Dark Web Course

Antonio Brandao

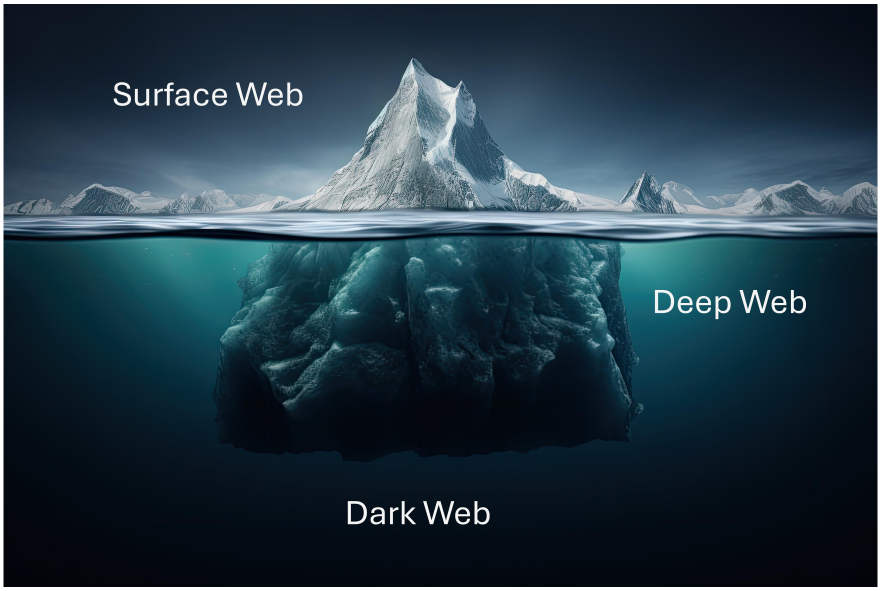
November 2024

DISCLAIMER:
This course aims to inform participants about the Dark Web without promoting or facilitating illegal activities. All activities are conducted with a focus on legality and ethics. Participants are reminded to adhere to all applicable laws and regulations. The course facilitators are not responsible for misusing the information provided during the workshop.

Contents

1 Introduction to the Dark Web

This section aims to provide a foundational understanding of the Dark Web by exploring its definitions, distinctions from other parts of the Internet, and its historical development. This will set the stage for more advanced topics later in the course.



1.1 Definitions

Understanding the different layers of the Internet is crucial for comprehending the Dark Web and how it fits into the broader context of the digital world.

1.1.1 Surface Web

- **Definition:** The Surface Web, also known as the Visible Web or Indexed Web, consists of all websites and web pages that are indexed by standard search engines like Google, Bing, and Yahoo.

- **Characteristics:**
  - **Accessibility:** Easily accessible to the general public using standard web browsers without special configurations.
  - **Indexing:** Content is crawled and indexed by search engine bots, making it searchable through common search queries.
  - **Examples:** News websites, social media platforms, company websites, blogs, and online stores.

- **Limitations:**
  - **Privacy:** Information is publicly available, which may raise privacy concerns for individuals and organizations.
  - **Content Control:** Subject to censorship and content regulations imposed by governments or hosting platforms.

1.1.2 Deep Web

- **Definition:** The Deep Web encompasses all online content not indexed by standard search engines. This includes any web pages behind paywalls that require authentication or are dynamically generated in response to a user's query.

- **Characteristics:**
  - **Accessibility:** Accessible to users with the correct URL or login credentials but not discoverable through search engines.
  - **Content Types:**
    - **Private Databases:** Academic journals, legal documents, medical records.
    - **Intranets:** Corporate networks, internal government systems.
    - **Dynamic Content:** Web pages generated in response to specific queries, such as airline flight schedules or bank account information.
  - **Importance:**
    - **Privacy and Security:** Protects sensitive information from being publicly accessible.
    - **Volume:** Estimated to be significantly larger than the Surface Web, containing vast amounts of data.

1.1.3 Dark Web

- **Definition:** The Dark Web is a subset of the Deep Web that is intentionally hidden and requires specific software, configurations, or authorization. It operates on overlay networks that use the Internet but require specialized protocols for access.

- **Characteristics:**
  - **Anonymity:** Both users and operators maintain anonymity through encryption and routing techniques. The most common network is Tor (The Onion Router).
  - **Access Requirements:** Requires special browsers or configurations (e.g., Tor Browser) to access .onion sites.
  - **Content:**
    - **Legitimate Uses:** Forums for free speech, whistleblowing platforms, secure communication channels.
    - **Illicit Activities:** Marketplaces for illegal goods, hacking services, and other criminal enterprises.

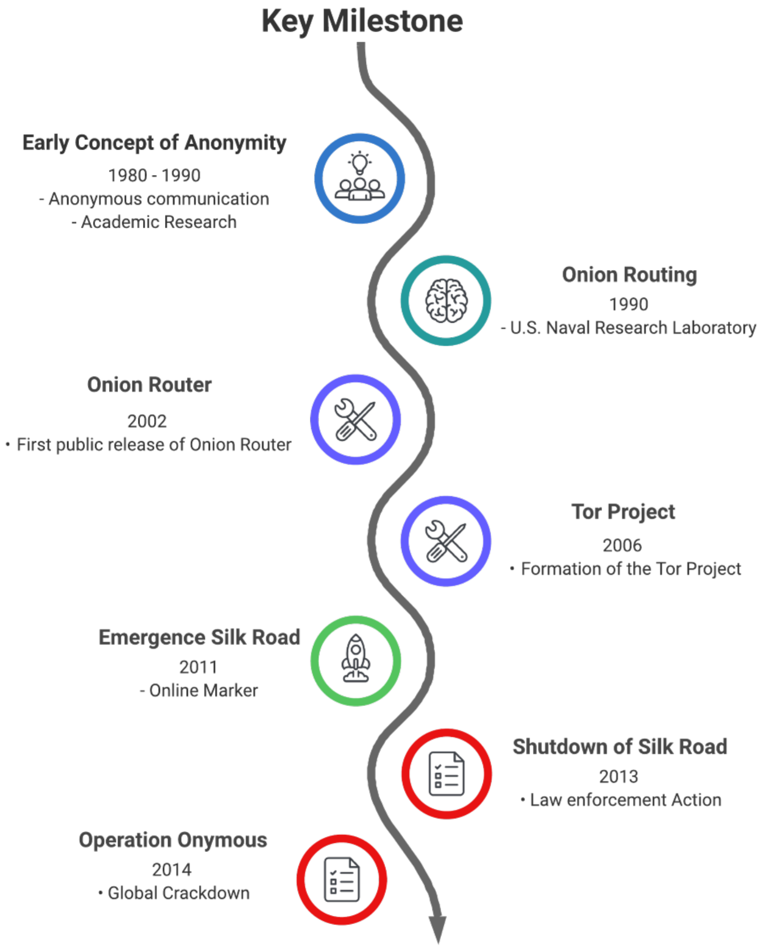
- **Misconceptions:**
  - **Not Entirely Illegal:** While the Dark Web is often associated with criminal activity, it is also a crucial tool for privacy advocates, journalists, and citizens under oppressive regimes.
  - **Size:** Contrary to popular belief, the Dark Web constitutes a small fraction of the Internet.

1.2 History and Evolution

Exploring the origins and development of the Dark Web provides insight into its current state and future trajectory.

1.2.1 Origins of the Dark Web

- **Early Concepts of Anonymity:**
  - **1980s-1990s:**
    The concept of anonymous communication networks began to take shape with the development of protocols that allowed for secure and private messaging.
  - **Academic Research:** Early research papers proposed methods for anonymous communication to protect privacy and free speech.
- **Development of Onion Routing:**
  - **1990s:**
    The US Naval Research Laboratory initiated the development of onion routing to protect sensitive government communications.
  - **Purpose:** To enable secure and anonymous communication over a public network by encrypting data in multiple layers (like an onion).
- **Introduction of Tor:**
  - **2002:** The first public release of Tor (The Onion Router) allowed civilians to use the network for anonymous browsing.
  - **Open Source:** Tor's code was made publicly available, encouraging



    transparency and community development.
  - **Funding and Support:** Initially funded by the U.S. government, later receiving support from various organizations advocating for privacy and free speech.

1.2.2 Key Milestones in Dark Web Development

- **2006 - Formation of The Tor Project:**
  - **Non-Profit Organization:** Established to maintain and develop the Tor network and software.
  - **Mission:** Promote privacy and freedom online by providing tools for anonymous communication.
- **2011 - Emergence of Silk Road:**
  - **Online Marketplace:** Silk Road became one of the first major darknet markets, facilitating the sale of illegal drugs and other contraband using Bitcoin.
  - **Impact:** Brought significant attention to the Dark Web and raised awareness about anonymous marketplaces.
- **2013 - Shutdown of Silk Road:**
  - **Law Enforcement Action:** The FBI seized Silk Road, arresting its founder, Ross Ulbricht.
  - **Aftermath:** Led to the proliferation of new darknet markets as successors attempted to fill the void.
- **2014 - Operation Onymous:**
  - **Global Crackdown:**A coordinated effort by international law enforcement agencies resulted in the seizure of several darknet sites and arrests of operators.
  - **Significance:** Demonstrated the authorities' ability to penetrate the Dark Web's anonymity under certain circumstances.
- **Development of Alternative Networks:**
  - **I2P (Invisible Internet Project):**
    - **Purpose:** Designed for secure and anonymous communication.
    - **Features:** Peer-to-peer architecture, fully distributed network.
  - **Freenet:**
    - **Function:** Allows for anonymous publishing and retrieval of information.
    - **Philosophy:** Emphasizes freedom of speech and resistance to censorship.
- **Technological Enhancements:**
  - **Improved Encryption Methods:** Strengthening encryption protocols to enhance privacy.
  - **User-Friendly Interfaces:** Development of more accessible tools, making it easier for non-technical users to access the Dark Web.
- **Emergence of Darknet Communities:**
  - **Forums and Social Networks:** Growth of communities centred around privacy, hacking,

and other niche interests.
- **Cultural Impact:** Formation of a subculture with its norms, languages, and ethical codes.

1.2.3 Current State of the Dark Web

- **Diversification of Content:**
  - **Illicit Marketplaces:** Continuation and evolution of marketplaces despite law enforcement efforts.
  - **Legitimate Uses:** Increased utilization by journalists, activists, and citizens seeking privacy.
- **Law Enforcement and Regulatory Responses:**
  - **Advanced Techniques:** Adoption of sophisticated methods to track and identify illegal activities.
  - **International Cooperation:** Collaboration among governments to address cross-border challenges posed by the Dark Web.
- **Public Perception:**
  - **Media Representation:** Often sensationalized, leading to misunderstandings about the nature and scale of the Dark Web.
  - **Education and Awareness:** Growing efforts to educate the public on digital privacy and security.

1.3 Importance of the Dark Web in Today's Digital Landscape

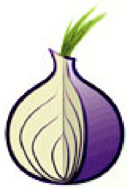
- **Privacy Concerns:**
  - **Data Surveillance:** In an era of increasing data collection by corporations and governments, the Dark Web offers tools for anonymity.
  - **Freedom of Expression:** Enables individuals in restrictive environments to communicate and share information without fear of reprisal.
- **Cybersecurity Implications:**
  - **Threat Vector:** The Dark Web is a marketplace for cybercriminals to buy and sell exploits, malware, and stolen data.
  - **Intelligence Gathering:** Organizations monitor Dark Web activities to anticipate and mitigate potential threats.
- **Ethical and Legal Debates:**
  - **Balancing Act:** The challenge of protecting privacy rights while preventing criminal activities.
  - **Policy Development:** Ongoing discussions around regulations, censorship, and internet governance.

1.4 Summary

By elaborating on these topics, participants will gain a comprehensive understanding of the Dark Web's definition, historical context, and its significance in the modern digital world. This foundational knowledge is essential for engaging with subsequent course sections, which will delve deeper into technical aspects, practical applications, risks, and ethical considerations.

Learning Objectives:

- Comprehend the distinctions between the Surface, Deep, and Dark Web.
- Trace the historical development of the Dark Web from its origins to the present day.
- Recognize the technological advancements that have shaped the Dark Web.
- Understand the Dark Web's dual nature, including its legitimate uses and associations with illicit activities.
- Appreciate the complexities and nuances that make the Dark Web a significant Internet component.



TOR Project Icon

1.5 Additional Resources

- **The Tor Project. "Overview."** The Tor Project,https://2019.www.torproject.org/about/overview.html.en#overview
- **The Tor Project. "History."** The Tor Project,https://www.torproject.org/about/history/
- **Onion Routing Explained** The Tor Project, https://2019.www.torproject.org/about/overview.html.en#thesolution
- **Information Technology Division / History and Heritage** U.S. Naval Research Laboratory, https://www.nrl.navy.mil/itd/history-heritage/
- **Anonymity** Electronic Frontier Foundation, https://www.eff.org/issues/anonymity.
- **Data Exploitation** Privacy International, https://privacyinternational.org/learn/data-exploitation
- **About SecureDrop** Freedom of the Press Foundation, https://securedrop.org/overview/

2 Understanding the Technology Behind the Dark Web

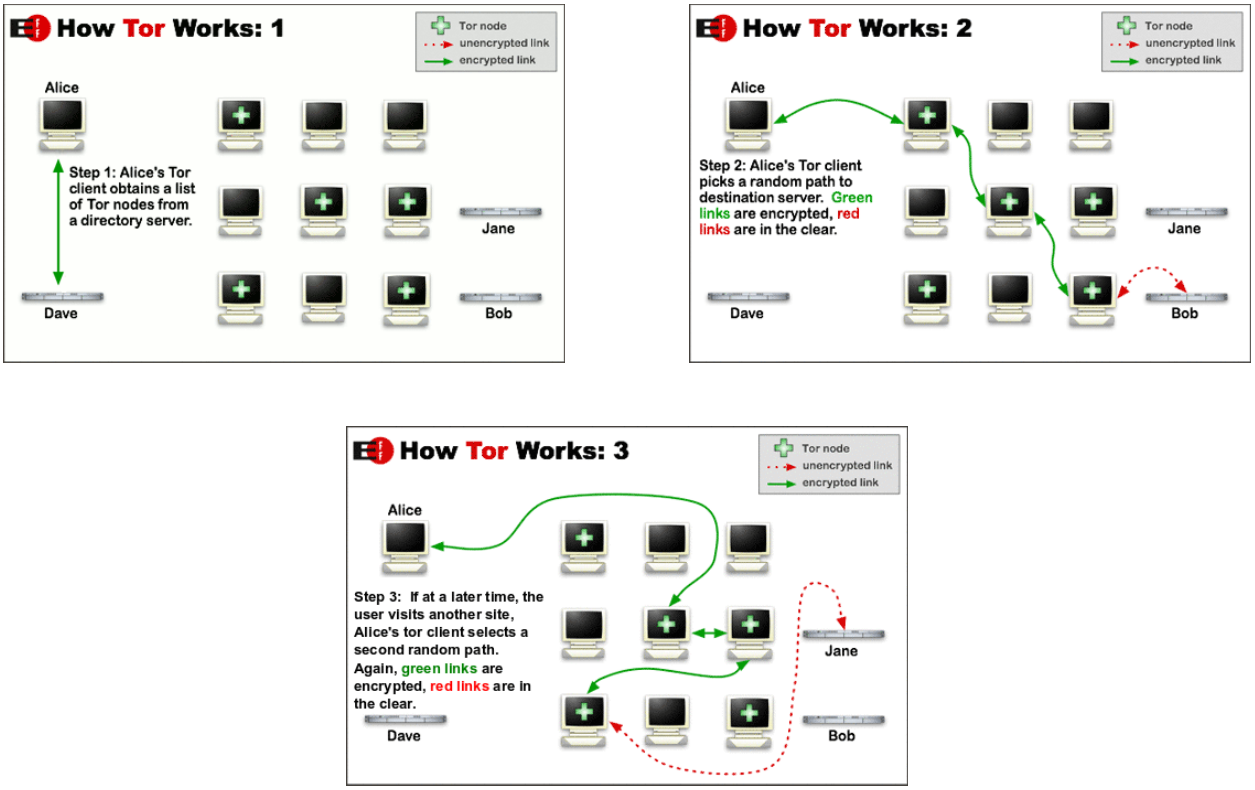
This section delves into the technological foundations that enable the Dark Web, focusing on the tools and protocols that ensure anonymity and secure communication. Understanding these technologies is essential for comprehending how the Dark Web operates and the measures taken to protect user privacy.

2.1 The Tor Network

The Tor Network is the most widely used platform for accessing the Dark Web. It allows users to browse the Internet anonymously by routing their traffic through volunteer-operated servers called nodes or relays.

2.1.1 How Tor Works

Tor, short for ”The Onion Router,” is designed to conceal a user’s identity and online activity from surveillance and traffic analysis by separating identification and routing.


How Tor Works: 1
Tor node
unencrypted link
encrypted link
Alice
Step 1: Alice's Tor client obtains a list of Tor nodes from a directory server.
Jane
Dave
Bob
How Tor Works: 2
Tor node
unencrypted link
encrypted link
Alice
Step 2: Alice's Tor client picks a random path to destination server. Green links are encrypted, red links are in the clear.
Jane
Dave
Bob
How Tor Works: 3
Tor node
unencrypted link
encrypted link
Alice
Step 3: If at a later time, the user visits another site, Alice's tor client selects a second random path. Again, green links are encrypted, red links are in the clear.
Jane
Dave
Bob

Source: Tor Project

- **Onion Routing:** Data is encrypted in multiple layers, like the layers of an onion. Each layer is peeled off by a successive Tor relay, decrypting a portion of the data and passing it to the next relay.

- **Circuit Establishment:**
  - **Entry Node:** The user's Tor client selects an entry node (guard node) to initiate the connection.
  - **Middle Relays:** The data is passed through randomly selected middle relays, each unaware of the full path.
  - **Exit Node:** The final relay decrypts the last layer and sends the data to the intended destination
- **Anonymity Preservation:** Since each relay only knows the preceding and following nodes, it is challenging for anyone to trace the data back to the user.

2.1.2 Onion Routing Explained

Onion routing is the core principle behind Tor's ability to provide anonymity.

- **Layered Encryption:** Before data is sent, the user's Tor client encrypts it multiple times using the public keys of each relay in the circuit, starting with the exit node and ending with the entry node
- **Data Transmission:**
  - **First Relay (Entry Node):** Removes the outermost layer of encryption, revealing the next relay in the path.
  - **Middle Relays:** Each subsequent relay removes another layer of encryption, learning only the identity of the next relay.
  - **Exit Node:** Removes the final layer, accessing the original data before sending it to the destination
- **Benefits:**
  - **Anonymity:** Difficult for adversaries to perform traffic analysis or identify the user's IP address.
  - **Security:** Protects against eavesdropping and man-in-the-middle attacks.

2.2 Anonymity and Encryption

Anonymity and encryption are fundamental to the functioning of the Dark Web, enabling users to communicate without revealing their identities or compromising their data.

2.2.1 Importance of Anonymity Online

- **Privacy Protection:** In an era of pervasive surveillance, anonymity safeguards personal freedoms and protects individuals from data exploitation.
- **Freedom of Expression:** Allows individuals, especially under oppressive regimes, to express dissenting opinions without fear of persecution.
- **Whistleblowing and Journalism:** Enables secure communication channels for whistleblowers to share sensitive information with journalists, promoting transparency and accountability

2.2.2 Encryption Techniques Used

- **Symmetric Encryption:**
  - **Definition:** Uses the same key for encryption and decryption.
  - **Example Algorithms:** AES (Advanced Encryption Standard).
  - **Usage:** Efficient for encrypting large amounts of data but requires secure key exchange
- **Asymmetric Encryption:**
  - **Hash Functions:**
    - **Purpose:** Converts data into a fixed-size hash value, ensuring data integrity.
    - **Example Algorithms:** SHA-256, SHA-3.
    - **Usage:** Verifying the integrity of transmitted data
- **Perfect Forward Secrecy (PFS):**
  - **Definition:** Ensures that session keys cannot be compromised even if the private key is compromised in the future.
  - **Implementation:** Utilizes ephemeral keys that are not stored after the session ends
- **Transport Layer Security (TLS):**
  - **Purpose:** Encrypts data between web applications and servers.
  - **Usage:** Secures communications over the Internet, including within the Tor network

2.3 Alternative Networks

In addition to Tor, other networks provide anonymous communication and data-sharing platforms.

2.3.1 I2P (Invisible Internet Project)

- **Overview:** I2P is an anonymous overlay network designed to facilitate censorship-resistant and peer-to-peer communication
- **Features:**
  - **Decentralization:** Lacks central servers, relying on a distributed architecture.
  - **Garlic Routing:** Enhances onion routing by bundling messages together, improving efficiency and reducing the ability to perform traffic analysis
- **Service:**
  - **Eepsites:** Anonymous websites accessible only within the I2P network.
  - **Email and Messaging:** Encrypted communication tools.
  - **File Sharing:** Supports anonymous file transfers using protocols like BitTorrent
- **Advantages:**
  - **Dynamic Tunnels:** Routes are frequently changed, enhancing anonymity.
  - **Integrated Applications:** Offers built-in tools for various services, making it versatile for users

2.3.2 Freenet

- **Overview:** Freenet is a peer-to-peer platform that aims to provide censorship-resistant communication and publishing.

- **Features:**
  - **Distributed Data Storage:** Users contribute storage space, and data is stored across the network in encrypted form.
  - **Anonymity Modes:**
    - **Opennet Mode:** Connects to any user on the network.
    - **Darknet Mode:** Connects only to trusted peers, enhancing security and resistance to attacks
- **Services:**
  - **Freesites:** Websites hosted within Freenet, accessible only through the network.
  - **Content Publishing:** Users can publish content anonymously or pseudonymously.
  - **Messaging and Forums:** Supports anonymous communication channel
- **Advantages:**
  - **Censorship Resistance:** Designed to be resilient against attempts to remove content.
  - **Privacy Protection:** Strong focus on user anonymity and data security

2.4 Summary

This section explores the technological foundations that make the Dark Web possible, focusing on the tools and protocols that ensure anonymity and secure communication for its users.

- **The Tor Network:** The Tor Network is central to the Dark Web. It enables anonymous browsing by routing user traffic through volunteer-operated servers called relays. The concept of onion routing is explained, where data is encrypted in multiple layers, each peeled off by successive relays, making it difficult to trace the origin of the data. This process preserves user anonymity and protects against surveillance and traffic analysis.
- **Anonymity** and Encryption: The importance of anonymity online is highlighted, emphasizing its role in protecting privacy, enabling freedom of expression, and facilitating secure communication for whistleblowers and journalists. Various encryption techniques are discussed, including symmetric and asymmetric encryption, hash functions, and protocols like Perfect Forward Secrecy (PFS) and Transport Layer Security (TLS), which are essential for securing data transmission and maintaining confidentiality on the Dark Web.
- **Alternative Networks:**
  - **I2P (Invisible Internet Project):** I2P is introduced as an anonymous overlay network designed for secure, peer-to-peer communication. Features like garlic routing enhance anonymity by bundling messages together. I2P supports anonymous websites (eepsites), email, messaging, and file sharing.
  - **Freenet:** Freenet is presented as a peer-to-peer platform focused on censorship-resistant communication and publishing. It utilizes distributed data storage across the network, with strong anonymity modes and services like anonymous websites (freesites), content publishing, messaging, and forums.
- **Learning Objectives:**
  - Understand how the Tor Network functions and the principles of onion routing that enable anonymous communication.
  - Explain the significance of anonymity and encryption in protecting online privacy and enabling secure communication, along with the key encryption techniques used.

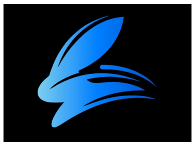
  - Identify and describe alternative anonymous networks like I2P and Freenet, understanding their features, purposes, and how they differ from the Tor Network.
  - Appreciate the technological mechanisms that underpin the Dark Web, gaining a deeper insight into its operations and the tools that safeguard user anonymity and data security.

2.5 Additional Resources:

- **"Tor: The Second-Generation Onion Router."** https://www.usenix.org/legacy/event/sec04/tech/full_papers/dingledine/dingledine.pdf
- **"Onion Routing Explained."**https://2019.www.torproject.org/about/overview.html.en#thesolution
- **"Tor: Overview."** https://2019.www.torproject.org/about/overview.html.en#overview
- **"Anonymity."https://www.eff.org/issues/anonymity**
- **"Overview SecureDrop."**https://securedrop.org/overview/
- **"New Directions in Cryptography."**https://ieeexplore.ieee.org/document/1055638
- **"About OpenSSL."**https://www.openssl.org/
- **"About I2P."**https://geti2p.net/en/about/intro
- **"Garlic Routing."**https://geti2p.net/en/docs/how/garlic-routing
- **"What Is Freenet?"**https://freenetproject.org/pages/about.html
- **"Freenet: A Distributed Anonymous Information Storage and Retrieval System."**https://link.springer.com/chapter/10.1007/3-540-44702-4_4

Source: Tor Project

Source: I2P



Source: Freenet

3 Legitimate Uses of the Dark Web

This section explores the positive and lawful applications of the Dark Web, emphasizing how it serves as a vital tool for protecting privacy, enabling free speech, facilitating whistleblowing and activism, and supporting journalism, especially in oppressive environments.

3.1 Privacy and Free Speech

3.1.1 Protection from Surveillance

- **Privacy Concerns:** In an age where digital surveillance by governments and corporations is pervasive, individuals seek methods to safeguard their personal information and communications. The Dark Web provides tools that enhance online privacy and anonymity.
- **Anonymity Networks:** The Tor network allows users to conceal their IP addresses and browsing activities, protecting them from tracking and profiling.
- **Legal Use Cases:** Ordinary citizens utilize the Dark Web to avoid unwanted surveillance, maintain confidentiality in their online interactions, and protect sensitive personal data.

3.1.2 Platforms for Open Dialogue

- **Freedom of Expression:** The Dark Web offers platforms where individuals can freely express their opinions without fear of censorship or retaliation, particularly in countries with strict internet regulations.
- **Forums and Communities:** Anonymous forums on the Dark Web facilitate open discussions on sensitive topics, enabling users to share ideas and information without revealing their identities.
- **Protection of Minority Voices:** It provides a voice to marginalized groups who might otherwise be silenced, promoting diversity of thought and opinion.

3.2 Whistleblowing and Activism

3.2.1 Secure Channels for Information Leaks

- **Whistleblower Protections:** The Dark Web offers secure channels for whistleblowers to leak information about corporate or governmental misconduct without risking their safety or careers.
- **Encryption and Anonymity:** Tools provide end-to-end encryption and anonymity, enabling secure communication between sources and journalists.

3.2.2 Case Studies (e.g., SecureDrop):

- **SecureDrop:** An open-source platform media organizations use to securely receive documents and communicate with anonymous sources. Major organizations like The Washington Post, The New York Times, and The Guardian use SecureDrop to facilitate confidential submissions from whistleblowers. https://www.nytimes.com/2019/11/12/opinion/whistleblower.html
- **Impact on Journalism:** SecureDrop has enabled significant investigative reporting by protecting sources who expose corruption, human rights abuses, and other critical issues. https://www.cjr.org/tow_center_reports/guide_to_securedrop.php

3.3 Journalism

3.3.1 Communicating Under Oppressive Regimes

- **Circumventing Censorship:** Journalists operating in countries with strict censorship laws use the Dark Web to access blocked information and communicate securely.
- **Safe Reporting:** The anonymity provided by the Dark Web allows reporters to investigate and report on sensitive topics without risking government surveillance or repression.

3.3.2 Protecting Sources and Data

- **Confidentiality:** Maintaining the confidentiality of sources is paramount in journalism. The Dark Web's secure communication channels help protect the identities of informants.
- **Data Security:** Journalists can securely store and transfer sensitive documents using encrypted services on the Dark Web, reducing the risk of interception or hacking.

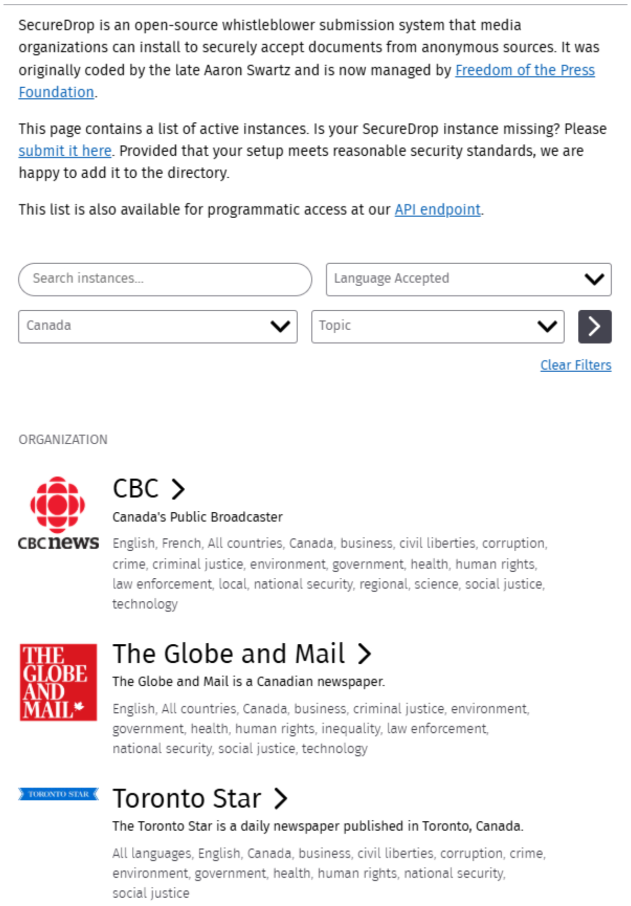
3.4 Summary

- **Learning Objectives:**
  - Understand how the Dark Web facilitates privacy and free speech, protecting individuals from surveillance and enabling open dialogue in restrictive environments.
  - Recognize the role of the Dark Web in supporting whistleblowers and activists through secure channels for information leaks and understand case studies like SecureDrop.
  - Appreciate the importance of the Dark Web for journalists in communicating under oppressive regimes and protecting sources and data, ensuring the free flow of information.
  - Critically assess the legitimate uses of the Dark Web, acknowledging its value in promoting transparency, accountability, and human rights.

3.5 Additional Resources

- ”About SecureDrop.” https://securedrop.org/
- ”List of SecureDrops” https://securedrop.org/directory/
- ”How to Be a Whistle-Blower” https://www.nytimes.com/2019/11/12/opinion/whistleblower.html
- ”Got a confidential news tip?” NYTIMES https://www.nytimes.com/tips
- ” Guide to SecureDrop” https://www.cjr.org/tow_center_reports/guide_to_securedrop.php

- ”Securing newsrooms against digital threats: lessons from new media innovators” https://reutersinstitute.politics.ox.ac.uk/sites/default/files/2022-08/RISJ%20Paper_Paul%20F_TT22_Final%20%281%29.pdf
- ”Responsible Journalism and National Security in the Age of Data” https://reutersinstitute.politics.ox.ac.uk/news/responsible-journalism-and-national-security-age-data
- ”Five digital security tools to protect your work and sources” https://www.icij.org/inside-icij/2018/01/five-digital-security-tools-to-protect-your-work-and-sources/



SecureDrop is an open-source whistleblower submission system that media organizations can install to securely accept documents from anonymous sources. It was originally coded by the late Aaron Swartz and is now managed by Freedom of the Press Foundation.

This page contains a list of active instances. Is your SecureDrop instance missing? Please submit it here. Provided that your setup meets reasonable security standards, we are happy to add it to the directory.

This list is also available for programmatic access at our API endpoint.

CBC >

Canada's Public Broadcaster

English, French, All countries, Canada, business, civil liberties, corruption, crime, criminal justice, environment, government, health, human rights, law enforcement, local, national security, regional, science, social justice, technology

The Globe and Mail >

The Globe and Mail is a Canadian newspaper.

English, All countries, Canada, business, criminal justice, environment, government, health, human rights, inequality, law enforcement, national security, social justice, technology

Toronto Star >

The Toronto Star is a daily newspaper published in Toronto, Canada.

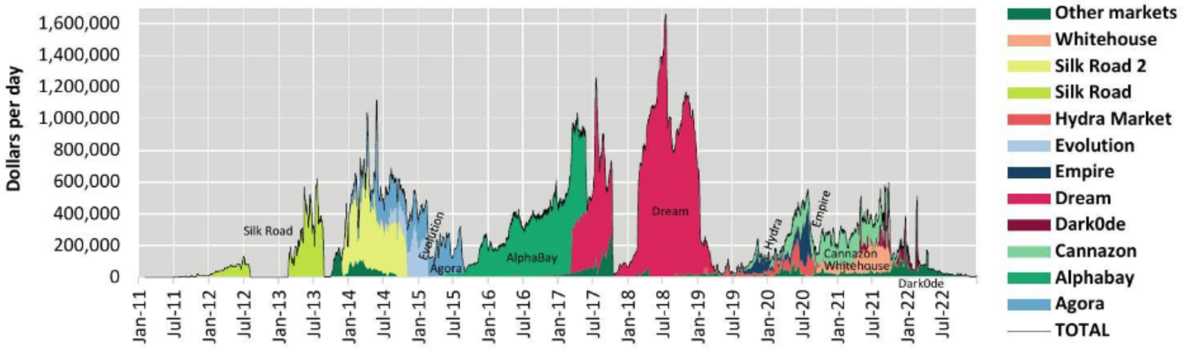
All languages, English, Canada, business, civil liberties, corruption, crime, environment, government, health, human rights, national security, social justice

SecureDrops in Canada

4 Illicit Activities on the Dark Web

This section explores the various illicit activities on the Dark Web, providing an overview of illegal content and services, cybersecurity threats, and the ethical considerations surrounding these activities. Understanding these aspects is crucial for comprehending the challenges posed by the Dark Web to law enforcement, cybersecurity professionals, and society as a whole.

FIG. 16 Daily minimum sales (mostly drug-related: >90 per cent) on 39 major global darknet markets, 2011–2022



Source: UNODC analysis based on Hikari Labs data (see online Methodological Annex).

4.1 Overview of Illegal Content and Services

4.1.1 Cybercrime Marketplaces

- **Definition and Function:** Cybercrime marketplaces on the Dark Web are online platforms where illegal goods and services are bought and sold anonymously. These include stolen data, counterfeit documents, illicit software, and hacking tools.
  - Examples:
    - **Silk Road:** One of the first and most notorious darknet markets, primarily known for drug trafficking, before its shutdown in 2013.
    - **Evolution, AlphaBay, and Hansa:** Successor marketplaces that emerged after the shutdown of Silk Road, dealing in drugs, weapons, and other illegal items.
  - **Transactions and Anonymity:** Use of cryptocurrencies like Bitcoin and Monero to facilitate anonymous transactions.

4.1.2 Illegal Substances and Weapons

- **Drug Trafficking:** The Dark Web has become a significant platform for the sale of illegal drugs, offering a variety of substances with purported quality assurances and user reviews.
- **Arms Trafficking:** Sale of firearms, explosives, and other weapons, although less prevalent compared to drug markets.

4.1.3 Human Trafficking and Exploitation

- **Illicit Services:** The Dark Web hosts platforms facilitating human trafficking, including forced labour and sexual exploitation.
- **Child Exploitation Material:** Distribution of illegal content involving minors poses significant challenges to law enforcement due to encryption and anonymity.

4.2 Cybersecurity Threats

4.2.1 Hacking Services

- **Hacking-for-Hire:** Services offering to conduct cyberattacks, such as Distributed Denial of Service (DDoS), intrusion, and data theft, for a fee.
- **Sale of Exploits and Malware:** Marketplaces for zero-day exploits, ransomware, keyloggers, and other malicious software.

4.2.2 Malware Distribution

- **Ransomware-as-a-Service (RaaS):** Platforms that provide ransomware tools to affiliates in exchange for a share of the profits from attacks.
- **Botnets and Distributed Attacks:** Sale and rental of botnets used to conduct large-scale cyberattacks.

4.3 Ethical Considerations

4.3.1 Moral Implications

- **Normalization of Crime:** The anonymity of the Dark Web can lower psychological barriers to engaging in criminal activities.
- **Ethical Dilemmas for Researchers:** Studying illicit activities on the Dark Web poses ethical challenges, including exposure to illegal content and potential legal risks.

4.3.2 Impact on Society

- **Economic Consequences:** Cybercrime leads to significant financial losses for individuals, businesses, and economies.
- **Threats to Security and Privacy:** The proliferation of illegal activities on the Dark Web undermines trust in digital systems and poses threats to national security.
- **Social Harm:** Activities such as human trafficking and the distribution of child exploitation material have profound negative impacts on victims and society.

4.4 Summary

This section examines the range of illicit activities that occur on the Dark Web, including the operation of cybercrime marketplaces, the sale of illegal substances and weapons, and human trafficking. It also explores cybersecurity threats from the Dark Web, such as hacking services and malware distribution. The ethical considerations are discussed, highlighting these activities' moral implications and societal impact. Understanding these issues is essential for recognizing the challenges posed by the Dark Web and the importance of collaborative efforts to combat cybercrime.

- **Learning Objectives:**
  - Identify and describe the types of illegal content and services available on the Dark Web, including cybercrime marketplaces, illegal substances, weapons, and human trafficking.
  - Understand the cybersecurity threats associated with the Dark Web, such as hacking services and malware distribution, and their implications for individuals and organizations.
  - Analyze the ethical considerations and moral implications of illicit activities on the Dark Web and assess their impact on society, including economic, security, and social harm.
  - Recognize the challenges law enforcement and policymakers face in addressing Dark Web-related crimes while balancing privacy and security concerns.

4.5 Additional Resources

- ”Massive Blow to Criminal Dark Web Activities after Globally Coordinated Operation.” https://www.europol.europa.eu/media-press/newsroom/news/massive-blow-to-criminal-dark-web-activities-after-globally-coordinated-operation
- ”World Drug Report 2023: Drug Market Trends.” Chapter 07 - ”Use of the Dark Web and Social Media for Drug Supply” p.223 https://wdr.unodc.org/wdr2020/field/WDR20_Booklet_3.pdf
- ”Internet Organised Crime Threat Assessment (IOCTA) 2024” https://www.europol.europa.eu/cms/sites/default/files/documents/Internet%20Organised%20Crime%20Threat%20Assessment%20IOCTA%202024.pdf
- ”Comprehensive Cyber Review 2022 - US Department of Justice” https://www.justice.gov/usdoj-media/dag/media/1232936/dl?inline

5 Risks Associated with the Dark Web

This section explores the various risks of accessing and using the Dark Web. It highlights the security threats, legal implications, and personal safety concerns that users may encounter. Understanding these risks is crucial for anyone considering navigating the Dark Web, ensuring they take appropriate precautions.

5.1 Security Risks

5.1.1 Malware and Phishing Attacks

- **Prevalence of Malware:** The Dark Web is notorious for hosting malicious software. Users may inadvertently download malware such as viruses, trojans, ransomware, and spyware when accessing certain websites or downloading files. **Drive-by Downloads** - Some Dark Web sites are designed to automatically download malware onto a user's device without their knowledge.
- **Phishing Schemes:** Cybercriminals use phishing tactics to trick users into revealing sensitive information like login credentials, financial details, or personal data. **Clone Websites** - Imitating legitimate Dark Web sites to steal user information.
- **Exploitation of Anonymity:** The anonymity provided by the Tor network can embolden malicious actors, making it challenging to trace and combat cyber threats from the Dark Web.

5.1.2 Scams and Fraudulent Activities

- **Marketplace Scams:** Many Dark Web marketplaces are rife with fraudulent listings. Buyers may pay for goods or services that are never delivered. **Exit Scams** - Marketplace administrators shut down the site abruptly, taking all the escrowed funds.
- **Financial Fraud:** The Sale of stolen credit card information, counterfeit currency, and financial account details can lead to identity theft and significant financial losses.

5.2 Legal Risks

5.2.1 Understanding Local and International Laws

- **Illegality of Certain Activities:** Accessing or engaging with illegal content and services on the Dark Web can violate local, national, and international laws, leading to severe legal consequences. **Examples of Illegal Content:** Drugs, weapons, stolen data, and illicit pornography.
- **Jurisdictional Challenges:** The global nature of the Dark Web complicates legal enforcement, as activities may cross multiple jurisdictions with varying laws and regulations.

5.2.2 Potential Legal Consequences

- **Criminal Prosecution:** Individuals involved in illegal activities on the Dark Web may face criminal charges, including fines, imprisonment, and a permanent criminal record. **Case Examples** - Arrests following operations like Operation Onymous targeting Dark Web marketplaces.

- **Surveillance and Investigation:** Law enforcement agencies actively monitor the Dark Web to identify and prosecute illegal activities, potentially compromising user anonymity.

5.3 Personal Safety

5.3.1 Protecting Personal Information

- **Data Exposure Risks:** Users may inadvertently expose personal information through insecure connections, poor operational security, or interactions with malicious actors. **IP Leaks** - Misconfiguration or vulnerabilities can reveal a user's IP address.
- **Mitigation Strategies:**
  - **Use of VPNs:** Employing Virtual Private Networks (VPNs) alongside Tor can add an extra layer of security, though this has its risks and considerations.
  - **Operational Security (OpSec):** Practicing good OpSec by avoiding sharing personal details, using strong, unique passwords, and regularly updating software.

5.3.2 Recognizing Dangerous Situations

- **Avoiding Suspicious Links and Downloads:** Users should exercise caution when clicking links or downloading files, as they may contain malware or lead to phishing sites. **Verification** - Use trusted sources and verify URLs before accessing.
- **Awareness of Social Engineering Tactics:** Malicious actors may use manipulation techniques to extract information or gain trust. **Common Techniques** - Impersonation, urgency, and appealing to emotions.

5.4 Summary

This section highlights the significant risks users may encounter when accessing the Dark Web, including security threats such as malware, phishing attacks, scams, and fraudulent activities. It emphasizes the legal risks associated with engaging in or inadvertently encountering illegal content, stressing the importance of understanding local and international laws to avoid legal consequences. Personal safety concerns are addressed, and strategies for protecting personal information and recognizing potentially dangerous situations are provided. Overall, the section underscores the necessity of exercising caution and adopting robust security practices when navigating the Dark Web.

- **Learning Objectives**
  - Identify the security risks associated with the Dark Web, including malware, phishing attacks, scams, and fraudulent activities, and understand how to mitigate them.
  - Understand the legal risks of accessing and using the Dark Web, including complying with local and international laws to avoid potential legal consequences.
  - Recognize personal safety concerns when navigating the Dark Web, including protecting personal information and being aware of dangerous situations.
  - Apply best practices for secure and responsible use of the Dark Web, ensuring personal and legal safety.

5.5 Additional Resources

- ”Darknet Cybercrime Threats to Southeast Asia 2020” https://www.unodc.org/roseap/uploads/archive/documents/Publications/2021/Darknet_Cybercrime_Threats_to_Southeast_Asia_report.pdf
- ”Teaching Module Series - Organized Crime / Cybercrime - Module 13 Cyber Organized Crime” https://sherloc.unodc.org/cld/en/education/tertiary/cybercrime/module-13/index.html
- ”The Art of Deception: Controlling the Human Element of Security” https://www.wiley.com/en-us/The+Art+of+Deception%3A+Controlling+the+Human+Element+of+Security-p-9780764542800
- ”Baseline cyber threat assessment: Cybercrime” https://www.cyber.gc.ca/en/guidance/baseline-cyber-threat-assessment-cybercrime
- ”Cyber threat bulletin: Modern ransomware and its evolution” https://www.cyber.gc.ca/en/guidance/cyber-threat-bulletin-modern-ransomware-and-its-evolution
- ”National Cyber Threat Assessment 2023-2024” https://www.cyber.gc.ca/en/guidance/national-cyber-threat-assessment-2023-2024
- ”Ransomware” https://www.cyber.gc.ca/en/guidance/ransomware

6 Legal and Ethical Considerations

This section delves into the legal frameworks governing the use of the Dark Web and the ethical implications of anonymity and privacy. Understanding these aspects is crucial for anyone engaging with the Dark Web to ensure compliance with laws and to navigate the moral complexities involved.

6.1 Legal Framework

6.1.1 Laws Governing Dark Web Usage

- **International and National Laws:** The Dark Web operates globally, and users are subject to international laws as well as the laws of their respective countries. Engaging in illegal activities on the Dark Web can lead to prosecution under various legal statutes. **Computer Misuse and Cybercrime Laws** - Many countries have laws that criminalize unauthorized access to computer systems, distribution of malware, and other offences.
  - **Illicit Goods and Services:** Laws prohibit the trafficking of drugs, weapons, human trafficking, and other illegal goods and services commonly found on the Dark Web.
  - **Intellectual Property Laws:** The distribution of pirated software, music, movies, and other copyrighted materials is illegal under intellectual property laws.
- **Case Law and Precedents:** High-profile cases have set precedents in prosecuting Dark Web activities. **United States v. Ross Ulbricht** - The founder of Silk Road was convicted on multiple charges, including money laundering, computer hacking, and conspiracy to traffic narcotics.

6.1.2 Law Enforcement Agencies and Operations

- **International Cooperation:** Law enforcement agencies collaborate internationally to combat illegal activities on the Dark Web.
  - **Europol and Interpol:** Coordinate efforts among member countries to investigate and dismantle Dark Web marketplaces.
  - **Joint Operations:** Operations such as Operation Onymous and Operation Bayonet have led to the shutdown of major Dark Web marketplaces and arrests of key figures.
- **Technological Tools and Techniques:** Agencies employ advanced technologies to trace anonymous activities.
  - **Cyber Forensics:** Techniques used to analyze digital evidence and track illicit transactions.
  - **Blockchain Analysis:** Tracking cryptocurrency transactions to uncover illegal activities.

6.2 Ethical Implications

6.3 Anonymity vs. Accountability

- **Value of Anonymity:** Anonymity protects individuals' privacy, freedom of expression, and security, especially in oppressive regimes. **Whistleblowing and Activism** - Provides a safe platform for whistleblowers and activists to expose wrongdoing without fear of retribution.
- **Need for Accountability:** Without accountability, anonymity can be exploited for illegal activities, posing ethical dilemmas. **Criminal Exploitation** - Anonymity enables

cybercriminals to operate with reduced risk of detection.

- **Ethical Theories Applied:**
  - **Utilitarian Perspective:** Balancing the greatest good for the greatest number—protecting privacy vs. preventing harm.
  - **Deontological Ethics:** Focus on the moraions themselves, asserting that engaging in illegal activities is inherently wrong regardless of outcomes.

6.3.1 Balancing Privacy Rights with Security Needs

- **Privacy as a Fundamental Right:** Recognized by international laws and declarations as essential to individual autonomy and dignity. **Canada** - The Canadian Charter of Rights and Freedoms - Section 8 of the Charter protects privacy from unreasonable searches and seizures. **US** - Universal Declaration of Human Rights - Article 12 protects against arbitrary interference with privacy.
- **Security and Public Safety:** government responsible for protecting citizens from criminal activities that may require surveillance and regulation. **Surveillance Laws: Canada** - The Anti-Terrorism Act (2001), this act allows the Attorney General to delay notifying individuals of wiretaps for up to three years in certain circumstances. The Public Safety Act (PSA)(2002) allows private sector organizations to collect personal information without consent in certain circumstances, including when a government institution requests the information. The Security of Information Act (SOIA)(2001) criminalizes activities that may harm Canada, such as spying, economic espionage, and foreign-influenced threats. The Personal Information Protection and Electronic Documents Act (PIPEDA)(2000) requires organizations to obtain consent before collecting, using, or disclosing personal information. The Access to Information Act and the Privacy Act give Canadian citizens and permanent residents the right to access government-held information. **US** - Legislation like the USA PATRIOT Act expands government powers for surveillance to combat terrorism.
- **Ethical Balancing Act: Proportionality:** Measures taken to ensure security should be proportionate to the threats and minimally invasive to privacy. **Transparency and Oversight:** Implementing checks and balances to prevent abuse of surveillance powers.
- **Public Debate and Policy Making:** Encouraging open discussions on privacy rights and security needs to shape policies that reflect societal values.

6.4 Summary

This section examines the legal frameworks that govern the use of the Dark Web, highlighting laws related to cybercrime, illicit goods, and intellectual property. It discusses the role of law enforcement agencies in combating illegal activities through international cooperation and technological advancements. The ethical implications are explored by analyzing the tension between anonymity and accountability and the need to balance privacy rights with security needs. Ethical theories such as utilitarianism and deontology are applied to understand these complex issues. The section underscores the importance of adhering to legal statutes and engaging in ethical considerations when interacting with the Dark Web.

- Learning Objectives
  - Understand the legal frameworks governing Dark Web usage, including national and international laws, and recognize the potential legal consequences of engaging in illegal

activities.

  - Identify the roles and operations of law enforcement agencies in combating Dark Web crimes, including international cooperation and technological tools used in investigations.
  - Analyze the ethical implications of anonymity versus accountability, applying ethical theories to assess the moral complexities involved.
  - Evaluate the challenges of balancing privacy rights with security needs, understanding the importance of proportionality, transparency, and public discourse in shaping policies.

6.5 Additional Resources

- ”Anonymity, Privacy, and Security Online” https://www.eff.org/issues/anonymity
- ”Anti-terrorism Act, 2001” https://www.justice.gc.ca/eng/cj-jp/ns-sn/act-loi.html
- ”Public Safety Act, 2002 (PSA)” https://laws-lois.justice.gc.ca/eng/acts/P-31.5/
- ”Operational Standard for the Security of Information Act, 2001 (SOIA)” https://www.tbs-sct.canada.ca/pol/doc-eng.aspx?id=12323
- ”Personal Information Protection and Electronic Documents Act, 2000 (PIPEDA)” https://laws-lois.justice.gc.ca/eng/acts/p-8.6/
- ”Universal Declaration of Human Rights” https://www.un.org/en/about-us/universal-declaration-of-human-rights
- ”Uniting and Strengthening America by Providing Appropriate Tools Required to Intercept and Obstruct Terrorism (USA PATRIOT Act) Act of 2001” https://www.congress.gov/107/plaws/publ56/PLAW-107publ56.pdf

7 Conclusion

Exploring the **Dark Web** is a complex journey that intersects technology, privacy, security, ethics, and law. This course has provided a comprehensive understanding of its architecture, legitimate uses, potential risks, and the profound legal and ethical considerations it entails.

The Dark Web operates as a **double-edged sword**. On one side, it offers a haven for privacy, free speech, journalism, and activism, especially in oppressive regimes where censorship and surveillance are prevalent. Tools like the Tor network empower individuals to **communicate securely, protect their anonymity, and access information without fear of reprisal**. This underscores the fundamental human rights to privacy and freedom of expression as enshrined in documents like the Universal Declaration of Human Rights (United Nations, 1948, ttps://www.un.org/en/about-us/universal-declaration-of-human-rights).

On the other side, the Dark Web harbours illicit activities, including cybercrime marketplaces, trafficking of illegal substances and weapons, and other forms of exploitation. These activities pose significant challenges to law enforcement agencies worldwide and raise severe ethical and legal issues. The anonymity that protects dissidents and whistleblowers also shields criminals, creating a tension between the need for privacy and the imperative of security.

Throughout the course, we have emphasized the importance of understanding the risks associated with the Dark Web. Security threats such as malware, phishing attacks, and scams are pervasive. Legal risks are substantial, as engaging with illegal content or services can lead to severe consequences, including criminal prosecution. Personal safety concerns necessitate vigilant protection of personal information and awareness of potential dangers.

Creating and interacting with hidden services requires adherence to best practices for security and ethical hosting. It is crucial to comply with local and international laws, respect ethical norms, and prioritize users' safety and privacy. Virtual machines and sandboxed environments can mitigate some risks but do not eliminate the legal and moral responsibilities involved.

The course has also delved into the ethical implications of anonymity versus accountability. Ethical theories such as utilitarianism and deontological ethics provide frameworks for analyzing these dilemmas. Balancing privacy rights with security needs remains a central challenge in the digital age, requiring ongoing public discourse, transparent policymaking, and a commitment to upholding fundamental human rights.

7.1 Final Reflections

As technology continues to evolve, so too does the landscape of the Dark Web. Users, policymakers, and society must engage with these developments thoughtfully. Embracing the Dark Web's positive potentials—such as protecting privacy and enabling free speech—must be balanced against proactive efforts to combat and prevent its misuse for illegal activities.

Individuals can make informed decisions about engaging with this hidden part of the internet by fostering a deep understanding of the Dark Web's mechanics, risks, and ethical considerations. Education and awareness are crucial to responsibly navigating the Dark Web's complexities.

- Overall Learning Objectives
  - Comprehend the structure and functioning of the Dark Web and the technologies that enable its operation, such as the Tor network.
  - Identify legitimate uses of the Dark Web, including promoting privacy, free speech, journalism, whistleblowing, and activism.
  - Recognize the risks associated with the Dark Web, including security threats, legal implications, and personal safety concerns, and understand strategies to mitigate these risks.
  - Understand the legal frameworks and ethical considerations surrounding using the Dark Web, including the balance between anonymity and accountability and the tension between privacy rights and security needs.
  - Apply best practices for secure and ethical engagement with the Dark Web, ensuring compliance with legal requirements and ethical norms while protecting personal and community well-being.

8 Using the Tor Browser - Hands-On Experience

This section provides a practical guide on installing, configuring, and navigating the Tor Browser safely and responsibly. Participants will learn how to enhance their online privacy and understand the implications of various settings within the browser.

8.1 Guide to Downloading the Tor Browser

- **Access the Official Tor Project Website:**
  - Visit the official Tor Project website to ensure you download a legitimate and up-to-date version of the Tor Browser. URL: https://www.torproject.org/download/
- **Selecting the Correct Version:**
  - Choose the appropriate version of the Tor Browser for your operating system (Windows, macOS, Linux, or Android). URL: https://tb-manual.torproject.org/installation/.
- **Verifying the Download Using Digital Signatures:**
  - **Understanding the Importance of Verification:**Verifying the download ensures that the Tor Browser has not been tampered with and that you are installing an authentic version provided by the Tor Project.
  - **Steps to Verify the Download:**
    - **How can I verify Tor Browser's signature?** https://support.torproject.org/tbb/#tbb_how-to-verify-signature
- **Configuration**
  - **Adjusting Security Settings within the Tor Browser** https://tb-manual.torproject.org/security-settings/
  - **Plugins and Add-ons:** Do not install additional plugins or extensions. Plugins like Flash or extensions not provided by the Tor Project can compromise anonymity. https://www.torproject.org/download/#warning
- **Accessing .onion Websites Responsibly:**
  - **Understanding .onion Domains:** Special-use top-level domain suffix designating an anonymous hidden service reachable via the Tor network.
  - **Locating Reliable .onion Services:** Use reputable directories or official sources to find legitimate .onion sites.
  - **Avoiding Illegal Content and Recognizing Suspicious Links:** Accessing certain content may be illegal and could lead to legal repercussions.
- **Best Practices for Safe Browsing:**
  - **Keep Tor Browser Updated:** Regular updates include security patches and improvements.
  - **Avoid Downloading Files:** Downloading and opening files can expose your real IP address.
- **Understanding the Risks:**
  - **Exit Node Surveillance:** Exit nodes can potentially monitor unencrypted traffic.

8.2 Summary

This section provides a comprehensive guide on the practical aspects of using the Tor Browser to enhance online privacy and security. It covers the steps for securely downloading and installing the browser, configuring settings to balance usability and protection, and navigating the Dark Web responsibly. Emphasis is placed on verifying downloads to prevent security breaches, understanding the implications of adjusting browser features, and recognizing the importance of legal and ethical considerations when accessing content.

- **Learning Objectives:**
  - Install the Tor Browser securely by downloading it from the official website and verifying the download using digital signatures to ensure authenticity.
  - Configure the Tor Browser's security settings to enhance privacy, understanding the impact of turning on or off features like JavaScript, cookies, and plugins.
  - Navigate .onion websites responsibly, recognizing how to find legitimate sites, avoiding illegal content, and identifying suspicious links to protect against security threats.
  - Understand the legal and ethical implications of accessing content on the Dark Web and adopt best practices for safe and responsible browsing.

8.2.1 Additional Resources

- ”Download Tor Browser.” https://www.torproject.org/download/
- ”Tor Browser User Manual: Installation.” https://tb-manual.torproject.org/installation/
- ”Verifying Signatures for Windows and MacOS.” https://support.torproject.org/tbb/how-to-verify-signature/
- ”Tor Browser Verification Keys and Signatures.” https://www.torproject.org/docs/signing-keys.html.en
- ”Security Settings.” https://tb-manual.torproject.org/security-settings/
- ”Want Tor to Really Work?” https://www.torproject.org/download/download-easy.html.en#warning
- ”The Use of the Internet for Terrorist Purposes.” United Nations Office on Drugs and Crime. https://www.unodc.org/documents/frontpage/Use_of_Internet_for_Terrorist_Purposes.pdf
- ”Phishing: Don't get reeled in” Govern of Canada https://www.getcybersafe.gc.ca/en/phishing
- ”Don't Torrent Over Tor.” https://blog.torproject.org/bittorrent-over-tor-isnt-good-idea/

9 Creating a Hidden Service (Dark Web Website) - Hands-on Experience

This section provides a step-by-step guide on creating a hidden service (also known as an onion service) on the Tor network. It emphasizes understanding the technology, setting up a local web server, configuring Tor to host the service, and adhering to best practices for secure and ethical hosting.

9.1 Understanding Onion Services

9.1.1 Definition and Purpose

- **Onion Services:** Special services accessible only through the Tor network, providing anonymity to both the server and the client. They use the .onion top-level domain.
- **Anonymity for Both Parties:** Unlike regular websites, onion services hide the server's IP address, making it difficult to trace the server's physical location.

9.1.2 Legitimate Uses and Ethical Considerations

- **Privacy Protection:** Onion services enable secure communication, protecting users from surveillance and censorship.
- **Ethical Hosting:** When creating an onion service, using it for legal and ethical purposes, such as secure communication, whistleblowing platforms, or private websites, is essential.

9.1.3 Setting Up a Local Web Server

- Steps (for Debian / Ubuntu)
  - **Install Tor:**

    bash

    ```
    sudo apt update
    sudo apt install tor
    ```

  - **Configure Tor for a Hidden Service:** Edit the Tor configuration file to set up your hidden service.

    bash

    ```
    sudo nano /etc/tor/torrc
    ```

    Add or uncomment the following lines:

    torrc

    ```
    HiddenServiceDir /var/lib/tor/hidden_service/
    HiddenServicePort 80 127.0.0.1:80
    ```

    - HiddenServiceDir specifies where Tor will store your hidden service's data.
    - HiddenServicePort maps the service's virtual port to your local machine's port.

  - **Restart Tor:** Apply the changes by restarting the Tor service.

    ```bash
    sudo systemctl start tor
    OR
    sudo service tor start
    ```

  - **Retrieve Your .onion Address:** Tor generates your .onion address after starting the service.

    ```bash
    sudo cat /var/lib/tor/hidden_service/hostname
    ```

    This command displays your new .onion address (e.g., abc123def456ghi.onion).

  - **Set Up Your Web Server:**Ensure your web server is running and configured to serve content on localhost.

    ```bash
    sudo apt install nginx
    sudo systemctl start nginx
    sudo systemctl enable nginx
    OR
    sudo apt install nginx
    sudo service nginx start
    sudo service nginx status
    ```

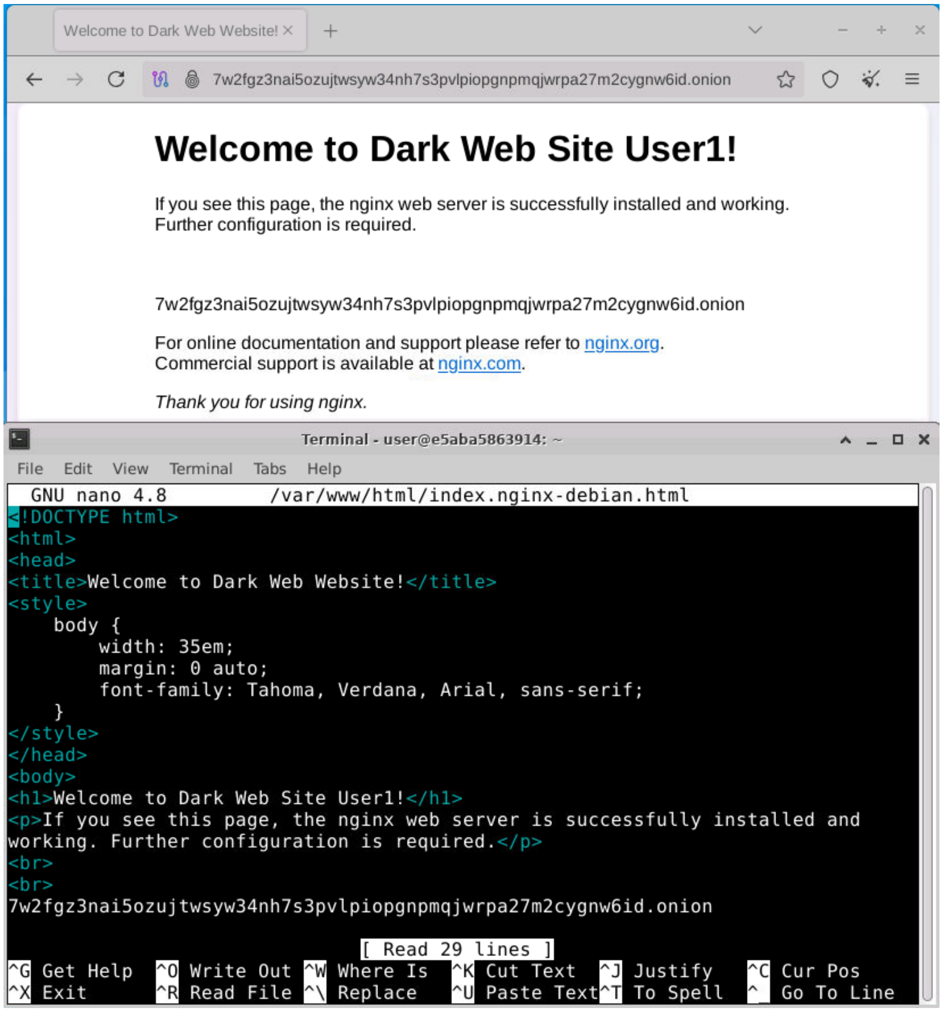
9.1.4 Test Your .onion Website

To access your website:

- Download the Tor Browser from the official website.
- Open the Tor Browser and navigate to your .onion address.

You should see your web server's default page. If you want, you can change the initial page for Ngnix. Altering the file in the following folder:

```bash
sudo nano /var/www/html/index.nginx-debian.html
```



Welcome to Dark Web Website!
7w2fgz3nai5ozujtwsyw34nh7s3pvlpiopgnpmqjwrpa27m2cygnw6id.onion
Terminal - user@e5aba5863914: ~
File Edit View Terminal Tabs Help

Welcome to Dark Web Site User1!

If you see this page, the nginx web server is successfully installed and working. Further configuration is required.

7w2fgz3nai5ozujtwsyw34nh7s3pvlpiopgnpmqjwrpa27m2cygnw6id.onion

For online documentation and support please refer to nginx.org.
Commercial support is available at nginx.com.

Thank you for using nginx.

```
  GNU nano 4.8          /var/www/html/index.nginx-debian.html
<!DOCTYPE html>
<html>
<head>
<title>Welcome to Dark Web Website!</title>
<style>
    body {
        width: 35em;
        margin: 0 auto;
        font-family: Tahoma, Verdana, Arial, sans-serif;
    }
</style>
</head>
<body>
<h1>Welcome to Dark Web Site User1!</h1>
<p>If you see this page, the nginx web server is successfully installed and
working. Further configuration is required.</p>
<br>
<br>
7w2fgz3nai5ozujtwsyw34nh7s3pvlpiopgnpmqjwrpa27m2cygnw6id.onion

                               [ Read 29 lines ]
^G Get Help   ^O Write Out  ^W Where Is   ^K Cut Text   ^J Justify    ^C Cur Pos
^X Exit       ^R Read File  ^\ Replace    ^U Paste Text^T To Spell    ^_ Go To Line
```

9.1.5 Keep Software Updated

Regularly update Tor and your web server to protect against vulnerabilities.

bash

```
sudo apt update
sudo apt upgrade
```

9.1.6 Monitor Logs (Optional)

Monitoring logs can help you troubleshoot issues.

- **Tor logs:**

  Folder

  ```
  /var/log/tor/
  ```

- **Web server logs:** Nginx (access.log and error.log)

Folder

```
/var/log/nginx/
```

9.2 Is the .onion Address Permanent?

The .onion address for your hidden service is permanent as long as the private key associated with it remains intact. Restarting Tor will not change your .onion address.

9.2.1 Understanding How the .onion Address Works

- **Private Key Generation:** When you first configure a hidden service, Tor generates a private key and stores it in the HiddenServiceDir you specified in your torrc file. The .onion address is derived mathematically from this private key using cryptographic algorithms.
- **Address Persistence:** As long as the private key file remains the same, your .onion address will stay consistent. Restarting Tor does not affect the private key or the .onion address because the service reads the existing key upon startup.

9.2.2 When Does the .onion Address Change?

Your .onion address will only change under certain conditions:

- **Deletion of the Hidden Service Directory:** If you delete or move the HiddenServiceDir, Tor cannot find the existing private key and will generate a new one upon restarting. This results in a new .onion address.
- **Modification or Corruption of the Private Key:** If the private key file (private key or hs ed25519 secret key, depending on Tor version) is altered or corrupted, the derived .onion address will change. Always ensure the integrity of your private key file.
- **Permission Issues:** may prevent Tor from accessing the private key file, leading it to generate a new one. Ensure that the Tor user (usually debian-tor) owns the directory and files

bash

```bash
sudo chown -R debian- tor debian-tor /var/lib/tor/hidden_service/
sudo chmod -R 700  /var/lib/tor/hidden_service/
```

- **Manual Regeneration:** If you intentionally generate a new private key, the .onion Address will change accordingly.

9.2.3 Best Practices to Maintain your .onion address

- **Backup the Hidden Service Directory:** Regularly back up your HiddenServiceDir, especially the private key file. This allows you to restore your service with the same .onion address in case of system failure or migration.
- **Avoid Unnecessary Changes:** Do not delete or modify files within the HiddenServiceDir unless necessary. Be cautious when changing configurations that might affect the hidden service.

- **Secure the Private Key:** Keep the private key file secure to prevent unauthorized access. Unauthorized users with access to your private key can impersonate your hidden service.

9.2.4 Creating a Vanity .onion Address

If you desire a more memorable or specific .onion address (e.g., starting with certain characters), you can create a vanity .onion address. This process repeatedly generates a new hidden service until one matches your desired pattern.

- **Understanding Vanity .onion Addresses:** Complexity: Generating vanity addresses is computationally intensive, especially for .onion v3 addresses, which are 56 characters long. The likelihood of generating a specific pattern is extremely low, making this process time-consuming. Tools Required: Specialized tools like mkp224o or Scallion are used to generate vanity addresses. However, support for .onion v3 is limited due to their complexity.
- **Using mkp224o for Vanity .onion Addresses:**
  - Ensure you have the necessary build tools and libraries.

    bash

    ```
    sudo apt update
    sudo apt install git build-essential libssl-dev libevent-dev
    sudo apt install gcc libc6-dev libsodium-dev make autoconf
    ```

  - Download and Compile mkp224o. The mp224o tool will handle the heavy lifting of generating key combinations, so let's ensure it's downloaded and installed.

    bash

    ```
    git clone https://github.com/cathugger/mkp224o.git
    cd mkp224o
    # Configure and compile
    ./autogen.sh
    ./configure
    make
    ```

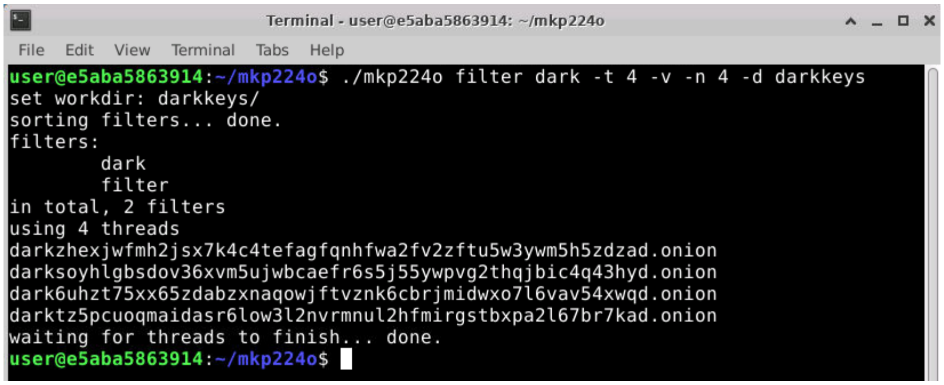
  - Creating the hostname, private and public keys for the domain: (e.g. Let's create a domain starting with ”dark”)

    bash

    ```
    ./mkp224o filter dark -t 4 -v -n 4 -d darkkeys
    cd darkkeys
    ```

    The options for the mkp224o command
    - filter dark: Look for the string dark - our vanity string
    - -t 4 : using 4 threads
    - -v : be verbose

- ∗ -n 4 : Generate 4 suggestions
- ∗ -d /folder/ : Write generated key matter out to the folder



```
user@e5aba5863914:~/mkp224o$ ./mkp224o filter dark -t 4 -v -n 4 -d darkkeys
set workdir: darkkeys/
sorting filters... done.
filters:
        dark
        filter
in total, 2 filters
using 4 threads
darkzhexjwfmh2jsx7k4c4tefagfqnhfwa2fv2zftu5w3ywm5h5zdzad.onion
darksoyhlgbsdov36xvm5ujwbcaefr6s5j55ywpvg2thqjbic4q43hyd.onion
dark6uhzt75xx65zdabzxnaqowjftvznk6cbrjmidwxo7l6vav54xwqd.onion
darktz5pcuoqmaidasr6low3l2nvrmnul2hfmirgstbxpa2l67br7kad.onion
waiting for threads to finish... done.
user@e5aba5863914:~/mkp224o$
```

./mkp224o filter dark -t 4 -v -n 4 -d darkkeys

- Change the folder to the domain you have chosen, copy the files and change ownership and the permissions

bash

```
cd darkkeys
mkdir /var/lib/tor/dark_new_domain/
cd dark[.....].onion
sudo mv * /var/lib/tor/dark_new_domain/
sudo chown -R debian-tor:debian-tor /var/lib/tor/dark_new_domain/
sudo chmod -R u+rwX,og-rwx /var/lib/tor/dark_new_domain/
```

- Update the torrc file

bash

```
nano /etc/tor/torrc
HiddenServiceDir /var/lib/tor/dark_new_domain/
```

- Restart the Tor Service:

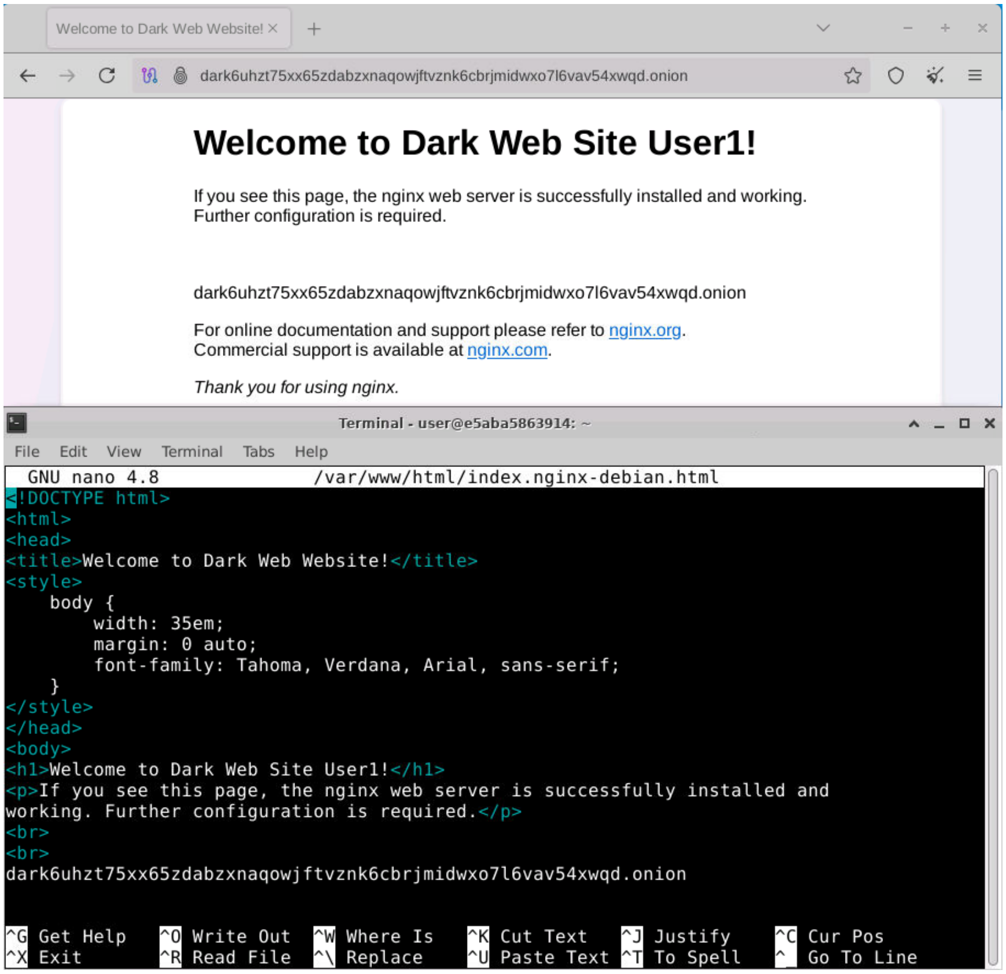
bash

```
sudo systemctl restart tor
OR
sudo service tor restart
```

- Verify the Vanity Address

bash

```
sudo cat /var/lib/tor/dark_new_domain/hostname
```


Welcome to Dark Web Website!
dark6uhzt75xx65zdabzxnaqowjftvznk6cbrjmidwxo7l6vav54xwqd.onion
Terminal - user@e5aba5863914: ~
File Edit View Terminal Tabs Help

Welcome to Dark Web Site User1!

If you see this page, the nginx web server is successfully installed and working.
Further configuration is required.

dark6uhzt75xx65zdabzxnaqowjftvznk6cbrjmidwxo7l6vav54xwqd.onion

For online documentation and support please refer to nginx.org.
Commercial support is available at nginx.com.

Thank you for using nginx.

```
  GNU nano 4.8              /var/www/html/index.nginx-debian.html
<!DOCTYPE html>
<html>
<head>
<title>Welcome to Dark Web Website!</title>
<style>
    body {
        width: 35em;
        margin: 0 auto;
        font-family: Tahoma, Verdana, Arial, sans-serif;
    }
</style>
</head>
<body>
<h1>Welcome to Dark Web Site User1!</h1>
<p>If you see this page, the nginx web server is successfully installed and
working. Further configuration is required.</p>
<br>
<br>
dark6uhzt75xx65zdabzxnaqowjftvznk6cbrjmidwxo7l6vav54xwqd.onion


^G Get Help    ^O Write Out   ^W Where Is    ^K Cut Text    ^J Justify     ^C Cur Pos
^X Exit        ^R Read File   ^\ Replace     ^U Paste Text  ^T To Spell    ^_ Go To Line
```

9.3 Summary

This section guides participants through creating a hidden service on the Tor network. It begins with understanding onion services and their purpose in providing anonymity for servers and users. Participants learn how to set up a local web server using software like Nginx, configure Tor to host an onion service by modifying configuration files, and assign a unique .onion address. The section emphasizes the best secure and ethical hosting practices, including implementing security measures, complying with legal requirements, and ensuring responsible use.

- **Learning Objectives:**
  - Understand the concept of onion services and their role in providing anonymity on the Tor network.
  - Set up a local web server using appropriate software and prepare it for hosting content accessible via Tor.
  - Configure the Tor software to host a hidden service, including modifying configuration files and obtaining a unique .onion address.
  - Apply best practices for secure and ethical hosting, ensuring compliance with legal and ethical standards while protecting the server and user privacy.

9.4 Additional Resources

- ”The Tor Browser and Using Onion Services.” https://www.eff.org/pages/tor-and-https.
- ”Tor Social Contract.” The Tor Project, https://2019.www.torproject.org/about/torusers.html.en
- ”NGINX Web Server.” https://docs.nginx.com/
- ”Tor Manual.” https://support.torproject.org/
- ”How to create a vanity Tor .onion web address” https://opensource.com/article/19/8/how-create-vanity-tor-onion-address

10 Capturing Screenshots of Dark Web Sites Without Direct Access - Hands-on Experience

This section explores methods for capturing screenshots of Dark Web sites without directly accessing them, emphasizing web archives, snapshot services, and secure environments like virtual machines and sandboxed setups. It also discusses the ethical and legal considerations associated with these practices.

10.1 Setting Up the Environment

10.1.1 Install Tor

bash

```
sudo apt update
sudo apt install tor net-tools curl -y
```

bash

```
sudo systemctl start tor
sudo systemctl enable tor
sudo systemctl status tor
OR
sudo service tor start
sudo service tor status
```

10.1.2 Test the Tor ports

bash

```
netstat -tulnp | grep tor
```

Output

```
TCP 0   0   127.0.0.1:9050  0.0.0.0:*   LISTEN  <pid>/tor
```

10.1.3 Test Tor Connectivity

Before running the crawler, ensure Tor functions correctly by making a test request.

bash

```
curl --socks5-hostname localhost:9050 https://check.torproject.org/api/ip
```

Output - json

```
{
  "IsTor": true,
  "IP": "<Your Tor Exit Node IP>"
}
```

This confirms that your requests are routed through the Tor network.

10.1.4 Install Python, PIP, and Virtual Environment

bash

```
sudo apt install python3 python3-pip python3-venv -y
```

10.1.5 Create Project Directory and Virtual Environment

bash

```
mkdir flask_tor_app
cd flask_tor_app
python3 -m venv venv
```

Activate the Virtual Environment

bash

```
source venv/bin/activate
```

10.1.6 Installing Required Python Packages

env

```
pip install --upgrade pip
pip install Flask requests[socks] validators markupsafe wkhtmltopdf
```

Explanation:

- Flask: Web framework for Python.
- requests[socks]: HTTP library with SOCKS proxy support.
- validators: Library for input validation.
- markupsafe: Provides Markup class for escaping content.

10.2 Creating the Flask Application

10.2.1 Application Structure

10.2.2 Main Application File (app.py)

Create the main application file app.py in the Flask tor app directory.

```bash
nano app.py
```

Paste the following code into app.py. Copy the code from Appendix - Code

10.2.3 HTML Template (home.html)

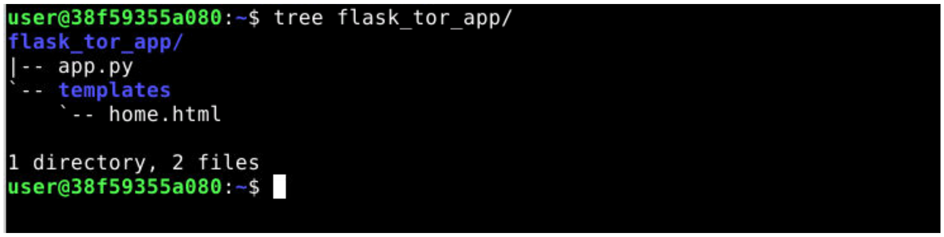
Create the templates directory and the HTML template. Create templates Directory

```bash
mkdir templates
```

Create home.html in the templates directory

```bash
nano templates/home.html
```

Paste the following code into home.html. Paste the following code into home.html. Copy the code from Appendix - Code



```
user@38f59355a080:~$ tree flask_tor_app/
flask_tor_app/
|-- app.py
`-- templates
    `-- home.html

1 directory, 2 files
user@38f59355a080:~$
```

10.3 Configuring Tor

Ensure that Tor is correctly configured to accept SOCKS5 proxy connections.

10.3.1 Verify Tor is Listening on Port 9050

```bash
ss -tuln | grep 9050
```

Expected output:

```Output
TCP     LISTEN  0   128 127.0.0.1:9050  0.0.0.0:*
```

10.3.2 Check Tor Configuration File (Optional)

If you need to modify Tor's configuration:

```bash
sudo nano /etc/tor/torrc
```

Ensure the following line is present and uncommented:

```torrc
SOCKSPort 9050
```

Restart Tor if changes are made:

```bash
sudo systemctl restart tor
OR
sudo service tor restart
```

10.4 Running the Application

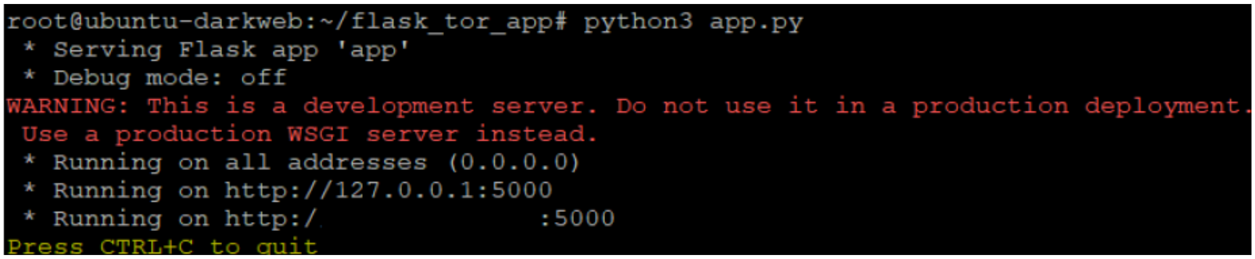
With everything set up, run the Flask application.

10.4.1 Activate Virtual Environment

```bash
source venv/bin/activate
```

10.4.2 Run the Flask Application

```bash
python3 app.py
```



```
root@ubuntu-darkweb:~/flask_tor_app# python3 app.py
 * Serving Flask app 'app'
 * Debug mode: off
WARNING: This is a development server. Do not use it in a production deployment.
 Use a production WSGI server instead.
 * Running on all addresses (0.0.0.0)
 * Running on http://127.0.0.1:5000
 * Running on http:/[redacted]:5000
Press CTRL+C to quit
```

10.4.3 Testing the Application

Access the Application in a Web Browser **Local Machine:**

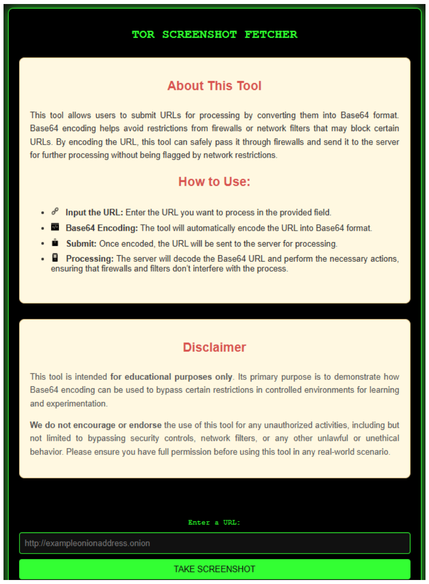
Browser

```
http://127.0.0.1:5000/
OR
http://localhost:5000/
```

Remote Server: Replace 127.0.0.1 with your server's IP address (ensure the firewall allows port 5000).

Use the Application **Enter a URL:** .onion URL: http://torlinksge6enmcyyuxjpjkoouw4oorgdgeo7ftnq3zodj7g2zxi3kyd.onion (Tor Links)

Description: TorLinks serves as a backup or secondary directory site to the popular Hidden Wiki. It's divided into commercial links (from crypto services to gambling sites) and noncommercial links (like social media). But it's not as detailed as the Hidden Wiki. As always, use Tor Links carefully, as it includes onion links to dubious or illegal activity.



TOR SCREENSHOT FETCHER

About This Tool

This tool allows users to submit URLs for processing by converting them into Base64 format. Base64 encoding helps avoid restrictions from firewalls or network filters that may block certain URLs. By encoding the URL, this tool can safely pass it through firewalls and send it to the server for further processing without being flagged by network restrictions.

How to Use:

- 🔗 **Input the URL:** Enter the URL you want to process in the provided field.
- **Base64 Encoding:** The tool will automatically encode the URL into Base64 format.
- **Submit:** Once encoded, the URL will be sent to the server for processing.
- **Processing:** The server will decode the Base64 URL and perform the necessary actions, ensuring that firewalls and filters don't interfere with the process.

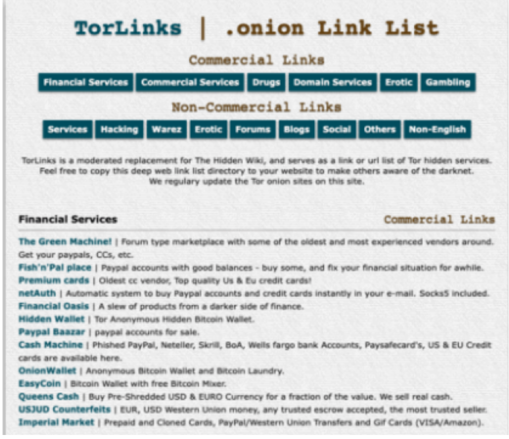
Disclaimer

This tool is intended **for educational purposes only**. Its primary purpose is to demonstrate how Base64 encoding can be used to bypass certain restrictions in controlled environments for learning and experimentation.

We do not encourage or endorse the use of this tool for any unauthorized activities, including but not limited to bypassing security controls, network filters, or any other unlawful or unethical behavior. Please ensure you have full permission before using this tool in any real-world scenario.

Enter a URL:

http://exampleonionaddress.onion



TorLinks | .onion Link List

Commercial Links

Non-Commercial Links

TorLinks is a moderated replacement for The Hidden Wiki, and serves as a link or url list of Tor hidden services.
Feel free to copy this deep web link list directory to your website to make others aware of the darknet.
We regulary update the Tor onion sites on this site.

Financial Services Commercial Links

The Green Machine! | Forum type marketplace with some of the oldest and most experienced vendors around. Get your paypals, CCs, etc.
Fish'n'Pal place | Paypal accounts with good balances - buy some, and fix your financial situation for awhile.
Premium cards | Oldest cc vendor, Top quality Us & Eu credit cards!
netAuth | Automatic system to buy Paypal accounts and credit cards instantly in your e-mail. Socks5 included.
Financial Oasis | A slew of products from a darker side of finance.
Hidden Wallet | Tor Anonymous Hidden Bitcoin Wallet.
Paypal Baazar | paypal accounts for sale.
Cash Machine | Phished PayPal, Neteller, Skrill, BoA, Wells fargo bank Accounts, Paysafecard's, US & EU Credit cards are available here.
OnionWallet | Anonymous Bitcoin Wallet and Bitcoin Laundry.
EasyCoin | Bitcoin Wallet with free Bitcoin Mixer.
Queens Cash | Buy Pre-Shredded USD & EURO Currency for a fraction of the value. We sell real cash.
USJUD Counterfeits | EUR, USD Western Union money, any trusted escrow accepted, the most trusted seller.
Imperial Market | Prepaid and Cloned Cards, PayPal/Western Union Transfers and Gif Cards (VISA/Amazon).

10.5 Summary

This section guides users in safely capturing screenshots of Dark Web sites without direct access, emphasizing security and ethical practices. Employing virtual machines and sandboxed environments is recommended to mitigate malware and unauthorized access risks. The moral and legal considerations underscore the importance of avoiding illegal content, respecting privacy, and adhering to local and international laws. Users are advised to exercise caution and seek legal counsel when necessary. With this section, You have successfully created a Flask web application that integrates with the Tor network to fetch content from user-provided URLs, including .onion links. This application demonstrates combining web technologies with privacy-focused networks to build secure and anonymous services.

- Learning Objectives
  - Recognize the importance of utilizing virtual machines and sandboxed environments for secure and isolated interaction with the Dark Web.
  - Identify the ethical and legal considerations involved in capturing and possessing screenshots of Dark Web sites, including the risks associated with illegal content.

10.5.1 Additional Resources

- ”Flask Documentation” https://flask.palletsprojects.com/"
- ”Tor Project” https://www.torproject.org/"
- ”Requests Library” https://requests.readthedocs.io/
- ”Validators Library” https://validators.readthedocs.io/
- ”MarkupSafe” http://markupsafe.palletsprojects.com/
- ”WK <html> TOpdf” https://wkhtmltopdf.org/

11 Dark Web Crawler Python Script - Hands-on Experience

Web crawling on the Dark Web presents unique challenges due to the decentralized and anonymous nature of the Tor network. This activity focuses on creating a simple yet effective crawler to index legal content on .onion sites, adhering to ethical and legal guidelines.

11.1 Setting Up the Environment

11.1.1 Install Tor

bash

```
sudo apt update
sudo apt install tor net-tools curl -y
```

bash

```
sudo systemctl start tor
sudo systemctl enable tor
sudo systemctl status tor
OR
sudo service tor start
sudo service tor status
```

11.1.2 Test the Tor ports

bash

```
netstat -tulnp | grep tor
```

Output

```
TCP 0   0   127.0.0.1:9050  0.0.0.0:*   LISTEN  <pid>/tor
```

11.1.3 Test Tor Connectivity

Before running the crawler, ensure Tor functions correctly by making a test request.

bash

```
curl --socks5-hostname localhost:9050 https://check.torproject.org/api/ip
```

Output - json

```
{
  "IsTor": true,
  "IP": "<Your Tor Exit Node IP>"
}
```

This confirms that your requests are routed through the Tor network.

11.1.4 Install Python, PIP, and Virtual Environment

bash

```
sudo apt install python3 python3-pip python3-venv -y
```

11.1.5 Create Project Directory and Virtual Environment

bash

```
mkdir tor_script
cd tor_script
python3 -m venv venv
```

Activate the Virtual Environment

bash

```
source venv/bin/activate
```

11.1.6 Installing Required Python Packages

venv

```
pip3 install requests[socks] beautifulsoup4
```

Explanation:

- requests[socks]: Allows requests to handle SOCKS5 proxies.
- beautifulsoup4: For parsing HTML content.

11.2 Python Script File (crawler.py)

Create the Python script file crawler.py in the tor_script directory.

venv

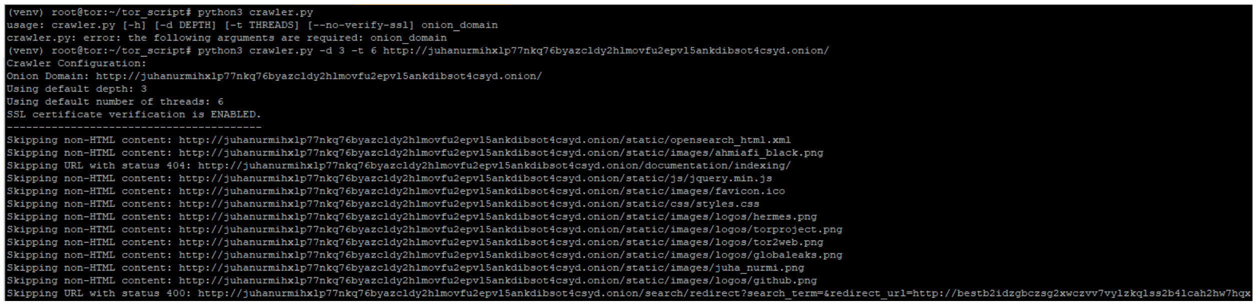
```
nano crawler.py
```

Paste the following code into crawler.py. Copy the code from Appendix - Code

11.3 Run the Python Script

venv

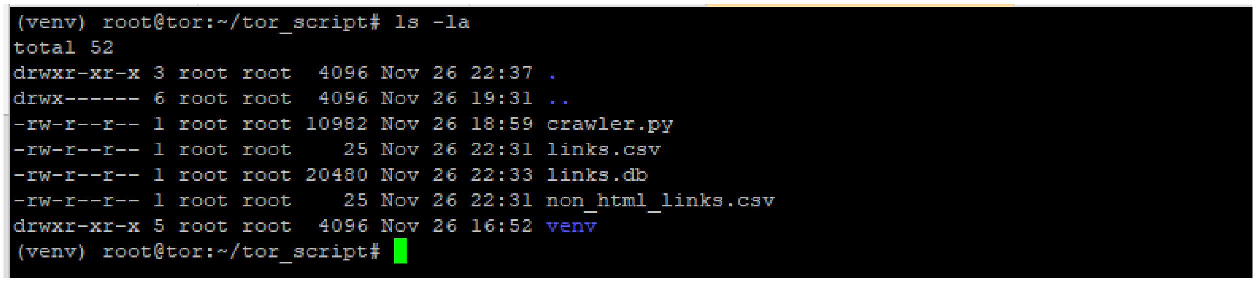
```
chmod +x crawler.py
python3 crawler.py
```



```
(venv) root@tor:~/tor_script# python3 crawler.py
usage: crawler.py [-h] [-d DEPTH] [-t THREADS] [--no-verify-ssl] onion_domain
crawler.py: error: the following arguments are required: onion_domain
(venv) root@tor:~/tor_script# python3 crawler.py -d 3 -t 6 http://juhanurmihxlp77nkq76byazcldy2hlmovfu2epvl5ankdibsot4csyd.onion/
Crawler Configuration:
Onion Domain: http://juhanurmihxlp77nkq76byazcldy2hlmovfu2epvl5ankdibsot4csyd.onion/
Using default depth: 3
Using default number of threads: 6
SSL certificate verification is ENABLED.
----------------------------------------
Skipping non-HTML content: http://juhanurmihxlp77nkq76byazcldy2hlmovfu2epvl5ankdibsot4csyd.onion/static/opensearch_html.xml
Skipping non-HTML content: http://juhanurmihxlp77nkq76byazcldy2hlmovfu2epvl5ankdibsot4csyd.onion/static/images/ahmiafi_black.png
Skipping URL with status 404: http://juhanurmihxlp77nkq76byazcldy2hlmovfu2epvl5ankdibsot4csyd.onion/documentation/indexing/
Skipping non-HTML content: http://juhanurmihxlp77nkq76byazcldy2hlmovfu2epvl5ankdibsot4csyd.onion/static/js/jquery.min.js
Skipping non-HTML content: http://juhanurmihxlp77nkq76byazcldy2hlmovfu2epvl5ankdibsot4csyd.onion/static/images/favicon.ico
Skipping non-HTML content: http://juhanurmihxlp77nkq76byazcldy2hlmovfu2epvl5ankdibsot4csyd.onion/static/css/styles.css
Skipping non-HTML content: http://juhanurmihxlp77nkq76byazcldy2hlmovfu2epvl5ankdibsot4csyd.onion/static/images/logos/hermes.png
Skipping non-HTML content: http://juhanurmihxlp77nkq76byazcldy2hlmovfu2epvl5ankdibsot4csyd.onion/static/images/logos/torproject.png
Skipping non-HTML content: http://juhanurmihxlp77nkq76byazcldy2hlmovfu2epvl5ankdibsot4csyd.onion/static/images/logos/tor2web.png
Skipping non-HTML content: http://juhanurmihxlp77nkq76byazcldy2hlmovfu2epvl5ankdibsot4csyd.onion/static/images/logos/globaleaks.png
Skipping non-HTML content: http://juhanurmihxlp77nkq76byazcldy2hlmovfu2epvl5ankdibsot4csyd.onion/static/images/juha_nurmi.png
Skipping non-HTML content: http://juhanurmihxlp77nkq76byazcldy2hlmovfu2epvl5ankdibsot4csyd.onion/static/images/logos/github.png
Skipping URL with status 400: http://juhanurmihxlp77nkq76byazcldy2hlmovfu2epvl5ankdibsot4csyd.onion/search/redirect?search_term=&redirect_url=http://bestb2idzgbczsg2xwczvv7vylzkqlss2b4lcah2hw7hqx
```

11.4 Output Files

- links.db - SQLite3 database with all the HTML links and non-HTML links
- links.csv - onions links in the database
- non_html_links.csv - non-onions links in the database



```
(venv) root@tor:~/tor_script# ls -la
total 52
drwxr-xr-x 3 root root  4096 Nov 26 22:37 .
drwx------ 6 root root  4096 Nov 26 19:31 ..
-rw-r--r-- 1 root root 10982 Nov 26 18:59 crawler.py
-rw-r--r-- 1 root root    25 Nov 26 22:31 links.csv
-rw-r--r-- 1 root root 20480 Nov 26 22:33 links.db
-rw-r--r-- 1 root root    25 Nov 26 22:31 non_html_links.csv
drwxr-xr-x 5 root root  4096 Nov 26 16:52 venv
(venv) root@tor:~/tor_script# 
```

11.5 Database Inspection

After the crawler finishes, you can inspect the links.db SQLite database to see the collected URLs. Install SQLite CLI

bash

```bash
sudo apt-get install sqlite3
```

Open table

bash

```bash
sqlite3 links.db
```

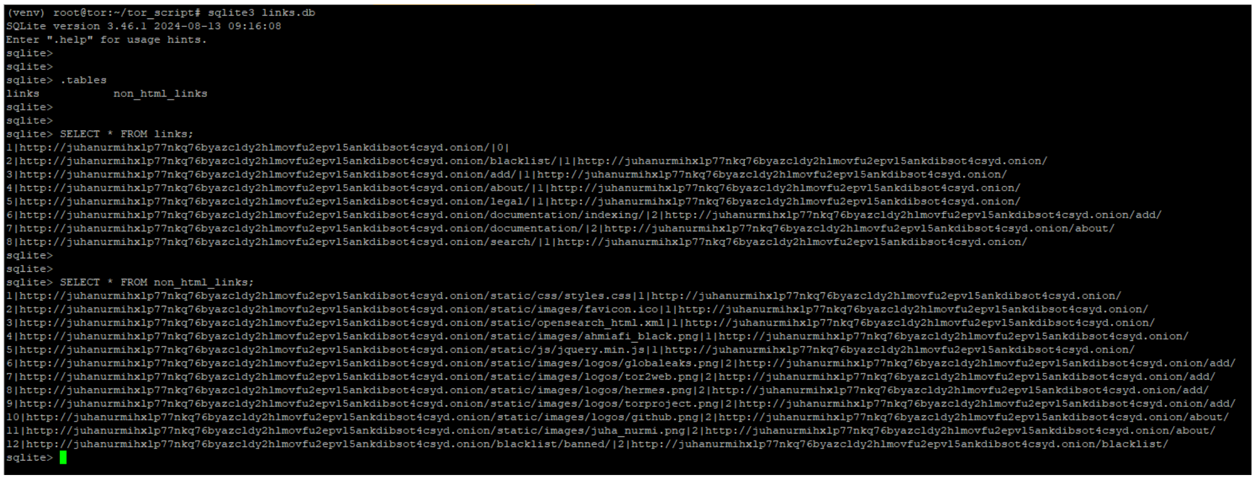
View tables

SQL

```sql
.tables
```

View the links

```sql
SELECT * FROM links;
SELECT * FROM non_html_links;
```



```
(venv) root@tor:~/tor_script# sqlite3 links.db
SQLite version 3.46.1 2024-08-13 09:16:08
Enter ".help" for usage hints.
sqlite>
sqlite>
sqlite> .tables
links           non_html_links
sqlite>
sqlite>
sqlite> SELECT * FROM links;
1|http://juhanurmihxlp77nkq76byazcldy2hlmovfu2epvl5ankdibsot4csyd.onion/|0|
2|http://juhanurmihxlp77nkq76byazcldy2hlmovfu2epvl5ankdibsot4csyd.onion/blacklist/|1|http://juhanurmihxlp77nkq76byazcldy2hlmovfu2epvl5ankdibsot4csyd.onion/
3|http://juhanurmihxlp77nkq76byazcldy2hlmovfu2epvl5ankdibsot4csyd.onion/add/|1|http://juhanurmihxlp77nkq76byazcldy2hlmovfu2epvl5ankdibsot4csyd.onion/
4|http://juhanurmihxlp77nkq76byazcldy2hlmovfu2epvl5ankdibsot4csyd.onion/about/|1|http://juhanurmihxlp77nkq76byazcldy2hlmovfu2epvl5ankdibsot4csyd.onion/
5|http://juhanurmihxlp77nkq76byazcldy2hlmovfu2epvl5ankdibsot4csyd.onion/legal/|1|http://juhanurmihxlp77nkq76byazcldy2hlmovfu2epvl5ankdibsot4csyd.onion/
6|http://juhanurmihxlp77nkq76byazcldy2hlmovfu2epvl5ankdibsot4csyd.onion/documentation/indexing/|2|http://juhanurmihxlp77nkq76byazcldy2hlmovfu2epvl5ankdibsot4csyd.onion/add/
7|http://juhanurmihxlp77nkq76byazcldy2hlmovfu2epvl5ankdibsot4csyd.onion/documentation/|2|http://juhanurmihxlp77nkq76byazcldy2hlmovfu2epvl5ankdibsot4csyd.onion/about/
8|http://juhanurmihxlp77nkq76byazcldy2hlmovfu2epvl5ankdibsot4csyd.onion/search/|1|http://juhanurmihxlp77nkq76byazcldy2hlmovfu2epvl5ankdibsot4csyd.onion/
sqlite>
sqlite>
sqlite> SELECT * FROM non_html_links;
1|http://juhanurmihxlp77nkq76byazcldy2hlmovfu2epvl5ankdibsot4csyd.onion/static/css/styles.css|1|http://juhanurmihxlp77nkq76byazcldy2hlmovfu2epvl5ankdibsot4csyd.onion/
2|http://juhanurmihxlp77nkq76byazcldy2hlmovfu2epvl5ankdibsot4csyd.onion/static/images/favicon.ico|1|http://juhanurmihxlp77nkq76byazcldy2hlmovfu2epvl5ankdibsot4csyd.onion/
3|http://juhanurmihxlp77nkq76byazcldy2hlmovfu2epvl5ankdibsot4csyd.onion/static/opensearch_html.xml|1|http://juhanurmihxlp77nkq76byazcldy2hlmovfu2epvl5ankdibsot4csyd.onion/
4|http://juhanurmihxlp77nkq76byazcldy2hlmovfu2epvl5ankdibsot4csyd.onion/static/images/ahmiafi_black.png|1|http://juhanurmihxlp77nkq76byazcldy2hlmovfu2epvl5ankdibsot4csyd.onion/
5|http://juhanurmihxlp77nkq76byazcldy2hlmovfu2epvl5ankdibsot4csyd.onion/static/js/jquery.min.js|1|http://juhanurmihxlp77nkq76byazcldy2hlmovfu2epvl5ankdibsot4csyd.onion/
6|http://juhanurmihxlp77nkq76byazcldy2hlmovfu2epvl5ankdibsot4csyd.onion/static/images/logos/globaleaks.png|2|http://juhanurmihxlp77nkq76byazcldy2hlmovfu2epvl5ankdibsot4csyd.onion/add/
7|http://juhanurmihxlp77nkq76byazcldy2hlmovfu2epvl5ankdibsot4csyd.onion/static/images/logos/tor2web.png|2|http://juhanurmihxlp77nkq76byazcldy2hlmovfu2epvl5ankdibsot4csyd.onion/add/
8|http://juhanurmihxlp77nkq76byazcldy2hlmovfu2epvl5ankdibsot4csyd.onion/static/images/logos/hermes.png|2|http://juhanurmihxlp77nkq76byazcldy2hlmovfu2epvl5ankdibsot4csyd.onion/add/
9|http://juhanurmihxlp77nkq76byazcldy2hlmovfu2epvl5ankdibsot4csyd.onion/static/images/logos/torproject.png|2|http://juhanurmihxlp77nkq76byazcldy2hlmovfu2epvl5ankdibsot4csyd.onion/add/
10|http://juhanurmihxlp77nkq76byazcldy2hlmovfu2epvl5ankdibsot4csyd.onion/static/images/logos/github.png|2|http://juhanurmihxlp77nkq76byazcldy2hlmovfu2epvl5ankdibsot4csyd.onion/about/
11|http://juhanurmihxlp77nkq76byazcldy2hlmovfu2epvl5ankdibsot4csyd.onion/static/images/juha_nurmi.png|2|http://juhanurmihxlp77nkq76byazcldy2hlmovfu2epvl5ankdibsot4csyd.onion/about/
12|http://juhanurmihxlp77nkq76byazcldy2hlmovfu2epvl5ankdibsot4csyd.onion/blacklist/banned/|2|http://juhanurmihxlp77nkq76byazcldy2hlmovfu2epvl5ankdibsot4csyd.onion/blacklist/
sqlite>
```

11.6 Summary

Building a web crawler for the Dark Web combines technical proficiency with ethical awareness. Participants will learn about Tor's unique challenges, web scraping tools, and the importance of responsible data collection.

Consideration about the web scraping tools:

- Ensure the crawler indexes only legal and ethical content. Respect the anonymity of Dark Web users by not collecting sensitive data.
- Avoid sites that host illicit material or violate privacy norms. Ensure adherence to local and international laws regarding web crawling and data collection.
- Learning Objectives
  - Understand the technical challenges of crawling the Dark Web and how to address them.
  - Gain hands-on experience with web crawling technologies, focusing on the Tor network.
  - Develop an appreciation for the ethical and legal considerations in web crawling.

11.7 Additional Resources

- ”BeautifulSoup” https://www.crummy.com/software/BeautifulSoup/
- ”SQLite3” https://docs.python.org/3/library/sqlite3.html
- ”Threading” https://docs.python.org/3/library/threading.html

12 Appendix - Code

12.1 app.py - Capturing Screenshots of Dark Web Sites Without Direct Access

Python

```python
import validators, os, time, subprocess, logging, base64, urllib.parse
from flask import Flask, render_template, request, send_file

app = Flask(__name__, static_folder='static')

# Configure logging
logging.basicConfig(level=logging.INFO)

# Path to save screenshots
SCREENSHOT_PATH = "screenshots"

# Create a screenshots directory if it doesn't exist
if not os.path.exists(SCREENSHOT_PATH):
    os.makedirs(SCREENSHOT_PATH)

@app.route('/', methods=['GET', 'POST'])
def home():
    screenshot_file = None
    error = None
    if request.method == 'POST':
        encoded_url = request.form.get('url')

        user_ip = request.headers.get('X-Forwarded-For', request.remote_addr)
        # Optionally, you can print the IP address or log it
        print(f"Request received from IP: {user_ip}")

        # Decode the base64 URL from the form
        try:
            url_bytes = base64.b64decode(encoded_url)  # Base64 decode the URL
            url = urllib.parse.unquote(url_bytes.decode('utf-8'))  # Decode URL
             ↪  encoding
        except Exception:
            error = "Error decoding URL. Please enter a valid base64-encoded
             ↪  URL."
            return render_template('home.html', screenshot=None, error=error)

        # Prepend 'http://' if the URL doesn't start with 'http://' or
         ↪  'https://'
        if not url.startswith('http://') and not url.startswith('https://'):
            url = 'http://' + url
        # Validate the URL
        if not validators.url(url):
            error = "Invalid URL. Please enter a valid URL."
            return render_template('home.html', screenshot=None, error=error)

        # Determine if the URL is a .onion address
        use_proxy = url.endswith('.onion')

        # Try to take a screenshot via wkhtmltoimage
        try:
            screenshot_file = take_screenshot(url, use_proxy)
        except Exception as e:
            error = f"Error taking screenshot: {e}"

    return render_template('home.html', screenshot=screenshot_file,
    ↪   error=error)

def take_screenshot(url, use_proxy):
    """Take a screenshot of the URL using wkhtmltoimage, optionally with Tor
    ↪   proxy."""
    # Define the output screenshot filename
    screenshot_filename = os.path.join(SCREENSHOT_PATH,
    ↪   f'screenshot_{int(time.time())}.png')

    # Construct the command for wkhtmltoimage
    command = ['wkhtmltoimage']
    if use_proxy:
        # Add proxy option only for .onion URLs
        command += ['--proxy', 'socks5://127.0.0.1:9050']
    command += [url, screenshot_filename]

    # Run the wkhtmltoimage command
    result = subprocess.run(command, stdout=subprocess.PIPE,
    ↪   stderr=subprocess.PIPE)

    # Check if the command was successful
    if result.returncode != 0:
        raise Exception(f"wkhtmltoimage failed:
        ↪   {result.stderr.decode('utf-8')}")

    return screenshot_filename

@app.route('/screenshot/<filename>')
def display_screenshot(filename):
    """Send the screenshot file to the user."""
    return send_file(os.path.join(SCREENSHOT_PATH, filename))

if __name__ == '__main__':
    app.run(debug=False, host='0.0.0.0', port=5000)
```

12.2 home.html - Capturing Screenshots of Dark Web Sites Without Direct Access

HTML

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Tor Screenshot Fetcher</title>
    <!-- Styling -->
    <style>
        body {
            font-family: 'Courier New', Courier, monospace;
            background-color: #111;  /* Dark background */
            color: #33ff33;  /* Neon green for text */
            margin: 0;
            padding: 0;}
        .container {
            max-width: 750px;
            margin: 50px auto;  /* auto for both left and right margin centers
             ↪  the container */
            padding: 20px;
            background-color: #000;
            border: 2px solid #33ff33;
            border-radius: 10px;
            box-shadow: 0 0 20px rgba(51, 255, 51, 0.7);
            text-align: center;  /* Center text and elements within the
             ↪  container */}
        h1 {
            font-size: 24px;
            text-align: center;
            color: #33ff33;
            text-transform: uppercase;}
        label {
            color: #33ff33;
            font-size: 14px;}
        input[type="text"] {
            width: 100%;  /* Make sure the input takes 100% of the available
             ↪  width */
            padding: 10px;
            margin: 10px 0;
            background-color: #111;
            border: 1px solid #33ff33;
            border-radius: 5px;
            font-size: 16px;
            color: #33ff33;
            box-sizing: border-box;  /* Ensures padding and borders are
            ↪  included in width calculation */}
        button {
            width: 100%;
            padding: 10px;
            background-color: #33ff33;
            color: #111;
            border: none;
            border-radius: 5px;
            font-size: 16px;
            cursor: pointer;
            text-transform: uppercase;}
        button:hover {
            background-color: #28cc28; /* Slightly darker neon on hover */}
        .error, .success {
            display: none;
            margin: 15px 0;}
        .error {color: red;}
        .success {color: green;}
        .screenshot-frame {
            margin-top: 20px;
            text-align: center;
            border: 1px solid #33ff33;
            border-radius: 5px;
            padding: 10px;
            background-color: #111;
            height: 100%;  /* Fixed height for the screenshot box */
            width: 100%;
            box-sizing: border-box; /* Ensures padding is included in the width
            ↪  */
            overflow: hidden;}
        .screenshot-frame img {
            max-height: 100%;
            max-width: 100%;
            display: block;
            margin: 0 auto;  /* This centers the image horizontally */
            border-radius: 5px;
            object-fit: contain; /* Ensures the image fits inside without
            ↪  stretching */
            padding-right: 10px;  /* Add padding to the right */
            padding-left: 10px;   /* Add padding to the left */}
    </style>
    <!-- JavaScript -->
    <script>
```

```
        function encodeUrl() {
            var input = document.getElementById("url");
            var encodedUrl = btoa(encodeURIComponent(input.value));  // base64
            ↪  encode the URL safely
            input.value = encodedUrl;
        }
    </script>
</head>
<body>
    <div class="container">
              <h1>Tor Screenshot Fetcher</h1>
      <!-- About This Tool Section -->
      <div style="max-width: 800px; margin: auto; margin-top: 30px; padding:
      ↪  20px; border-radius: 10px; background-color: #fff8e1; border: 1px
      ↪  solid #f7d794; box-shadow: 0 2px 10px rgba(0, 0, 0, 0.1);">
          <h2 style="font-family: Arial, sans-serif; font-size: 24px; color:
          ↪  #d9534f; text-align: center; padding-bottom: 10px;">About This
          ↪  Tool</h2>
          <p style="font-family: Arial, sans-serif; font-size: 16px;
          ↪  line-height: 1.6; color: #333; text-align: justify;">
              This tool allows users to submit URLs for processing by
              ↪  converting them into Base64 format.
              Base64 encoding helps avoid restrictions from firewalls or
              ↪  network filters that may block certain URLs.
              By encoding the URL, this tool can safely pass it through
              ↪  firewalls and send it to the server for further processing
              ↪  without being flagged by network restrictions.
          </p>
          <h3 style="font-family: Arial, sans-serif; font-size: 24px; color:
          ↪  #d9534f; text-align: center; padding-bottom: 10px"">How to
          ↪  Use:</h3>
          <ul style="font-family: Arial, sans-serif; font-size: 16px; color:
          ↪  #333; text-align: justify; padding-bottom: 10px;">
          <li style="margin-bottom: 10px;">
          <img src="https://img.icons8.com/ios-filled/16/000000/link.png"
          ↪  style="vertical-align: left; margin-right: 8px;" />
          <strong>Input the URL:</strong> Enter the URL you want to process in
          ↪  the provided field.
          </li>
          <li style="margin-bottom: 10px;">
          <img src="https://img.icons8.com/ios-filled/16/000000/code.png"
          ↪  style="vertical-align: left; margin-right: 8px;" />
          <strong>Base64 Encoding:</strong> The tool will automatically encode
          ↪  the URL into Base64 format.
          </li>
          <li style="margin-bottom: 10px;">
          <img src="https://img.icons8.com/ios-filled/16/000000/upload.png"
          ↪   style="vertical-align: left; margin-right: 8px;" />
          <strong>Submit:</strong> Once encoded, the URL will be sent to the
          ↪   server for processing.
          </li>
          <li style="margin-bottom: 10px;">
          <img src="https://img.icons8.com/ios-filled/16/000000/server.png"
          ↪   style="vertical-align: left; margin-right: 8px;" />
          <strong>Processing:</strong> The server will decode the Base64 URL
          ↪   and perform the necessary actions, ensuring that firewalls and
          ↪   filters don&rsquo;t interfere with the process.
          </li>
          </ul>
      </div>
      <!-- Disclaimer Section -->
      <div style="max-width: 800px; margin: auto; margin-top: 30px; padding:
      ↪   20px; border-radius: 10px; background-color: #fff8e1; border: 1px
      ↪   solid #f7d794; box-shadow: 0 2px 10px rgba(0, 0, 0, 0.1);">
          <h2 style="font-family: Arial, sans-serif; font-size: 24px; color:
          ↪   #d9534f; text-align: center; padding-bottom:
          ↪   10px;">Disclaimer</h2>

          <p style="font-family: Arial, sans-serif; font-size: 16px;
          ↪   line-height: 1.6; color: #555; text-align: justify;">
              This tool is intended <strong>for educational purposes
              ↪   only</strong>. Its primary purpose is to demonstrate how
              ↪   Base64 encoding can be used to bypass certain restrictions
              in controlled environments for learning and experimentation.
          </p>
          <p style="font-family: Arial, sans-serif; font-size: 16px;
          ↪   line-height: 1.6; color: #555; text-align: justify;">
              <strong>We do not encourage or endorse</strong> the use of this
              ↪   tool for any unauthorized activities, including but not
              ↪   limited to bypassing security controls, network filters,
              or any other unlawful or unethical behaviour. Please ensure you
              ↪   have full permission before using this tool in any real-world
              ↪   scenario.
          </p>
    </div>
    <br> <br> <br> <br>
        <!-- Form -->
        <form method="POST" action="/" onsubmit="encodeUrl()">
            <label for="url">Enter a URL:</label><br>
            <input type="text" id="url" name="url" required
            ↪  placeholder="http://exampleonionaddress.onion">
            <div id="error" class="error"></div>
            <button type="submit">Take Screenshot</button>
        </form>
        <!-- Display screenshot if it exists -->
        {% if screenshot %}
            <div class="screenshot-frame">
                <img src="{{ url_for('display_screenshot',
                ↪  filename=screenshot.split('/')[-1]) }}" alt="Screenshot of
                ↪  {{ url }}">
            </div>
        {% endif %}
        {% if stored_urls %}
            <h2>Previously Submitted URLs:</h2>
            <ul>
                {% for url_record in stored_urls %}
                    <li>{{ url_record.url }} - Submitted at {{
                    ↪  url_record.timestamp }}</li>
                {% endfor %}
            </ul>
        {% endif %}
    </div>
</body>
</html>
```

12.3 crawler.py - Dark Web Crawler Python Script

Python

```
import requests, sqlite3, threading, queue, time, csv, os, argparse, sys,
↪  socket, urllib3
from bs4 import BeautifulSoup
from urllib.parse import urljoin, urlparse

# Suppress only the single warning from urllib3 needed.
urllib3.disable_warnings(urllib3.exceptions.InsecureRequestWarning)

class LinkCrawler:
    def __init__(self, base_url, max_depth=3, num_threads=6, verify_ssl=True):
        self.base_url = base_url
        self.max_depth = max_depth
        self.num_threads = num_threads
        self.verify_ssl = verify_ssl
        self.visited_urls = set()
        self.visited_lock = threading.Lock()
        self.queue = queue.Queue()
        self.queue.put((self.base_url, 0, None))  # (url, depth, parent_url)
        self.conn = sqlite3.connect('links.db', check_same_thread=False)
        self.db_lock = threading.Lock()
        self.create_tables()
        self.session = self.create_session()

    def create_session(self):
        session = requests.Session()
        # Configure the session to use Tor's SOCKS5 proxy
        session.proxies = {
            'http': 'socks5h://localhost:9050',
            'https': 'socks5h://localhost:9050'
        }
        session.headers.update({'User-Agent': 'Mozilla/5.0 (compatible;
        ↪  LinkCrawler/1.0)'})
        return session

    def create_tables(self):
        with self.db_lock:
            with self.conn:
                self.conn.execute('''
                    CREATE TABLE IF NOT EXISTS links (
                        id INTEGER PRIMARY KEY,
                        url TEXT UNIQUE,
                        level INTEGER,
                        parent_url TEXT
                    )
                ''')
                self.conn.execute('''
                    CREATE TABLE IF NOT EXISTS non_html_links (
                        id INTEGER PRIMARY KEY,
                        url TEXT UNIQUE,
                        level INTEGER,
                        parent_url TEXT
                    )
                ''')

    def extract_links(self, soup, base_url):
        tags_attrs = {
            'a': 'href',
            'img': 'src',
            'script': 'src',
            'link': 'href',
            'iframe': 'src',
            'embed': 'src',
            'source': 'src',
            'track': 'src',
            'audio': 'src',
            'video': 'src',
            'object': 'data',
            'area': 'href',
            'form': 'action',
            'blockquote': 'cite',
            'q': 'cite',
            'ins': 'cite',
            'del': 'cite'
        }

        links = set()

        for tag, attr in tags_attrs.items():
            for element in soup.find_all(tag):
                url = element.get(attr)
                if url:
                    full_url = urljoin(base_url, url)
                    links.add(full_url)

        # Additionally, extract inline styles with background images
        for element in soup.find_all(style=True):
            style = element['style']
            if 'background-image' in style:
                start = style.find('url(') + 4
                end = style.find(')', start)
                if start > 3 and end > start:
                    url = style[start:end].strip('\'"')
                    full_url = urljoin(base_url, url)
                    links.add(full_url)

        return links

    def crawl_worker(self):
        while True:
            try:
                url, depth, parent_url = self.queue.get(timeout=10)  # Wait for
                ↪   10 seconds
            except queue.Empty:
                return  # Exit if no more URLs to process

            try:
                with self.visited_lock:
                    if url in self.visited_urls:
                        continue
                    self.visited_urls.add(url)

                if depth > self.max_depth:
                    continue

                try:
                    response = self.session.get(url, timeout=30,
                    ↪   verify=self.verify_ssl)
                    if response.status_code != 200:
                        print(f"Skipping URL with status
                        ↪   {response.status_code}: {url}")
                        continue

                    content_type = response.headers.get('Content-Type',
                    ↪   '').lower()
                    if 'html' not in content_type:
                        print(f"Skipping non-HTML content: {url}")
                        with self.db_lock:
                            with self.conn:
                                self.conn.execute('''
                                    INSERT OR IGNORE INTO non_html_links (url,
                                    ↪   level, parent_url)
                                    VALUES (?, ?, ?)
                                ''', (url, depth, parent_url))
                        continue

                    soup = BeautifulSoup(response.text, 'html.parser')
                    with self.db_lock:
                        with self.conn:
                            self.conn.execute('''
                                INSERT OR IGNORE INTO links (url, level, parent_url)
                                VALUES (?, ?, ?)
                            ''', (url, depth, parent_url))

                    links = self.extract_links(soup, url)
                    for link in links:
                        if self.is_same_domain(link):
                            with self.visited_lock:
                                if link not in self.visited_urls:
                                    self.queue.put((link, depth + 1, url))
                except requests.RequestException as e:
                    print(f"Error accessing {url}: {e}")
            finally:
                self.queue.task_done()

    def is_same_domain(self, url):
        base_domain = urlparse(self.base_url).netloc
        link_domain = urlparse(url).netloc
        return base_domain == link_domain

    def export_to_csv(self):
        print("Exporting data to CSV files...")

        # Export 'links' table
        links_csv = 'links.csv'
        try:
            with self.db_lock:
                cursor = self.conn.cursor()
                cursor.execute("SELECT * FROM links")
                rows = cursor.fetchall()
                headers = [description[0] for description in cursor.description]

            with open(links_csv, 'w', newline='', encoding='utf-8') as f:
                writer = csv.writer(f)
                writer.writerow(headers)
                writer.writerows(rows)
            print(f"Exported 'links' table to {links_csv}")
        except Exception as e:
            print(f"Error exporting 'links' table to CSV: {e}")

        # Export 'non_html_links' table
        non_html_csv = 'non_html_links.csv'
        try:
            with self.db_lock:
                cursor = self.conn.cursor()
                cursor.execute("SELECT * FROM non_html_links")
                rows = cursor.fetchall()
                headers = [description[0] for description in
                 ↪  cursor.description]

            with open(non_html_csv, 'w', newline='', encoding='utf-8') as f:
                writer = csv.writer(f)
                writer.writerow(headers)
                writer.writerows(rows)
            print(f"Exported 'non_html_links' table to {non_html_csv}")
        except Exception as e:
            print(f"Error exporting 'non_html_links' table to CSV: {e}")

    def start_crawl(self):
        threads = []
        for _ in range(self.num_threads):
            t = threading.Thread(target=self.crawl_worker)
            t.daemon = True
            t.start()
            threads.append(t)

        self.queue.join()  # Wait until all tasks are done

        # Optionally, wait for all threads to finish
        for t in threads:
            t.join(timeout=1)

        # Export data to CSV before closing the connection
        self.export_to_csv()

        self.conn.close()

def parse_arguments():
    parser = argparse.ArgumentParser(description='Dark Web (.onion) Link
     ↪  Crawler')
    parser.add_argument('onion_domain', type=str,
                        help='The target .onion domain to crawl (e.g.,
                         ↪  http://exampleonion.onion)')
    parser.add_argument('-d', '--depth', type=int, default=3,
                        help='Maximum crawl depth (default: 3)')
    parser.add_argument('-t', '--threads', type=int, default=6,
                        help='Number of concurrent threads (default: 6)')
    parser.add_argument('--no-verify-ssl', action='store_true',
                        help='Disable SSL certificate verification (accept
                         ↪  self-signed certificates)')

    args = parser.parse_args()

    return args

def check_tor():
    try:
        with socket.create_connection(("localhost", 9050), timeout=5):
            return True
    except OSError:
        return False

def main():
    args = parse_arguments()

    base_url = args.onion_domain
    max_depth = args.depth
    num_threads = args.threads
    verify_ssl = not args.no_verify_ssl  # If --no-verify-ssl is set,
     ↪  verify_ssl is False

    # Validate that the onion_domain is indeed an .onion URL
    parsed_url = urlparse(base_url)
    if parsed_url.scheme not in ['http', 'https']:
        print("Error: The Onion domain must start with http:// or https://")
        sys.exit(1)
    if not parsed_url.netloc.endswith('.onion'):
        print("Error: The domain provided is not a valid .onion domain.")
        sys.exit(1)

    # Notify the user if default values are being used
    default_depth = 3
    default_threads = 6
    notifications = []
    if max_depth == default_depth:
        notifications.append(f"Using default depth: {default_depth}")
    else:
        notifications.append(f"Depth set to: {max_depth}")

    if num_threads == default_threads:
        notifications.append(f"Using default number of threads: {default_threads}")
    else:
        notifications.append(f"Number of threads set to: {num_threads}")

    # Notify about SSL verification
    if not verify_ssl:
        notifications.append("SSL certificate verification is DISABLED. Accepting self-signed certificates.")
    else:
        notifications.append("SSL certificate verification is ENABLED.")

    print("Crawler Configuration:")
    print(f"Onion Domain: {base_url}")
    for note in notifications:
        print(note)
    print("-" * 40)

    # Ensure that Tor is running
    if not check_tor():
        print("Error: Tor is not running or not accessible at localhost:9050.")
        print("Please start Tor and ensure it's listening on port 9050.")
        sys.exit(1)

    # Initialize and start the crawler
    crawler = LinkCrawler(base_url, max_depth, num_threads, verify_ssl)
    start_time = time.time()
    crawler.start_crawl()
    end_time = time.time()
    print(f"Crawling completed in {end_time - start_time:.2f} seconds.")

    # Notify the user where the CSV files are located
    current_dir = os.getcwd()
    print(f"CSV files have been saved in: {current_dir}")

if __name__ == "__main__":
    main()
```