't just theoretical. In early 2009 Hal Roberts from the Berkman Center ran across a FAQ entry for a circumvention tool that offered to sell its users' clicklogs. I later talked to a different circumvention tool provider who explained that they had all the logs of every request ever made through their system "because you never know when you might want them."

I've left out the names of the tools here because the point is not that some tool providers may have shared user data; the point is that any tool with a centralized trust architecture *could* share user data, and its users have no way to tell whether it's happening. Worse, even if the tool provider means well, the fact that all the data flows through one location creates an attractive target for other attackers to come snooping.

Many of these tools see circumvention and user privacy as totally unrelated goals. This separation isn't necessarily bad, as long as you know what you're getting into – for example, we hear from many people in censoring countries that just reading a news website isn't going to get you locked up. But as we've been learning in many other contexts over the past few years, large databases of personal information tend to end up more public than we'd like.

6 Keeps you safe from websites too

Privacy isn't only about whether the tool operator can log your requests. It's also about whether the websites you visit can recognize or track you. Remember the case of Yahoo turning over information about one of its Chinese webmail users? What if a blog aggregator wants to find out who's posting to a blog, or who added the latest comment, or what other websites a particular blogger reads? Using a safer tool to reach the website means the website won't have as much to hand over.

Some circumvention tools are safer than others. At one extreme are open proxies. They often pass along the address of the client with their web request, so it's easy for the website to learn exactly where the request is coming from. At the other extreme are tools like Tor that include client-side browser extensions to hide your browser version, language preference, browser window size, time zone, and so on; segregate cookies, history, and cache; and prevent plugins like Flash from leaking information about you.

This level of application-level protection comes at a cost though: some websites don't work correctly. As more websites move to the latest "web 2.0" fads, they require more and more invasive features with respect to browser behavior. The safest answer is to disable the dangerous behaviors – but if somebody in Turkey is trying to reach Youtube and Tor disables his Flash plugin to keep him safe, his videos won't work.

No tools have solved this tradeoff well yet. Psiphon manually evaluates each website, and programs its central proxy to rewrite each page. Mostly they do this rewriting not for privacy but to make sure all links on the page lead back to their proxy service, but the result is that if they haven't manually vetted your destination site yet, it probably won't work for you. As an example, they seem to be in a constant battle to keep up with Facebook's changing frontpage. Tor currently disables some content that is probably safe in practice, because we haven't figured out a good interface to let the user decide in an informed way. Still other tools just let through any active content, meaning it's trivial to unmask their users.

7 Doesn't promise to magically encrypt the entire Internet

I should draw a distinction here between encryption and privacy. Most circumvention tools (all but the really simple ones like open proxies) encrypt the traffic between the user and the circumvention provider. They need this encryption to avoid the keyword filtering done by such censors as China's firewall. But none of the tools can encrypt the traffic between the provider and the destination websites – if a destination website doesn't support encryption, there's no magic way to make the traffic encrypted.

The ideal answer would be for everybody to use https (also known as SSL) when accessing websites, and for all websites to support https connections. When used correctly, https provides encryption between your web browser and the website. This "end-to-end" encryption means nobody on the network (not your ISP, not the backbone Internet providers, and not your circumvention provider) can listen in on the contents of your communication. But for a wide variety of reasons, pervasive encryption hasn't taken off. If the destination website doesn't support encryption, the best you can do is 1) not send identifying or sensitive information, such as a real name in a blog post or a password you don't want other people to learn, and then 2) use a circumvention tool that doesn't have any trust bottlenecks that allow somebody to link you to your destinations despite the precautions in step 1.

Alas, things get messy when you can't avoid sending sensitive info. Some people have expressed concern over Tor's volunteer-run network design, reasoning that at least with the centralized designs you know who runs the infrastructure. But in practice it's going to be strangers reading your traffic either way – the tradeoff is between volunteer strangers who don't know it's you (meaning they can't target you), or dedicated strangers who get to see your entire traffic profile (and link you to it). Anybody who promises "100% security" is selling something.

8 Provides consistently good latency and throughput

The next feature you might look for in a circumvention tool is speed. Some tools tend to be consistently fast, some consistently slow, and some provide wildly unpredictable performance. Speed is based on many factors, including how many users the system has, what the users are doing, how much capacity there is, and whether the load is spread evenly over the network.

The centralized-trust designs have two advantages here. First, they can see all their users and what they're doing, meaning they have a head start at spreading them out evenly and at discouraging behaviors that tax the system. Second, they can buy more capacity as needed, so the more they pay the faster the tool is. The distributed-trust designs on the other hand have a harder time tracking their users, and if they rely on volunteers to provide capacity, then getting more volunteers is a more complex process than just paying for more bandwidth.

The flip side of the performance question is flexibility. Many systems ensure good speed by limiting what their users can do. While Psiphon prevents you from reaching sites that they haven't manually vetted, Ultrareach and Freegate actually actively censor which destination websites you're allowed to reach so they can keep their bandwidth costs down. Tor, by contrast, lets you access any protocol and destination, meaning for example you can instant message through it too; but the downside is that the network is often overwhelmed by users doing bulk transfer.

9 Makes it easy to get the software and updates

Once a circumvention tool becomes well-known, its website is going to get blocked. If it's impossible to get a copy of the tool itself, who cares how good it is? The best answer here is to not require any specialized client software. Psiphon, for example, relies on a normal web browser, so it doesn't matter if the censors block their website. Another approach is a tiny program like Ultrareach or Freegate that you can instant message to your friends. Option three is Tor's Browser Bundle: it comes with all the software you need preconfigured, but since it includes large programs like Firefox it's harder to pass around online. In that case distribution tends to be done through social networks and USB sticks, or using our email autoresponder that lets you download Tor via Gmail.

Then you need to consider the tradeoffs that come with each approach. First, which operating systems are supported? Psiphon does well here too by not requiring any extra client software.

Ultrareach and Freegate are so specialized that they only work on Windows, whereas Tor and its accompanying software can run pretty much everywhere. Next, consider that client-side software can automatically handle failover from one proxy to the next, so you don't need to manually type in a new address if your current address disappears or gets blocked.

Last, does the tool have a track record for responding to blocking? For example, Ultrasurf and Freegate have a history of releasing quick updates when the current version of their tool stops working. They have a lot of experience at this particular cat-and-mouse game, so it's reasonable to assume they're ready for the next round. Along these lines, Tor prepared for its eventual blocking by streamlining its network communications to look more like encrypted web browsing, and introducing unpublished "bridge relays" that are harder for an attacker to find and block than Tor's public relays. Tor tries to separate software updates from proxy address updates – if the bridge relay you're using gets blocked, you can stick with the same software and just configure it to use a new bridge address. Our bridge design was put to the test in China in September of 2009, and tens of thousands of users seamlessly moved from the public relays to bridges.

10 Doesn't promote itself as a circumvention tool

Many circumvention tools launch with a huge media splash. The media loves this approach, and they end up with front page articles like "American hackers declare war on China!" But while this attention helps attract support (volunteers, profit, sponsors), the publicity also attracts the attention of the censors.

Censors generally block two categories of tools: 1) the ones that are working really well, meaning they have hundreds of thousands of users, and 2) the ones that make a lot of noise. In many cases censorship is less about blocking all sensitive content and more about creating an atmosphere of repression so people end up self-censoring. Articles in the press threaten the censors' *appearance* of control, so they are forced to respond.

The lesson here is that we can control the pace of the arms race. Counterintuitively, even if a tool has many users, as long as nobody talks about it much it tends not to get blocked. But if nobody talks about it, how do users learn about it? One way out of the paradox is to spread through word of mouth and social networks rather than the more traditional media. Another approach is to position the tool in a different context – for example, we present Tor primarily as a privacy and civil liberties tool rather than a circumvention tool. Alas, this balancing act is tough to maintain in the face of increasing popularity.

Conclusion

This article explains some of the issues you should consider when evaluating the strengths and weaknesses of circumvention tools. I've intentionally avoided drawing up a table of different tools and scoring them on each category. No doubt somebody will do that eventually and sum up how many checkmarks each tool gets, but the point here is not to find the "best" tool. Having a diversity of circumvention tools in wide use increases robustness for all the tools, since censors have to tackle every strategy at once.

Last, we should keep in mind that technology won't solve the whole problem. After all, firewalls are *socially* very successful in these countries. As long as many people in censored countries are saying "I'm so glad my government keeps me safe on the Internet," the social challenges are at least as important. But at the same time, there are people in all of these countries who want to learn and spread information online, and a strong technical solution remains a critical piece of the puzzle.

About the author

Roger Dingledine is project leader for The Tor Project, a US non-profit working on anonymity research and development for such diverse organizations as the US Navy, the Electronic Frontier Foundation, and Voice of America. In addition to all the hats he wears for Tor, Roger organizes academic conferences on anonymity, speaks at a wide variety of industry and hacker conferences, and also does tutorials on anonymity for national and foreign law enforcement.