Projekt PSI
Dokumentacja końcowa

Autorzy:

Zespół nr 34:

Igor Matynia

Wiktor Topolski

Bartłomiej Pełka

Andrii Gamalii

Treść zadania

Stworzyć prosty system anonimizujący, będący uproszczeniem rozwiązania Tor.

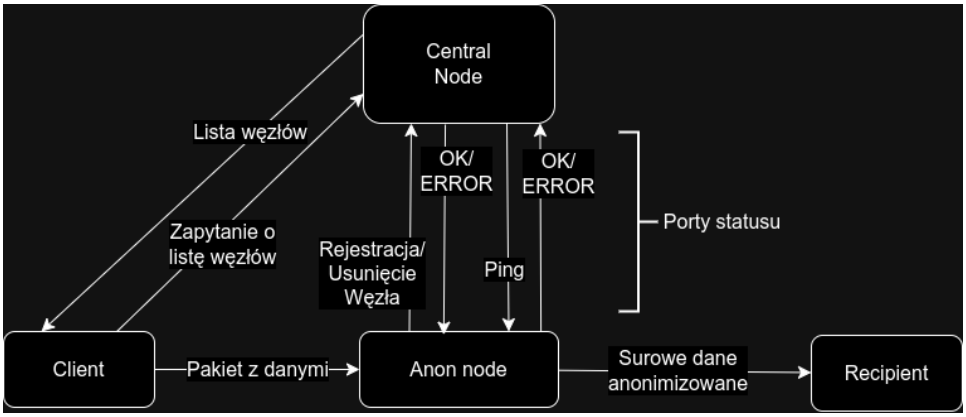
Klient pozyskuje listę węzłów pośredniczących.

Klient łączy się z wybranym węzłem pośredniczącym, tunelując ruch za pomocą wybranego rozwiązania (TLS? IPSec?). Proszę ograniczyć się do tunelowania ruchu TCP.

Węzeł pośredniczący łączy się z oryginalną lokalizacją docelową (nie korzystając z kolejnych węzłów pośredniczących jak w oryginalnym rozwiązaniu Tor).

Węzeł pośredniczący powinien wprowadzać pseudolosowe opóźnienia i dokonywać zmiany rozmiaru/segmentacji przesyłanego strumienia ruchu.

Opis projektu i implementacji



Implementację podzieliliśmy na 4 niezależne elementy - węzeł centralny, węzeł pośredniczący, klienta oraz odbiorcę. Dokładne zależności między elementami, ich zachowanie oraz komunikacja zostały opisane w dalszej części dokumentacji.

Definicje komunikatów

Elementy systemu komunikują się za pośrednictwem 2 portów - portu komunikacji oraz portu przesyłu danych. Wszystkie dane przesyłane na każdym z portów zaczynają się nagłówkiem, który składa się z danych zależnych od typu pakietu przesyłanych w reprezentacji JSON. Nagłówki są dzięki temu czytelne dla człowieka bez potrzeby dodatkowego parsera. Każdy nagłówek zawiera co najmniej jedno pole – type - definiujące typ pakietu.

Pakiety ping

Wysyłane są przez węzeł centralny do węzłów pośredniczących w celu sprawdzenia czy wciąż są aktywne – utrzymuje to aktualność listy węzłów nawet w przypadku, gdy któryś z węzłów przestanie działać.

Pakiety OK / ERROR

Przesyłają podstawową informację o powodzeniu/niepowodzeniu akcji. Pakiet error zawiera także pole content, w którym błąd może zostać opisany

Pakiety DataTransfer

W nagłówku znajdują się informacje odnośnie IP oraz portu docelowego serwera oraz wielkość przesyłanych danych. Po nagłówku wstawiany jest znak separacji (Domyślnie ustawiony na \0) po czym przesyłane są surowe dane aż do zakończenia połączenia.

Pakiety AnonNodeListRequest/Response

Pakietem Request klient może poprosić węzeł centralny o przesył wszystkich węzłów. Otrzymuje je w postaci pakietu typu Response. Pakiet zawiera pole nodes z listą deskryptorów węzłów - IP oraz porty data/status.

Pakiety Register/Unregister Node

Pakiety te używane są w celu zarejestrowania węzła pośredniego w systemie. Przesyłane są informacje o jego IP oraz portach. Te same dane potrzebne są przy odrejestrowaniu.

Scenariusze komunikacji

1. Węzeł chce się zarejestrować
    a. Węzeł pośredniczący wysyła do węzła centralnego pakiet typu RegisterNode
    b. Węzeł centralny odsyła OK/ERROR w zależności od pomyślności wykonania rejestracji
2. Węzeł chce się odrejestrować
    a. Węzeł pośredniczący wysyła do węzła centralnego pakiet typu UnregisterNode
    b. Węzeł centralny odpowiada pakietem OK/ERROR w zależności od pomyślności wykonania
3. Klient chce przesłać dane
    a. Klient prosi węzeł centralny o listę węzłów pośrednich - pakiet AnonListRequest
    b. Węzeł centralny zwraca listę pośredników pakietem AnonListResponse
    c. Klient zaczyna przesył danych do wybranego węzła pośredniczącego, rozpoczynając od przesyłu nagłówka DataTransfer
    d. Anon node odbiera dane oraz zanonimizuje je i przesyła jako surowe dane do odbiorcy
4. Klient chce otrzymać aktualną listę węzłów pośredniczących
    a. Klient prosi węzeł centralny o listę węzłów pośrednich - pakiet AnonListRequest
    b. Węzeł centralny zwraca listę pośredników pakietem AnonListResponse
5. Węzeł centralny chce sprawdzić czy węzeł pośredniczący wciąż funkcjonuje
    a. Węzeł centralny wysyła pakiet Ping do węzła pośredniczącego
    b. Węzeł pośredniczący odpowiada pakietem OK, jeśli wciąż funkcjonuje

Opis zachowania podmiotów komunikacji

Podmioty komunikacji: Nadawca, Centrala, Węzeł, Odbiorca.

Nadawca

Najpierw tworzy zapytanie do centrali o listę węzłów. Następnie do jednego, losowego węzła tworzy zapytanie z prośbą o wysłanie danych do odbiorcy

Centrala

Centrala wystawia port do nasłuchiwania. W zależności od rodzaju komunikatu:

- Rejestruje węzeł zapisując go w liście węzłów
- Wysyła listę węzłów do klienta

Dodatkowo co jakiś czas centrala odpytuje zarejestrowane węzły czy są nadal aktywne.

Węzeł

Inicjalizacja

1. Najpierw łączy się z centralą na status porcie w nowym wątku rejestruje się w centrali poprzez wysłanie pakietu typu RegisterNode i dalej trzyma to połączenie.
2. Następnie czeka na klientów na data porcie.

Obsługa klienta

1. Łączy się klient - uruchamia nowy wątek dla jego obsługi.
2. W nowym wątku czeka na informacje od klienta o tym do kogo ma transmitować dane (pakiet typu DataTransfer).
3. Po otrzymaniu DataTransfer i danych do przesłania dzieli dane na części (każda losowego rozmiaru <1024 bajtów) i przesyła je do Odbiorcy z losowym opóźnieniem (między 0,5 a 1,5 sekund).
4. Połączenie jest zamykane

Komunikacja z centralą

1. Co jakiś czas otrzymuje od centrali PingPakiet.
2. Odpowiada centrali pakietem OkPakiet.

Odbiorca

Odbiorca wystawia port do nasłuchiwania i wypisuje na terminal komunikaty o otrzymanych danych.

Podsumowanie

Zrealizowany system potrafi zanonimizować przesył danych pomiędzy klientem a odbiorcą. Ruch na każdym kroku jest szyfrowany poprzez TLS, co ogranicza możliwości podmiany danych podczas transportu i utajnia zawartość transmisji. Losowe opóźnienia i fragmentacja pakietów zapewniają trudności w zidentyfikowaniu uczestników transmisji. Prędkość przesyłu poprzez sieć jest jednak znacznie ograniczona przez wyłączenie algorytmu neagle'a oraz stosowanie arbitralnych opóźnień. Jest to wybór pomiędzy szybkością transmisji a anonimowością. Dalszy rozwój projektu mógłby wprowadzić obsługę komunikacji dwukierunkowej czy też możliwość korzystania z wielu węzłów pośredniczących. Dodanie sprawdzania certyfikatów poprzez CA zapewniłoby też autoryzację węzłów w sieci.