CMPT 403 Assignment 3

Mihir Rashmikant Kubavat

July 22, 2024

1 Q1

Fastest time: 1.0690104961395264 seconds to 'http://google.com'

1.1 a

1. **Selection of Fast Relays:** The Tor directory was filtered for relays marked as "fast," indicating that they have high bandwidth. This is aimed at reducing the time it takes to route data through each node in the circuit.
2. **Optimized Circuit Construction:** The circuit was manually constructed with nodes from different /16 subnets to avoid network bottlenecks that might occur when multiple nodes are on the same subnet. This spread can potentially enhance the route efficiency.
3. **Direct Use of SOCKS Proxy:** By configuring the requests library to use Tor's SOCKS proxy directly, it bypasses potential delays that might be introduced by additional handling layers or misconfigurations in system-wide proxy settings.
4. **Concurrent Connections:** While the script itself doesn't make parallel requests to avoid overwhelming the Tor network, using non-blocking I/O or asynchronous requests in a real-world application could be considered to manage multiple data streams simultaneously, potentially reducing waiting times.

1.2 b

1. **Selection of Fast Relays:**
   - **Implication:** Choosing relays labeled as fast could potentially limit the diversity of the relays used, as fewer relays meet this criterion. This reduction in randomness might make traffic patterns slightly more predictable or subject to analysis if these fast nodes are compromised or monitored.
   - **Analysis:** While faster nodes improve performance, they might slightly reduce the anonymity set because these nodes are more likely to be used by others seeking performance, thereby creating a pattern or reducing the total number of paths one's traffic might take.
2. **Optimized Circuit Construction:**
   - **Implication:** By ensuring each node is from a different /16 subnet, there's an attempt to reduce the likelihood of network-level adversaries easily correlating traffic. However, this deliberate selection might also deviate from Tor's usual random path selection, potentially affecting anonymity if the method for selecting nodes is predictable or if it reduces the diversity of path choices significantly.
   - **Analysis:** The choice to diversify the network segments of each relay is a double-edged sword; it protects against certain network observers but could also introduce a pattern in circuit construction that might be less random than Tor's default behavior.
3. **Direct Use of SOCKS Proxy:**
   - **Implication:** Directly interfacing with the SOCKS proxy ensures that the traffic is routed through Tor without leaks. However, if not configured properly, applications might bypass the proxy, leading to direct connections that expose the user's real IP address.
   - **Analysis:** This method requires careful implementation to ensure that all traffic is indeed routed through Tor and that no DNS leaks occur, as these could compromise the user's anonymity.

In summary, while methods to improve load times can enhance the user experience, they often come with trade-offs in privacy and security. Each method's implementation needs careful consideration to maintain the balance between performance and anonymity, fundamental to Tor's design philosophy.

2 Q2

2.1 a

Proposals: Differential Privacy, Private Information Retrieval (PIR)
Correct Choice: Differential Privacy
Why Suitable:
Differential privacy is ideal for this scenario because it allows the smart device company to collect aggregate data about user activity without risking individual privacy. By adding random noise to the data or using differentially private algorithms to process user inputs, the company can determine general trends like calories burned without knowing specific locations.
Why Not PIR:
Private Information Retrieval (PIR) is used when a user retrieves data from a server without revealing which data is retrieved. In this context, where the data being generated (movement data) is sensitive and the challenge is to prevent the device or app from tracking specific locations continuously, PIR is less applicable because it's more about querying existing data without disclosure, not about collecting new data securely.

2.2 b

Proposals: k-Anonymity, Private Information Retrieval (PIR)
Correct Choice: Private Information Retrieval (PIR)
Why Suitable:
PIR allows you to query a database (in this case, a DNS server) to check the availability of a domain without revealing the domain name being queried. This prevents any third party, including potentially untrustworthy DNS servers, from knowing what domain you are interested in and subsequently engaging in cybersquatting.
Why Not k-Anonymity:
k-Anonymity would involve mixing your query with the queries of at least k-1 other users to make your own query indistinguishable. In the context of domain name checks, this is not practical because you might not have control over other queries, and it doesn't inherently protect the identity of the domain being queried.

2.3 c

Proposals: k-Anonymity, Secure Multiparty Computation (SMPC)
Correct Choice: k-Anonymity
Why Suitable:
k-Anonymity can be used to ensure that the data published about patients and their locations is indistinguishable among at least k individuals. This means that any released data is generalized or suppressed enough so that individual patients cannot be re-identified, preserving privacy while still allowing researchers to analyze trends in cancer occurrences relative to building materials.
Why Not SMPC:
Secure Multiparty Computation (SMPC) is typically used to compute a function securely across multiple data holders without exposing their individual inputs. In this scenario, since the main concern is about publishing a dataset for research purposes rather than performing real-time computations with multiple stakeholders, SMPC is less relevant.

2.4 d

Proposals: Differential Privacy, Secure Multiparty Computation (SMPC)
Correct Choice: Secure Multiparty Computation (SMPC)
Why Suitable:
SMPC allows multiple parties to compute aggregate functions (like average salaries, median, etc.) without ever revealing their individual inputs (salaries in this case) to each other or to a third party. This fits perfectly for a service that compares salaries to check for fairness, as it ensures that the actual salary figures remain confidential while still allowing the necessary comparisons to be made.
Why Not Differential Privacy:
Differential privacy involves adding noise to the data, which might result in less accurate salary comparisons. While it would protect the privacy of the salary data, the added noise could compromise the service's ability to accurately determine if someone is being underpaid or not, which is critical for the service's functionality.

Each of these techniques provides a different approach to preserving privacy and must be chosen based on the specific requirements and context of the data involved.