What Is A Search Engine?

A search engine is a complex software system that searches the web to find web pages that answer users' search queries. The search results (SERPs) are presented in order of importance and relevance to what the user is looking for.

Modern search engines include different types of content in their results, including articles, videos, images, forum postings, and social media posts.

The **most popular search engine** is Google, with over 90% market share, followed by Bing, DuckDuckGo, and others.

How Do Search Engines Work

Search engines work by crawling publicly available pages using web crawlers. Web crawlers (aka spiders or bots) are special programs that crawl the web to find new pages or updates to existing pages and add this information to a search index.

This process is broken down into three main stages:

- The first stage is the process of discovering the information.
- The second stage is organizing the information.
- The third stage is deciding which pages to show in the results for a search query and in what order.

This is generally known as **crawling**, **indexing**, and **ranking**.

1. Crawling

During the crawling process, the goal of search engines is to find information that is publicly available on the Internet. This includes new content or updates made to existing content. They do these using a number of software programs called crawlers.

To simplify a complicated process, it's enough for you to know that the job of crawlers, is to scan the Internet and find the servers (also known as webservers) hosting websites.

They create a list of all the web servers and the number of websites hosted by each server.

They visit each website and use different techniques to find out how many pages they have and the type of content on each page (text, images, videos, etc). When visiting a webpage, they also follow any links (either pointing to pages within the site or to external websites), to discover more and more pages.

They do this continuously and keep track of changes made to a website so that they know when new pages are added or deleted, when links are updated, etc. If you consider that there are more than 130 trillion individual pages on the Internet today, you can imagine that this is a lot of work.

Why care about the crawling process?

Your first concern when optimizing your website for search engines is to ensure that they can access it correctly otherwise, if they cannot 'read' your website, you shouldn't expect much in terms of high rankings or search engine traffic.

As explained above, crawlers have a lot of work to do, and you should try and make their job easier.

There are a number of things to do to make sure that crawlers can discover and access your website in the fastest possible way without problems.

1. Use **Robots.txt** to specify which pages of your website you don't want crawlers to access. For example, pages like your admin or backend pages and other pages you don't want to be publicly available on the Internet.
2. Big search engines like Google and Bing have tools you can use to give them more information about your website (number of pages, structure, etc) so that they don't have to find it themselves.
3. Use an **XML sitemap** to list all important pages of your website so that the crawlers can know which pages to monitor for changes.
4. Use the "noindex" tag to instruct search engine crawlers not to index a particular page.

2. Indexing

Crawling alone is not enough to build a search engine. Information identified by the crawlers needs to be organized, sorted, and stored so that the search engine algorithms can process it before being made available to the end-user.

This process is called Indexing. Search engines don't store all the information found on a page in their index, but they keep things like when it was created/updated, title and description of the page, type of content, associated keywords, incoming and outgoing links, and a lot of other parameters that are needed by their algorithms.

Why care about the indexing process?

It's very simple, if your website is not in their index, it will not appear for any searches.

This also implies that the more pages you have in the search engine indexes, the more your chances of appearing in the search results when someone types a query.

How to find how many pages of your website are included in the Google index?

There are two ways to do that.

Open Google and use the *site* operator followed by your domain name. For example, *site:reliablesoft.net.* You will find out how many pages related to the particular domain are included in the Google Index.

The second way is to create a free Google Search Console account and **add your website**.

Then look at the *Indexed Pages* report located under *Pages* > *Indexing*.

3. Ranking

The third and final step in the process is for search engines to decide which pages to show in the **Search Engine Results Pages** (SERPS) and in what order when someone types a query. This is called the ranking process and is achieved through the use of search engine ranking algorithms.

In simple terms, these are pieces of software that use a number of rules to decide which are the best results for a search query.

How Do Search Engine Algorithms Work?

Search engine algorithms examine several factors and signals to find the best match for a user query. This includes looking at the relevancy of the content to the words typed by the user, the usability of a page, the user's location, what other users found useful for the particular query, and many other factors.

It's important to mention that search engine ranking algorithms have become really complex over the years. In the beginning (think 2001), it was as simple as matching the user's query with the page's title, but this is no longer the case.

Google's ranking algorithm considers more than 255 rules before making a decision, and nobody knows for sure what these rules are. Search engines use machine learning and AI to make decisions based on parameters inside and outside the boundaries of the content found on a web page.

To make it easier to understand, here is a simplified process of how search engine ranking factors work:

Step 1: Analyze User Query

The first step is for search engines to understand what kind of information the user is looking for.

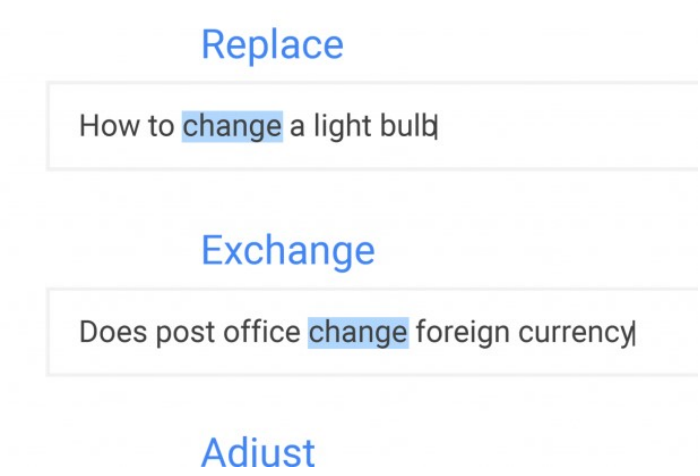
To do that, they analyze the user's query (search terms) by breaking it down into a number of meaningful keywords.

A keyword is a word that has a specific meaning and purpose.

For example, when you type "How to make a chocolate cake", search engines know from the words *how-to* that you are looking for instructions on how to make a chocolate cake, and thus the returned results will contain cooking websites with recipes.

If you search for "Buy refurbished ….", they know from the words *buy* and *refurbished* that you are looking to buy something, and the returned results will include eCommerce websites and online shops.

Machine learning has helped them associate related keywords together. For example, they know that the meaning of the query "how to change a light bulb" is the same as this "how to replace a light bulb".



They are also clever enough to interpret spelling mistakes, understand plurals, and extract the meaning of a query from natural language (either written or verbal in case of Voice search).

Step 2: Finding matching pages

The second step is to look into their index and decide which pages can provide the best answer for a given query.

This is a very important stage in the whole process for both search engines and webmasters. Search engines need to return the best possible results in the fastest possible way so that they keep their users happy. Webmasters want their websites to be picked up so that they get traffic and visits.

This is also the stage where good **SEO techniques** can influence the decisions made by the algorithms.

To give you an idea of how *matching* works, these are the most critical factors:

Title and content relevancy – how relevant are the title and content of the page to the user query?

Type of content – if the user asks for images, the returned results will contain images, not text.

Quality of the content – content needs to be thorough, **useful and informative**, unbiased, and cover both sites of a story.

Quality of the website – The overall quality of a website matters. Google will not show pages from websites that don't meet their quality standards.

Date of publication – For news-related queries, Google wants to show the latest results, so the publication date is also considered.

The popularity of a page – This doesn't have to do with how much traffic a website has but how other websites perceive the particular page. A page that has a lot of references (**backlinks**) from other websites is considered to be more popular than other pages with no links.

Language of the page – Users are served pages in their language, and it's not always English.

Webpage Speed – Websites that load fast (think 2-3 seconds) have a small advantage compared to websites that are slow to load.

Device Type – Users searching on mobile are served mobile-friendly pages.

Location – Users searching for results in their area, i.e., "Italian restaurants in Ohio," will be shown results related to their location.

That's just the tip of the iceberg. As mentioned before, Google uses more than 255 factors in its algorithms to ensure that its users are happy with the results they get.

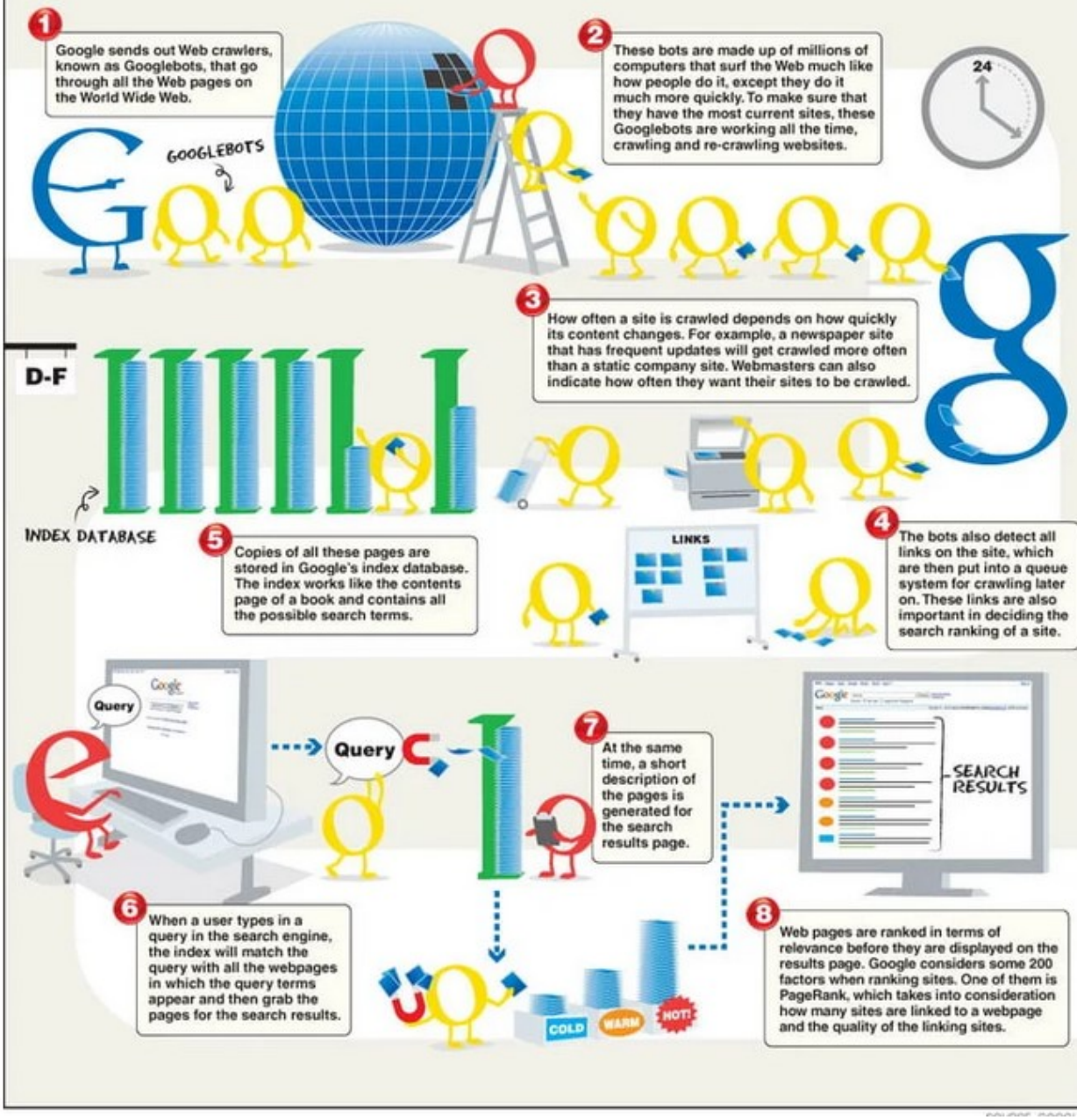
Step 3: Present the results to the users

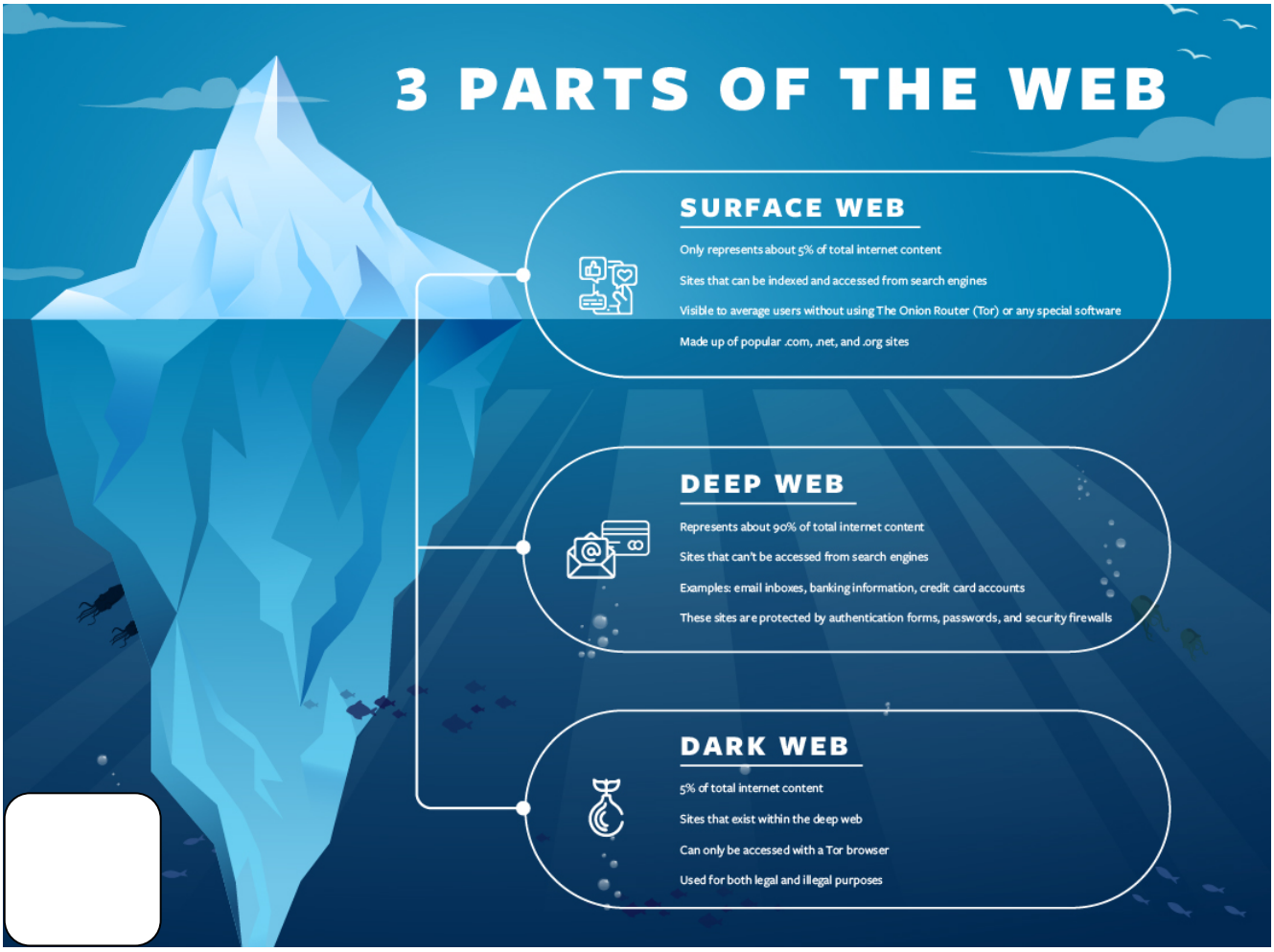
The search results, typically known as the **Search Engine Results Pages** (SERPs), are presented in an ordered list. The layout of SERPs often includes various elements such as organic listings, paid ads, featured snippets, knowledge graphs, rich snippets, and more, depending on the nature of the query.

For example, a search for a specific news item might bring up recent news articles, while a query for a local restaurant could display a map with nearby locations.

HOW GOOGLE SEARCH WORKS

Have you ever wondered what happens when you type in a query in Google's search field? Tham Yuen-C and Quek Hong Shin go behind the scenes of the search engine





The Dark Web - Online content that isn't indexed by search engines belongs to what has come to be called the "Deep Web"—that is, content on the World Wide Web that is hidden. Any site that suppresses Web crawlers from indexing it is part of the Deep Web. You need go no further than the world's number two Web site, Facebook, for a prominent example of a site that isn't indexed in search engines.

Entire networks exist that aren't searchable, particularly peer-to-peer networks.

The Deep Web includes:

- Database-generated Web pages or dynamic content
- Pages without links
- Private or limited access Web pages and sites
- Information contained in sources available through executable code such as JavaScript
- Documents and files that aren't in a form that can be searched, which includes not only media files, but information in non-standard file formats.

Although efforts are underway to enable information on the Deep Web to be searchable, the amount of information stored that is not accessible is many times larger than the amount of information that can currently be accessed. Some estimates at the size of the Dark Web suggest that it could be an order of magnitude larger than the content contained in the world's search engines. It is always a good idea to keep these search engine limitations in mind when you work with this technology.

The Deep Web - Unlike the dark web, which is deliberately obscured by layers of technology, the deep web exists just **below the surface web**. The majority of the deep web consists of regular websites that require users to create an account before they can be accessed.

If the surface web is the visible part of an iceberg above water, the deep web is the part submerged beneath – much larger but hidden from plain view. Some people use the terms 'dark web' and 'deep web' interchangeably; however, significant portions of the deep web are completely legal and safe to use.

A majority of the web consists of databases and intranets. The former includes privately protected file collections that one cannot access without the correct credentials. The latter includes internal networks for governments, educational facilities, and corporations. The 'members-only' parts of public platforms are also a part of the deep web.

All web pages not indexed by web crawlers are considered to be a part of the deep web. The content is generally more secure and clean than that of the surface web. This is because deep web content is usually well-maintained. Security tools such as firewalls help in this endeavor.

Simple examples of deep web content include financial data, social security databases, email inboxes, social media, medical documentation, legal files, blog posts that are pending review and web page redesigns that are in progress. These pages are mostly obscured from the surface web to secure user data and privacy rather than any nefarious purpose.

However, the deep web is not entirely devoid of danger. Some portions of the deep web allow users to overcome legal restrictions to access content that is not lawfully available in their geographical location. It is even possible to illegally download movies, music, and other digital media without paying for it. Naturally, these lawless segments of the deep web are rife with malware and other cyber threats.

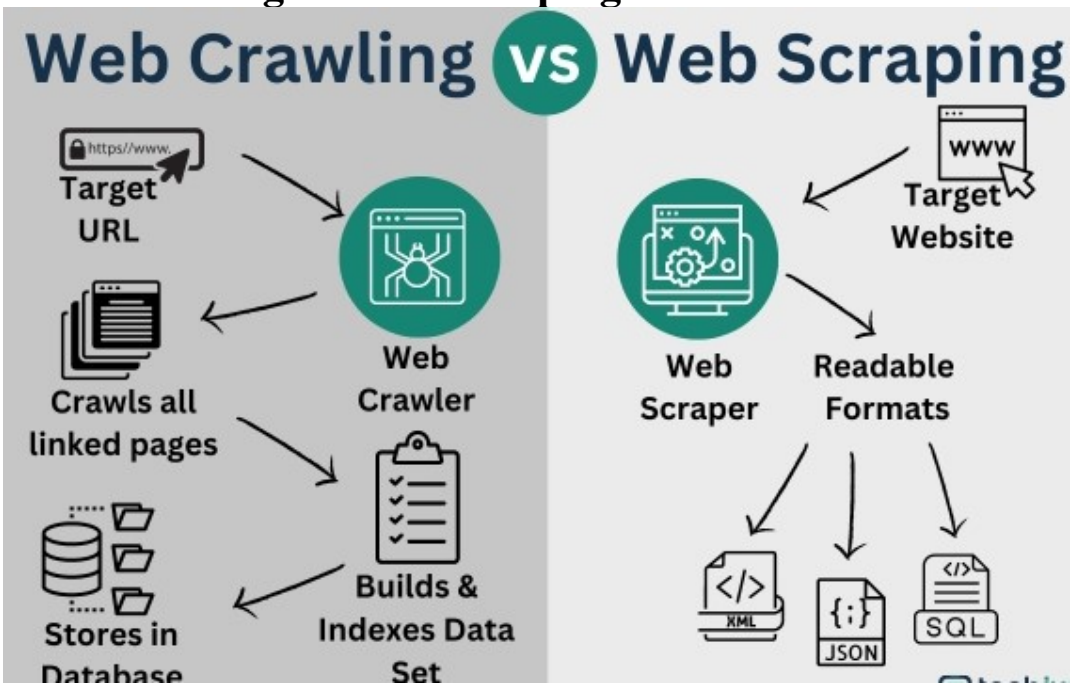
While the content on the dark web has the potential to be more dangerous, this content is usually walled off from regular users. However, it is entirely possible for regular users to accidentally come across harmful content while browsing the deep web, which is much more easily accessible. This makes deep web security important for individuals and enterprises alike.

Deep Web vs Dark Web

For better clarity, let's go through the difference between deep web and dark web in a tabular format:

Aspect	Deep Web	Dark Web
Definition	Deep web refers to all web content not indexed by standard search engines, which includes both legal and hidden content.	The dark web is a secretive part of the deep web known for hosting illegal activities and can only be accessed through specialized tools.
Accessibility	Accessible through regular web browsers	Requires specific browsers like Tor
Content	Legal and legitimate content	Often associated with illegal activities
Anonymity	Users are typically not anonymous	Provides a high level of anonymity
Web Pages	Hidden from public view	Intentionally hidden and encrypted
Legal Uses	Includes private databases,password-protected sites, and intranet resources.	Used by whistleblowers, activists, and journalists for secure communication and bypassing censorship.
Illicit Uses	May contain legal and illegal content such as sensitive databases and academic research.	Known for hosting illegal activities like black market transactions and drug trafficking.

Web Crawling vs Web Scraping



Main differences between web scraping and web crawling

As mentioned earlier, data crawling is for collecting links, and data scraping is for collecting data. Usually, they are used together: first, the crawler collects links, then the scraper processes them and collects the necessary data.

Let's compare web crawling and web scraping:

Category	Web Scraping	Web Crawling
Definition	Extracting data from websites using software or web scraping tools that simulate human browsing behavior.	Automated process of browsing the internet, indexing and cataloging information, and organizing it in a database.
Purpose	Extract specific data from web pages for analysis and use	Explore and index web pages for search engines and data analysis
Data Extraction Method	Pulls data from a website's HTML structure.	Crawls web pages and follows hyperlinks to find new pages to index.
Scope	Limited to specific websites or pages.	It can cover a broader range of websites or pages.
Targeted Sites	Generally targeted at specific websites or pages.	It can cover a wide range of websites or pages.
Data Analysis	Data can be extracted in an easily analyzed format and used for various purposes, such as price comparison or marketing research.	Crawler does not perform data analysis
Scalability	Done on both small and large scales	Mostly employed on a large scales
Process Requirements	Only requires a crawler to request and retrieve web pages	Requires both a crawler to retrieve web pages and a parser to extract the relevant data from those pages

Thus, the key difference between web crawling and web scraping is that web crawling involves automatically visiting every page it can find, while web scraping focuses on extracting specific data from a website. Web crawlers explore the entire internet by following links between pages, whereas web scrapers only target information related to their queries.

Use cases of web crawling and web scraping

Web scraping and web scanning are two distinct processes that can be used in tandem to significant effect. However, they can also be utilized independently, depending on the task.

Web crawling use cases - Web crawling, as mentioned earlier, is great for projects that require collecting links, may not have targeted resources, and need to retrieve entire page code without further parsing and processing. Here are some of the most common use cases:

Search Engines Indexing

Search engines like Google, Bing, and Yahoo use their teams of crawlers to find newly updated content or new pages. The searchers then store the information they find in an index, a massive database of all the content they find that they think is good enough to provide to users.

Improving the performance of your own site

Web crawling is a valuable tool for analyzing your website and improving its performance. By running a web crawler, you can detect broken links or images, duplicate content or meta tags, and other issues that could negatively affect your site's SEO performance. Web crawling also enables you to identify opportunities for optimization of the site's structure as a whole.

Analyzing competitors' websites for SEO purposes

You can also monitor changes not only on your website but also on your competitors. This will ensure that you are always aware of new changes from your competitors and can react to them successfully and quickly.

Data mining

Web crawlers can be used to collect and analyze large amounts of data from various sources on the internet. This makes it easier for researchers, businesses, or any other interested parties to gain valuable insights about a particular topic – allowing them to make informed decisions based on said findings.

Finding broken links on external sites

In addition to checking your pages, keeping all links on external sites up-to-date is important. While bugs on your pages can usually be found on the admin panel, finding broken links to external sites are much more difficult. To keep them up to date, you must either check them manually or use crawlers.

Content curation

Crawlers are also great at finding content-related topics quickly and efficiently, allowing businesses or individuals to curate them into collections based on specific criteria such as keywords or tags.

Web scraping use cases - On the other hand, web scraping is great when you already know what specific information you want to extract from a website. Some of its most common applications are:

1. Price Monitoring
2. Content Aggregation
3. Lead Generation
4. Competitor Analysis
5. Social Media Analysis
6. Online Reputation Management

For more information on web scraping and its use cases, check out our article, which details some of the most common applications.

Use Case	Web Crawling	Web Scraping
Price Monitoring	✔ Can collect prices from multiple websites and track changes over time	✔ Can extract prices from specific web pages to compare and analyze
Job Aggregation	✔ Can collect job listings from various sites and present them in a centralized manner	✔ Can extract job listings from specific websites and present them in a centralized manner
Content Aggregation	✔ Can collect content from multiple websites to create a centralized repository	✘ Not suitable, as it requires extracting specific information from each webpage
SEO Optimization	✔ Can analyze website structure and backlinks for search engine optimization	✘ Not suitable, as it requires access to specific website data such as keyword density and metadata
News Monitoring	✔ Can monitor news outlets and RSS feeds for updates	✘ Not suitable, as it requires extracting specific information from each news article
Product Review Aggregation	✔ Can collect reviews from multiple websites and analyze sentiment	✔ Can extract reviews from specific websites and analyze sentiment
Data Mining	✔ Can collect large amounts of data for analysis and modeling	✔ Can extract specific data points for analysis and modeling
Finding Broken Links on External Sites	✔ Can crawl external sites to identify broken links	✘ Not suitable, as it requires extracting specific information from each webpage
Lead Generation	✔ Can collect contact information from multiple websites	✔ Can extract contact information from specific websites
Competitor Analysis	✔ Can gather data on competitor websites for analysis	✔ Can extract specific data points on competitor websites for analysis
Online Reputation Management	✔ Can monitor online mentions and reviews for reputation management	✘ Not suitable, as it requires extracting specific information from each webpage

Aggregation and disintermediation

Aggregation pages are a great user service, but they are very controversial—as are a number of Google's search applications and services. It has long been argued that Google's display of information from various sites violates copyright laws and damages content providers. In several lawsuits, Google successfully defended its right to display capsule information under the Digital Millennium Copyright Act, while in other instances Google responds to requests from interested parties to remove information from its site. The Authors Guild's filed a class action suit in 2005 regarding unauthorized scanning and copying of books for the creation of the Google Books feature. Google reached a negotiated agreement with the Authors Guild

that specified Google's obligations under the fair use exemption. Google argues that the publicity associated with searchable content adds value to that content, and it is clear that this is an argument that will continue into the future.

What is clear is that Google has been a major factor in a trend referred to as disintermediation. Disintermediation is the removal of intermediaries such as a distributor, agent, broker, or some similar functionary from a supply chain. This connects producers directly with consumers, which in many cases is a very good thing. However, disintermediation also has the unfortunate side effect of impacting organizations such as news collection agencies (newspapers, for example), publishers, many different types of retail outlets, and many other businesses, some of which played a positive role in the transactions they were involved in.

Google began to introduce productivity applications starting in 2004 with Gmail. The expansion of these services has continued unabated ever since. Some of these applications are homegrown, but many of them were acquired by acquisition.

AdWords

AdWords (http://www.google.com/AdWords) is a targeted ad service based on matching advertisers and their keywords to users and their search profiles. This service transformed Google from a competent search engine into an industry giant and is responsible for the majority of Google's revenue stream. AdWords' two largest competitors are Microsoft adcenter (http:// adcenter.microsoft.com/) and Yahoo! Search Marketing (http://searchmarketing. yahoo.com/).

Ads are displayed as text, banners, or media and can be tailored based on geographical location, frequency, IP addresses, and other factors. AdWords ads can appear not only on Google.com, but on AOL search, Ask.com, and Netscape, along with other partners. Other partners belonging to the Google Display Network can also display AdSense ads. In all these cases, the AdWords system determines which ads to match to the user searches.

Here's how the system works: Advertisers bid on keywords that are used to match a user to their product or service. If a user searches for a term such as "develop abdominal muscles," Google returns products based on those terms. You might see an ad with Chuck Norris selling a modern day version of a torture rack that, if it doesn't give you a six-pack, at least makes your wallet lighter. Up to 12 ads per search can be returned.

Google gets paid for the ad whenever a user clicks it. The system is referred to as pay-per-click advertising, and the success of the ad is measured by what is called the click-through rate (CTR). Google calculates a *quality score* for ads based on the CTR, the strength of the connection between the ad and the keywords, and the advertiser's history with Google. This quality score is a Google trade secret and is used to price the minimum bid of a keyword.

In 2007, Google purchased DoubleClick, an Internet advertising services company. DoubleClick helps clients create ads, provides hosting services, and tracks results for analysis. DoubleClick ads leave browser cookies on systems that collect information from users that determine the number of times a user has been exposed to a particular ad, as well as various system characteristics. Some spyware trackers flag DoubleClick cookies as spyware. Both AdWords and DoubleClick are sold as packages to large clients.

Google Analytics

Google Analytics (GA; http://google.com/analytics) is a statistical tool that measures the number and types of visitors to a Web site and how the Web site is used. It is offered as a free service and has been adopted by many Web sites. GA is built on the Urchin 5 analytical package that Google acquired in 2006.

According to Builtwith.com (http://trends.builtwith.com/analytics/Google-Analytics), Google Analytics was in use on 54 percent of the top 10,000 and 100,000, and 35 percent of the top one million of the world's Web sites. Builtwith.com speculates that Google Analytics JavaScript tag is the most widely used URL in the world today.

The service BackendBattles.com (http://www.backendbattles.com/backend/Google_Analytics) sets GA's market share at 57 percent for the top 10,000 sites.

Analytics works by using a JavaScript snippet called the Google Analytics Tracking Code (GATC) on individual pages to implement a page tag. When the page loads, the JavaScript runs and creates a first-party browser cookie that can be used to manage return visitors, perform tracking, test browser characteristics, and request tracking code that identifies the location of the visitor. GATC requests and stores information from the user's account. The code stored on the user's system acts like a beacon and collects visitor data that it sends back to GA servers for processing.

Among the visitors that can be tracked are those that land from search engines; referral links in e-mail, documents, and Web pages; display ads; PPC networks; and some other sources. GA aggregates the data and presents the information in a visual form. GA also is connected to the AdWords system so it can track the performance of particular ads in different contexts. You can view referral location statistics and time spent on a page, and you can filter by visitor site. GA lets you save and store up to 50 individual site profiles, provided the site has less than 5 million page views per month. This restriction is lifted for an AdWords subscription.

GA cookies are blocked by a number of technologies, such as Firefox Adblock and NoScript or by turning off JavaScript execution in other browsers. You also can delete GA cookies manually or block them, which also defeats the system.

Google Translate

Of all the Google applications, the one that might have significant immediate impact is Google Translate. Computer technology is very close to having the necessary hardware and software to realize the dream of a "universal translator" that the TV show *Star Trek* proposed some 45 years ago. The current version of Google Translate performs machine translation as a cloud service between two of your choice of 35 different languages. That's not truly universal, but until aliens appear, it will do for most people.

Google Translate was introduced in 2007 and replaced the SYSTRAN system that many other computer services utilize. The translation method uses a statistical approach that was first developed by Franz-Joseph Och in 2003. Och now heads the Translate effort at Google.

Translate uses what is referred to as a corpus linguistics approach to translation. You start off building a translation system for a language pair by collecting a database of words and then matching that database to two bilingual text corpuses. A text corpus or parallel collection is a database of word- and phrase usage taken from the language in everyday use obtained by examining documents translated by professionals to software analysis. Among the documents that are analyzed are the translations of the United Nations and European Parliament, among others.
Google Translate can be accessed directly at http://translate.google.com/ translate_t?hl=en#, where you can select the language pair to be translated. You can do the following:

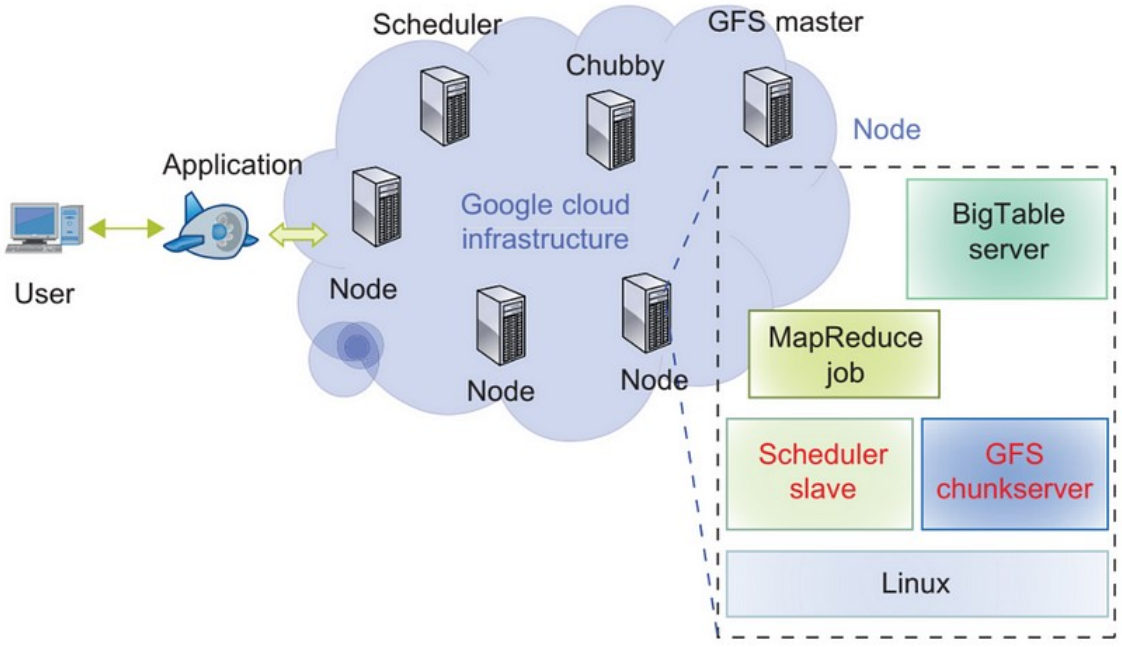
- Enter text directly into the text box, and click the Translate button to have the text translated. If you select the Detect Language option, Translate tries to determine the language automatically and translate it into English.
- Enter a URL for a Web page to have Google display a copy of the translated Web page.

- Enter a phonetic equivalent for script languages.
- Upload a document to the page to have it translated.

Translate parses the document into words and phrases and applies its statistical algorithm to make the translation. As the service ages, the translations are getting more accurate, and the engine is being added to browsers such as Google Chrome and through extension into Mozilla Firefox. The Google Toolbar offers page translation as one of its options, selectable in the Tools settings.

The Google Translator Toolkit (http://translate.google.com/toolkit) provides a means for using the Translate to perform translations that you can edit. The toolkit provides access to tools to aid you in editing the translation. Translation services have been in development for many years. IBM has had a large effort in this area, and the Microsoft Bing search engine also has a translation engine. There are many other translation engines, and some of them are even cloud-based like Google Translate. What makes Google's efforts potentially unique is the company's work in language transcription—that is, the conversion of voice to text. As part of Google Voice and its work with Android-based cell phones, Google is sampling and converting millions and millions of conversations. Combining these two Web services together could create a translation device based on a cloud service that would have great utility.

The Architecture of Google App Engine(GAE)



Google cloud platform and major building blocks, the blocks shown are large clusters of low-cost servers.
(Courtesy of Kang Chen, Tsinghua University, China)

Google App Engine is the the typical example of PaaS. Google App Engine is for developing and hosting web applications and these process are highly scalable. The applications are designed to serve a multitude of users simultaneously, without incurring a decline in overall performance. Third-party application providers can use GAE to build cloud applications for providing services. The applications run in data centers which is managed by Google engineers. Inside each data center, there are thousands of servers forming different clusters.

The building blocks of Google's cloud computing application include the Google File System, the MapReduce programming framework, and Big Table. With these building blocks, Google has built many cloud applications. The above Figure shows the overall architecture of the Google cloud infrastructure. GAE runs the user program on Google's infrastructure. As it is a platform running third-party programs, application developers now **do not need to worry about the maintenance of servers**. GAE can be thought of as the combination of several software components. The frontend is an application framework which is similar to other web application frameworks such as ASP, J2EE, and JSP. At the time of this writing, GAE supports Python and Java programming environments. The applications can run similar to web application containers. The frontend can be used as the dynamic web serving infrastructure which can provide the full support of common technologies.

Features of App Engine

1. **Popular language:** Users can build the application using language runtimes such as Java, Python, C#, Ruby, PHP etc.
2. **Open and flexible:** Custom runtimes allow users to bring any library and framework to App Engine by supplying a Docker container.
3. **Powerful application diagnostics:** Google App engine uses cloud monitoring and cloud logging to monitor the health and performance of the app and to diagnose and fix bugs quickly it uses cloud debugger and error reporting.
4. **Application versioning:** It easily hosts different versions of the app, and create development, test, staging, and production environments.

Google App Engine is one of the earliest PaaS Model which is fully scalable and simply a small example of how high frequency of requests can be efficiently handled. Despite the excellent architecture, there are too much restrictions, which makes the PaaS Model inclining towards the properties of a SaaS with too much Lock-in. For example, PHP can not run natively on the App Engine.

Tor (formerly an acronym for "The Onion Router") - is often touted as a way to browse the web anonymously. From human rights activists evading oppressive governments to drug dealers selling through online marketplaces, **Tor is a popular way to gain significantly more anonymity** than you would normally have online. At the same time, Tor isn't perfect, so it can provide a false sense of security if used incorrectly.

What does Tor do?

Using the Tor Browser is **similar to using any other web browser**. Although the process of starting up the browser differs slightly from Chrome or Firefox (Tor must configure a connection to the Tor network before the browser can start), actually browsing the web with Tor is pretty intuitive.

The main difference is that **when you browse the web with Tor, your real IP address and other system information is obscured** from the websites and services you're visiting. Additionally, it also hides what you're doing from your Internet Service Provider.

The primary uses of Tor are the following:

- Bypassing censorship and surveillance
- Visiting websites anonymously
- Accessing Tor hidden services (.onion sites)

Tor: Pros and Cons

Using Tor offers a number of privacy and anonymity protections over directly connecting to a website. That said, it also presents some challenges.

Pros

- If you use Tor correctly, **your real IP address cannot be determined** by the websites you visit.
- You can access websites without your internet service provider being aware of your browsing history.
- You can **bypass many kinds of censorship**.

Cons

- Tor is **very slow compared to VPNs** and regular web browsing, so downloading large files is usually not feasible.
- It's possible to deanonymize your browsing by making a simple mistake.
- Some governments and network operators can prevent Tor from functioning.
- Although using Tor is legal in and of itself, **using Tor may make your activity appear suspicious**.
- Websites may refuse to function when you're using Tor—generally to prevent anonymous spam and abuse.

Who created Tor?

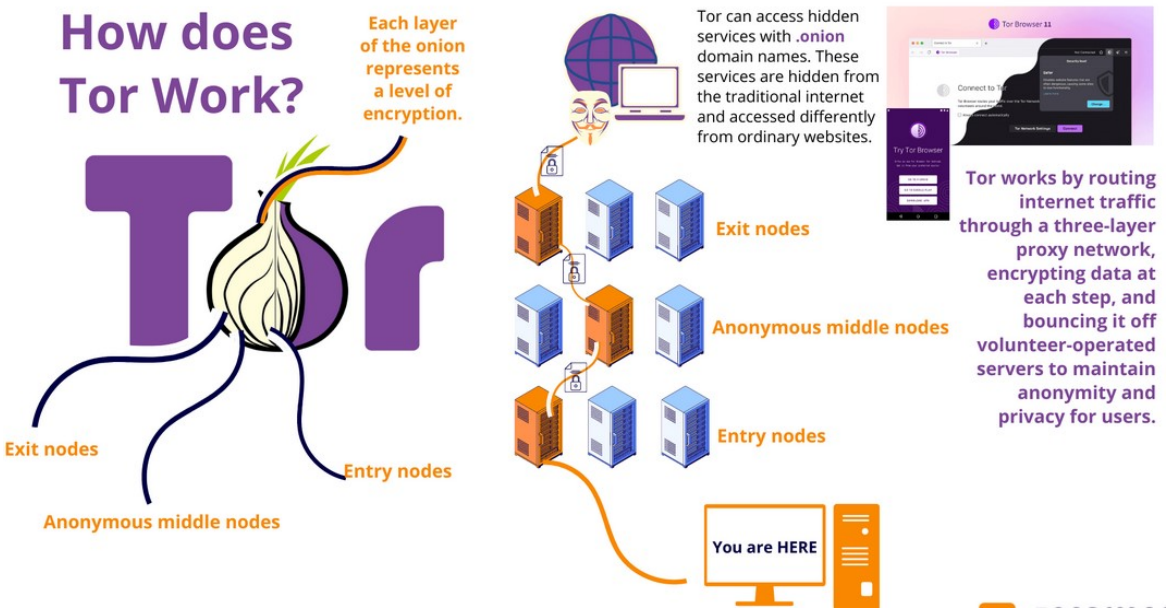
The concepts underpinning Tor — namely, onion routing — **were developed by the United States government in the 1990s**. It was originally designed to protect the communications of US intelligence agencies across the Internet. The original code for Tor was released under a free and open-source software license by the United States Naval Research Laboratory, allowing other people and organizations to contribute to the project.

Since 2006, a nonprofit called The Tor Project has been responsible for maintaining Tor and the Tor Browser. Financial support comes from corporations like Google, organizations such as Human Rights Watch, and many others.

How does Tor work?

There are two things people may mean when they say "Tor": the networking system and the Tor Browser.

To anonymize Internet usage, **Tor routes traffic through multiple randomly-chosen relay servers** before accessing the destination website. There are over 7,000 of these servers, which mostly belong to volunteers. The request is encrypted multiple times, so the relay servers only know the previous relay and the next relay, but not the request contents or the full circuit. The network request finally exits the Tor network at an exit node. From the website's perspective, you are browsing directly from the exit node.



Tor hidden services, which will be covered below, are accessed in a slightly different way from standard websites — they use **.onion domain names and are inaccessible from the regular web**.

To actually use Tor to anonymize your communications, you run the Tor Browser on your computer. The Tor Browser is a modified version of Mozilla Firefox that connects to the internet via the Tor network. In addition to the functionality necessary to use Tor, the Tor Browser also bundles a number of extensions that help users maintain their privacy. For example, the NoScript extension is bundled with Tor out of the box, meaning that users have to manually approve individual JavaScript files before they can run—helping to protect against fingerprinting and browser security exploits.

Tor browser security: how safe, anonymous, and secure is it?

Although Tor is frequently used by privacy-conscious people and those looking to avoid surveillance, it isn't perfect. Simple mistakes can make hours of meticulous privacy protection useless, so it pays to be especially careful when browsing with Tor. Here are some things you should look out for:

- **The final part of the communication is unencrypted**

Even though Tor encrypts data between the user's computer and servers in the Tor network and within the Tor network, it does not encrypt the final part of the connection between the exit node and the destination server. As a result, it is possible for a government or internet service provider to eavesdrop on traffic between the exit node and destination servers. Since the full list of Tor exit nodes is publicly available, any unencrypted traffic leaving exit nodes is likely to be monitored closely.

- **Your traffic may be deanonymized using timing-based statistical techniques**

Another security concern is when the entry relay and exit relay both exist on the same Internet autonomous system (AS)— like if the same network operator owns both IP addresses. If this is the case, it is possible for that network operator to use timing-based statistical techniques to determine that a particular network request originated from a particular computer. This technique is difficult to execute, so it's usually only possible for governments to pull off. Additionally, it can be expensive, so it is not usually a concern except for high-value targets.

- **Tor won't protect you against sophisticated fingerprinting methods**

Other signals may also be used to fingerprint users. If you happen to be browsing a compromised website using Tor with JavaScript enabled (or the website you're browsing uses compromised third-party JavaScript), it's possible for the attacker to determine who you are based on your mouse movements. Most people move their mouse in a distinct way which can be used to correlate a Tor browsing session with a regular, non-Tor browsing session.

- **Even Tor has bugs that can be exploited**

As with regular web browsing, it is always possible to have your web browser compromised as a result of a security bug. While modern browsers, including the Firefox-based Tor Browser, include very good exploit protection, the kinds of adversaries that target Tor users also stockpile browser exploits that browser vendors are completely unaware of (known as "zero-day" bugs).

Although most security considerations for Tor are only applicable to the most paranoid users, it's still a good idea to follow some safety guidelines. Anyone using Tor is automatically enough of a surveillance target that protecting your security is important.

How to protect yourself when using Tor

1. **Don't log into your usual accounts** - especially Facebook or Google.
2. **Try not to follow any unique browsing patterns** that may make you personally identifiable.
3. **Turn the Tor Browser's security level up to the max.** This will disable JavaScript on all sites, disable many kinds of fonts and images, and make media like audio and video click-to-play. This level of security significantly decreases the amount of browser code that runs while displaying a web page, protecting you from various bugs and fingerprinting techniques.
4. **Use the HTTPS Everywhere extension.** This will ensure you're only browsing HTTPS websites and protect the privacy of your data as it goes between the final node and the destination server.
5. **As a general rule, never use BitTorrent over Tor.** Although people illegally pirating copyrighted content may wish to obscure their real identity, BitTorrent is extraordinarily difficult to use in a way that does not reveal your real IP address. Tor is relatively slow, so BitTorrent is hardly worth using over Tor anyway.
6. Most importantly, **always keep Tor Browser (and any extensions) updated**, reducing your attack surface.

What are Tor hidden services?

Tor hidden services, "onion services", or "Tor websites" are websites that are only accessible from within the Tor network. All hidden service domain names end in .onion and consist of a very long of seemingly-random characters. Collectively, **Tor hidden services are sometimes referred to as the "dark web."**

These websites are not indexed by and won't appear on search engines like Google. Instead, a number of user-created directories of hidden services allow you to find the sites you're looking for.

The dark side of Tor

The term "dark web" generally evokes a mental image of criminals selling illegal services through sketchy Tor-based marketplaces. Although much of the news surrounding Tor and Tor hidden services makes it seem as though using them is like walking down a dark alley at night, there are plenty of perfectly legitimate uses for hidden services as well.

Sites like Facebook, DuckDuckGo, and The New York Times run versions of their sites with .onion addresses for journalists and activists living under oppressive regimes.

Some extremely privacy-conscious individuals make their blogs or websites exclusively accessible through Tor hidden services so that their real-life identity cannot be determined.

The US government has steadily improved its technologies for catching and prosecuting criminals who use Tor for illegal purposes. As mentioned above, Tor is vulnerable to timing attacks and other types of advanced compromises that require government-like resources. For that reason, **a government with sufficient resources can deanonymize some Tor requests** if they control the right internet service providers (which is almost always true of the American government).

While using Tor itself is perfectly legal, there is plenty of illegal activity involving Tor, so be sure to watch out when visiting hidden services so that you don't accidentally end up on a sketchy .onion site. If you decide to access the Dark or the Deep web, do it in a safe way.

Tor vs. VPN services: what's the difference?

Virtual private network (VPN) services are frequently marketed as a way to improve privacy or gain anonymity. In reality, **VPN services can be hit-and-miss when it comes to privacy**. Some are certainly better at protecting data than Tor, whereas others will compromise your privacy more than not using one at all.

VPNs are point-to-point tunnels. Network traffic travels in an encrypted tunnel from the user's computer all the way to the VPN provider's network and is forwarded onto the destination server. In that sense, they are similar to Tor.

The most problematic aspect of VPN service marketing is the "no-logging policy." Most VPN providers advertise that they do not log how their services are used, so users can feel confident using the service with no legal ramifications. However, there are few ways this claim can be proven beyond a reasonable doubt. So if you do end up choosing a VPN for privacy, it's best to go with one that has had its **no-logs policy tested in the wild due to some police investigation**, or at least one that has had the infrastructure audited by a trustworthy, independent third-party.

The biggest advantages of VPNs over Tor include **much better connection speeds** for downloading videos and other large files. Also, with a VPN, you can **choose the server** your data is routed through, or at least the country in which that server is located. Finally, it is far **easier to use a VPN systemwide** than it is to use Tor outside of Tor Browser.

Whether you're using Tor or a VPN service, be careful not to leak your actual IP address through browser plugins or by logging into an account that would nullify your anonymity.

Note- For Toolkit/ API Name - Refer any textbook.