ДОНЕЦЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ

ІМЕНІ ВАСИЛЯ СТУСА

КАФЕДРА РАДІОФІЗИКИ ТА КІБЕРБЕЗПЕКИ

**КУРСОВА РОБОТА**

з дисципліни «Стандартизація та сертифікація в галузі інформаційної безпеки»

на тему:

Методи деанонімізації при використанні TOR технологій

Студента 3 курсу

спеціальності/напряму підготовки

б17\_д/125А Кібербезрека

Рудого Максима Вікторовича

Науковий керівник:

к.т.н. Барибін О.І.

Національна шкала \_\_\_\_\_\_\_\_\_\_\_\_

Кількість балів \_\_\_\_\_\_\_\_\_\_\_

Шкала ECTS \_\_\_\_\_\_\_\_\_\_\_

Члени комісії:

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

(підпис) (ПІБ)

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

(підпис) (ПІБ)

Вінниця 2020

**ЗМІСТ**

[АНОТАЦІЯ 2](#_Toc40051389)

[ВСТУП 3](#_Toc40051390)

[РОЗДІЛ 1. ЗАГАЛЬНІ ВІДОМОСТІ ПРО ТЕХНОЛОГІЮ TOR 4](#_Toc40051391)

[1.1 Мережа Tor 4](#_Toc40051392)

[1.2 Огляд мережі 4](#_Toc40051393)

[1.3 Осередки 6](#_Toc40051394)

[1.4 Приховані сервіси 7](#_Toc40051395)

[РОЗДІЛ 2. СХЕМИ НАПАДУ 11](#_Toc40051396)

[2.1 Проблеми браузерів 11](#_Toc40051397)

[2.2 Маніпуляції з осередками Tor 12](#_Toc40051398)

[2.3 Метод підрахунку осередків 14](#_Toc40051399)

[2.3.1 Снайперська атака 15](#_Toc40051400)

[2.3.2 Деанонімізація прихованих сервісів 16](#_Toc40051401)

[2.4 Метод заповнення осередків 17](#_Toc40051402)

[2.5 Website fingerprinting атака 19](#_Toc40051403)

[2.6 Звукові маячки 22](#_Toc40051404)

[ВИСНОВОК 24](#_Toc40051405)

[СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ 25](#_Toc40051406)

АНОТАЦІЯ

Приховані сервіси Tor використовуються для надання послуг на основі TCP користувачам, при цьому не розкриваючи IP-адресу прихованого сервера, щоб отримати прихованість і анти цензуру. Однак приховані сервіси використовуються різними способами. Приховані сервіси використовуються для незаконних цілей, таких як торгівля наркотиками, зброєю, розповсюдження порнографії і т. д. Через неправильне використання прихованих сервісів Tor, досліди для виявлення цих сервісів були неминучі.

У цій статті буде огляд існуючих атакуючих схем і пояснення їх ключових ідей.

ВСТУП

Tor - це система анонімних комунікацій з малою затримкою, яка підтримує TCP-додатки через Інтернет, які надають користувачам такі послуги, як перегляд веб-сторінок, безпечна оболонка і обмін миттєвими повідомленнями. Він надає користувачам сервіс анонімності, допомагає обійти інтернет-цензуру і підтримує приховані сервіси для збереження анонімності веб-сервісів. В даний час Tor підтримує кілька мільйонів користувачів. Tor підтримує приховані сервіси, щоб зберегти непомітність цих веб-сервісів.

Приховані сервіси - це веб-сайти, розташовані всередині мереж Tor, які отримують вхідні повідомлення тільки через Tor. Tor необхідний для доступу до прихованим сервісів. Доступ до прихованих сервісів здійснюється через його цибульну адресу, що дозволяє користувачеві публікувати інтернет-сайти, приховуючи свою IP-адресу і місце розташування.

Однак приховані служби використовуються для незаконних сервісів, таких як інформація про торгівлю наркотиками, сайти дитячої порнографії, незаконна торгівля зброєю і т. д. На прихованих сервісах Tor існує багато чорних ринків. Slik road – був один з них, він широко займався контрабандою товарів з використанням біткоїнів в якості валюти.

Наразі існує безліч різних варіантів атак, що розкривають приховані служби Tor. Розкриваючи ці приховані сервіси з’явиться змога закрити ті приховані служби які розміщують нелегальний контент. [1]

РОЗДІЛ 1. ЗАГАЛЬНІ ВІДОМОСТІ ПРО ТЕХНОЛОГІЮ TOR

* 1. Мережа Tor

Tor – це оверлейна мережа на рівні додатків, яка дозволяє здійснювати анонімний обмін даними між її клієнтами та довільними інтернет – адресами з допомогою цибулевої маршрутизації. Клієнти здійснюють анонімний зв’язок із сервером, тунелюючи свій трафік через ланцюжок із трьох ретрансляторів Tor. У цьому розділі ми спочатку ознайомимось із мережею Tor, а потім ознайомимось із її базовими операціями та протоколами прихованих сервісів.

* 1. Огляд мережі

Тор – це оверлейна мережа для анонімного зв’язку, в якій кожен цибулевий маршрутизатор (ЦМ) працює як звичайний процес на рівні користувача без будь – яких особливих привілеїв. Це проект із відкритим вихідним кодом який забезпечує сервіс анонімності для додатків TCP. Кожний (ЦМ) підтримує TLS з’єднання із кожним (ЦМ). Кожний користувач запускає локальне програмне забезпечення яке називається onion proxy (OP), для вибору каталогів, встановлення каналів в мережі і опрацювання з’єднань із клієнтськими додатками. Ці (OP) приймають TCP потоки і мультиплесують їх по каналам. (ЦМ) на іншій стороні ланцюга з’єднуються із запрошеними адресами і передають дані. [1]

Кожен цибульний маршрутизатор підтримує довгостроковий ідентифікаційний ключ і короткостроковий цибульний ключ. Ідентифікаційний ключ використовується для підпису TLS сертифікатів, дескриптора цибулевого маршрутизатора (об'єднання його ключів, адресів, політик точок виходу і т.п.), каталогів (використовується серверами каталогів).

Цибулевий ключ використовується для розшифровки запитів від користувачів на встановлення ланцюжка і для узгодження тимчасових ключів. TLS протокол також встановлює короткостроковий ключ з'єднання при спілкуванні між цибульними маршрутизаторами

Короткострокові ключі міняються періодично і незалежно один від одного, для того щоб зменшити шкоду від компрометації ключа.

Рисунок 1.2.1 ілюструє базову архітектуру мережі Tor. Наступні компоненти беруть участь в типовому використані мережі Tor.



Рисунок 1.2.1

* Tor клієнти: Клієнт Tor запитує дані для завантаження з сервера. Він встановлює локальне програмне забезпечення, onion proxy (OP), в якому дані додатки упаковуються в осередок однакового розміру (512 байт) і доставляють їх у мережу Tor. Осередок - це основний блок передачі Tor.
* Цибулеві маршрутизатори (onion router): OR - це реле, добровільно запропоноване різними волонтерами по всьому світу.
* Сервери каталогів: Сервери каталогів містять інформацію про операції OR і прихованих службах, таких як відкриті ключі маршрутизаторів і прихованих серверів.
* Сервер додатків: Він підтримує програми TCP, такі як веб-служба та служба IRC.
  1. Осередки

Цибулеві маршрутизатори спілкуються один з одним, а також з користувачами цибульних проксі через TLS з'єднання, зашифрованими тимчасовими ключами. Використання TLS захищає дані з'єднання з досконалою прямою секретністю, не дає можливості атакуючому змінити дані або видати себе за цибульний маршрутизатор. [2]

Трафік в цих з'єднаннях передається осередками фіксованого розміру. Кожний осередок має розмір 512 байт і складається з заголовка і корисної інформації. Заголовок містить ідентифікатор осередка (circID), який визначає, на який ланцюжок посилається ця група (в одному TLS з'єднанні можуть бути мультипликсовано багато ланцюжків) і команда, яка визначає що робити з корисною інформацією осередка (ідентифікатор осередка є заданим для з'єднання: кожний ланцюжок має різний circID на кожному з'єднанні «цибулевий маршрутизатор - цибулевий проксі» або «цибулевий маршрутизатор - цибулевий маршрутизатор», які вона проходить). Залежно від їх команди осередки можуть бути керуючими або передаючими. Керуючі інтерпретуються вузлами, які їх отримують. Передавальні несуть в собі дані, які потрібно передати наскрізь через мережу. Керуючі команди: вирівняти (в даний момент використовується для keepalive, може також бути використана для вирівнювання посилань); створити або створено (служить для створення нового ланцюжка); знищити (завершення ланцюжка).

Передавальні ланцюжки мають ще один заголовок (передавальний заголовок), який знаходиться перед корисною інформацією і містить streamID (ідентифікатор потоку: на один ланцюжок може бути мультиплексовано кілька потоків) і наскрізну контрольну суму для перевірки цілісності, довжину переданої корисної інформації і передану команду. Весь вміст переданого заголовка і переданої корисної інформації осередку шифрується і дешифрується разом кожен раз, коли передавальний осередок рухається по ланцюжку. Для цього використовується 128 бітний AES шифрувальник, що працює в режимі рахівника для генерації шифрованого потоку. Команди передачі: передати дані, почати передачу даних (для відкриття потоку), закінчити передачу даних (для коректного закриття потоку), завершити передачу даних (для закриття зіпсованого потоку), передачу даних розпочато (для оповіщення цибульних проксі, що початок передачі був успішним), розширити ланцюжок і ланцюжок розширено (для розширення ланцюжка за один перехід і для отримання підтвердження), скоротити ланцюжок і ланцюжок скорочено (для того, щоб завершити тільки частину ланцюжка, і щоб послати підтвердження), передати самому собі (слугує для контролю перевантаження) і перервати передачу даних (служить для створення пустишок на довгі дистанції).

* 1. Приховані сервіси

Tor забезпечує анонімність для веб-сайтів та інших серверів. Сервери, призначені для прийому вхідних з'єднань тільки через Tor, називаються прихованими сервісами. Замість того, щоб розкривати IP-адресу сервера, до прихованого сервісу звертаються через його цибульний адрес, який є хешем його відкритого ключа. Прихований сервіс Tor складається з наступних важливих вузлів:

* Introduction Point (IP): Це реле Tor, які вибираються прихованими сервісами і використовуються для встановлення зв'язку з клієнтом.
* Rendezvous Poin (RPO): вони вибираються клієнтом Tor і використовуються для пересилання даних між клієнтом і прихованим сервером.
* Directory Server (DS): DS володіє інформацію про вузли мережі Tor, і ця інформація використовується для зв'язку з прихованими сервісами.
* Hidden server (HS): це сервер всередині мережі Tor, на якому розміщені приховані сервіси.

Принцип роботи прихованої служби заснований на підключенні двох ланцюгів, одного створеного клієнтом, а іншого - HS, на загально узгодженому реле Tor. Це реле Tor, таким чином, називається Rendezvous Poin, яка передає осередок шляхом пересилання вихідних повідомлень із схеми на стороні клієнта до ланцюга на стороні сервера та навпаки. Щоб захистити RPO від атак, прихована служба вибирає групу реле Tor як точки введення, які працюють так само, як RPO. IP-адреси використовуються виключно для передачі одного повідомлення, що містить розміщення обраного RPO. Що стосується лише прийняття запитів клієнта, прихована служба публікує прихований дескриптор служби, що містить підписаний список IP-адрес для DS, звідки клієнти можуть їх завантажувати. [1]

Нормальне налаштування зв'язку між клієнтом та конкретним HS показано на рисунку 1.4.1. Процедура цього повідомлення полягає в наступному:

1. HS спочатку вибирає декілька OR в якості IP і будує схеми до цих IP, надсилаючи осередок RELAY\_COMMAND\_ESTABLISH\_INTRO, а IP у відповідь відповідає осередоком RELAY\_COMMAND\_INTRO\_ESTABLISHED, щоб повідомити HS про те, що схема встановлена.
2. Як тільки схеми до IP встановлені, HS встановлює ланцюг до DS та оголошує дескриптор служби DS, включаючи відкритий ключ HS та інформацію щодо IP. Тоді власник HS може розмістити цибулеву адресу у громадському місці, щоб залучити користувачів до доступу до прихованого сервісу через Tor.
3. Коли клієнт Tor отримує цибулеву адресу, клієнт створює схему DS і отримує відповідну інформацію, оголошену прихованою службою. Потім клієнт дізнається IP прихованої служби.
4. Тепер клієнт вибирає RPO і створює схему для RPO. Клієнт відправить осередок RELAY\_COMMAND\_ESTA-BLISH\_RENDEZVOUS, який містить cookie-файл рандеву, і RPO який відповідає осередоком RELAY\_COM-MAND\_RENDEZVOUS\_ESTABLISHED, щоб підтвердити успішне встановлення каналу.
5. Клієнт створює схему трьох стрибків для одного з IP і передає осередок RELAY\_COMMAND\_INTRODUCE1 на обраний IP. В команді зберігається така інформація, як RPO, cookie-файли рандеву і дані Діффі-Хеллмана які генерується клієнтом Tor.
6. Як тільки IP отримує осередок RELAY\_COMMAND\_INTRODUCE1, він відповідає клієнтові RELAY\_COMMAND\_INTRODUCE\_ACK. Після того, як клієнт отримає цей осередок ACK, він розриває цю схему до IP.
7. IP перепаковує осередок RELAY\_COMMAND\_INTRODUCE1 у осередок RELAY\_COMMAND\_INTRODUCE2, а потім відправляє осередок RELAY\_COMMAND\_INTRODUCE2 в HS. Як тільки HS отримує цей осередок, він знає інформацію про RPO, файли cookie та дані Діффі-Хеллмана . HS може генерувати дані Діффі-Хеллмана і отримати ключ K = .
8. Тоді HS будує ланцюг до RPO та посилає в RPO осередок RELAY\_COMMAND\_RENDEZVOUS1 до RPO. Коли RPO отримує цей осередок, він порівнює переглянуті cookie з осередком та те, що йде від клієнта Tor. Після того, як файли cookie, що переглядаються, збігаються, RPO видаляє файли cookie, що переглядаються, з осередоку RELAY\_COMMAND\_ RENDEZVOUS1 і перепаковує решту даних у осередок RELAY\_COMMAND\_RENDEZVOUS2, а потім пересилає осередок клієнту.
9. Коли клієнт Tor отримує осередок RELAY\_COMMAND\_RENDEZVOUS2, він може генерувати ключ K = ., використовуючи , і перевірити його на основі H (K). Таким чином клієнт та HS завершують рукостискання. Потім клієнт відправляє осередок RELAY\_COMMAND\_BEGIN для встановлення потоку в HS через шість стрибків.
10. Коли клієнт відправляє осередок RELAY\_COMMAND\_BEGIN до HS, щоб відкрити потік між клієнтом і HS, наш контрольований RPO виявляє цей спеціальний осередок на основі прихованого протоколу обслуговування

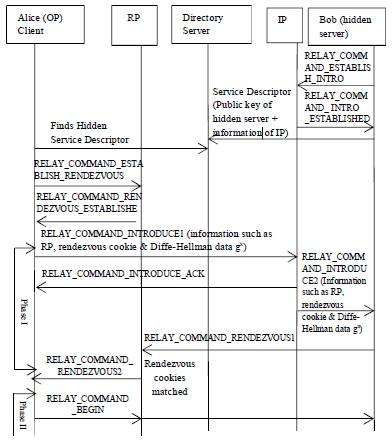


Рисунок 1.4.1

РОЗДІЛ 2. СХЕМИ НАПАДУ

Прихований сервіс надає комунікатору непомітність для постійних послуг. Користувачі можуть підключитися до сервісу через Tor, не знаючи його місцезнаходження. Ці приховані сервіси можуть розміщувати незаконні веб-сайти, наслідки яких можуть бути серйозними. Тож ці HS потрібно розкрити.

Деанонімізація прихованої служби - це стратегія, в якій розкривається ip-адреса анонімних веб-сайтів. Один із способів деанонімізувати ці приховані служби - це напад на них. До цього часу дослідники розробили безліч різних схем атаки, про які йде мова нижче.

1. Проблеми браузерів

З розсекречених документів NSA можна дізнатися, що спецслужби не гребують використовувати експлойти до браузеру Firefox, на базі якого побудований Tor Browser. Однак використання методів експлуатації вразливостей, як пишуть у своїй же презентації NSA, не дозволяє здійснювати постійне стеження за користувачами даркнета, так як життєвий цикл експлойтів дуже короткий, і існування різних версій браузера (що містять конкретну вразливість і не містять її) ставлять під удар дуже вузьке коло користувачів. [3]

Крім псевдо офіційних документів Tor - спільнота знає багато інших, більш цікавих і хитрих атаках на клієнтську сторону. Так, наприклад, дослідниками з Массачусетського Технологічного Інституту було встановлено, що Flash створює виділений канал комунікації між спеціальним сервером зловмисника, який фіксує реальний IP-адресу клієнта, і жертвою, що повністю її дискредитує. Однак розробники Tor Browser оперативно відреагували на дану проблему, виключивши обробники Flash-контенту зі свого дітища (рисунок 2.1.1).

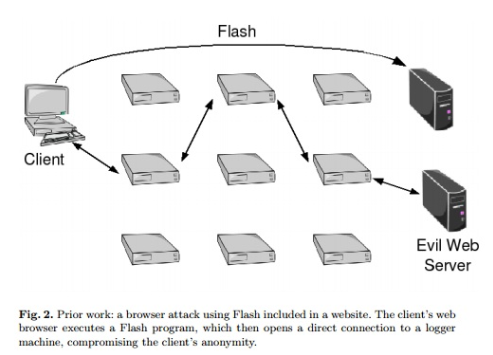


Рисунок 2.1.1

Для реалізації іншого, більш свіжого способу компрометації браузера використовується бібліотека WebRTC. Вона призначена для організації каналу передачі відеопотоку між браузерами з підтримкою HTML5, і за аналогією з вищеописаним Flash раніше дозволяла встановити реальний IP-адресу жертви. Так звані STUN-запити WebRTC йдуть в незашифрованому вигляді в обхід Tor з усіма витікаючими наслідками. Однак і це «непорозуміння» також було оперативно виправлено розробниками Tor Browser, який тепер за замовчуванням блокує WebRTC.

1. Маніпуляції з осередками Tor

Цей метод заснований на підході виявлення на рівні протоколу. Зловмисник управляє клієнтом Tor, RPO, кількома вхідними OR та центральним сервером. За допомогою цього методу легко розгорнути систему виявлення. [1]

Порівняно з методами, заснованими на аналізі трафіку, цей підхід є значно швидшим, повністю автоматичним і може швидко знайти HS, використовуючи лише кілька осередків.

Цей підхід є точним зі спостережуваною частотою виявлення 100% та має низьку похибку 0%. Він працює на рівні протоколу і не звертає уваги на схеми трафіку, тобто він є більш загальним і може використовуватися для виявлення шкідливих прихованих служб.

У цьому методі процес виявлення HS поділяється на три фази. У фазі 1 клієнт, імовірно, ідентифікує HS, як показано на рисунку 2.2.1. Після ідентифікації HS, у фазі 2 перевіряється HS. Наприкінці 1 фази клієнт відправляє осередок RELAY\_COMMAND\_BEGIN в HS. Коли RPO ловить цей осередок, він змінює один біт комірки і пересилає її до HS. Модифікація комірки в RPO показана на рисунку 2.2.1. RPO також повинен надіслати часову мітку керованої комірки на центральний сервер. Через відсутність перевірки цілісності інші OR не можуть виявити керований осередок.

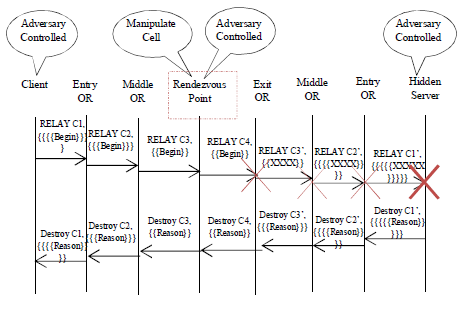


Рисунок 2.2.1

Коли маніпульований осередок досяг HS, HS не може розпізнати цей осередок, і ланцюг між клієнтом і HS розірветься, відправивши осередок CELL\_Destroy. Контрольований запис OR спочатку отримує цей осередок, і він повідомляє про тип осередку, часову мітку осередка, ідентифікатор схеми та IP-адресу джерела осередка на центральний сервер. Знищити осередок, який досягає клієнта через RPO. У RPO потрібно повідомити центральний сервер про часову мітку цього осередку. Нарешті, за часовою кореляцією ми можемо знайти IP-адресу HS.

У цьому способі прихована служба повинна вибрати вузол введення противника як свій захисний вузол. Щоб це сталося, кількість компрометованих вузлів входу повинна бути дуже великою, майже 30% від всіх вхідних маршрутизаторів Tor, що з фінансової сторони дуже дорого.

1. Метод підрахунку осередків

У цьому методі довільні реле Tor відключаються анонімно та вибірково з дуже низькими витратами для нападника, ця атака ефективна настільки що зловмисник, зацікавлений в цензурі може вимкнути Tor замість того щоб блокувати Tor простим вимкненням всіх реле або лише орієнтуючись на ті що забезпечують найбільшу пропускну здатність чи найважливіші служби каталогів. Ця атака також відома як снайперська атака, вона класифікується як атака з підрахунком осередків, оскільки вона деанонімізує прихований сервіс, підраховуючи кількість осередків в противника. [1]

У цьому методі будуть деанонімізовані приховані служби, вибірковим відключенням реле за допомогою снайперської атаки, сильно впливаючи на шляхи до тих, хто контролює противника. Таким чином, ця атака створює реальні, значні загрози для користувачів Tor, і це є найбільш руйнівною атакою проти мережі Tor на сьогодні.

1. Снайперська атака

Снайперська атака виконується з метою вбити будь-яке довільне реле. На рисунку 2.3.1.1 показані етапи цієї снайперської атаки:

* Клієнт створює схему, використовуючи ціль як запис.
* Клієнт запитує дані з сервера, а сервер відповідає.
* Клієнт перестає читати дані з цільового запису.
* Клієнт періодично надсилає пакет SENDME на сервер, проте він не може прочитати цільові записи з TCP з’єднання.
* Цільовий запис буферизує дані до тих пір, поки ОС не припинить процес Tor

Оскільки цей метод вимагає, щоб цільовий вузол входу прихованої служби припинявся буферизацією всієї вільної пам'яті, для системи з великою кількістю вільної пам'яті може знадобитися багато часу. [1]

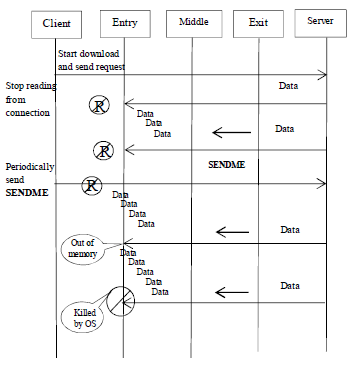


Рисунок 2.3.1.1

1. Деанонімізація прихованих сервісів

Блок-схема процесу деанонімізації прихованих служб показана на рисунку 2.3.2.1

* Перш за все клієнт – зловмисник змушує HS будувати нові канали рандеву для того щоб визначити його захисну систему.
* Примусово змусити HS змінити захисну систему.
* Повторювати поки HS не обере ворожу систему

Спочатку захисник противника запускає одне захисне реле (). Для деанонімізації прихованого сервісу, HS повинен вибрати одне з ворожих реле як захисний вузол. Для цього HS будує нову схему в RP. Зловмисник відправляє 50 нових осередків в HS через . Якщо лічильник осередків в захисному ретрансляторі, запущеному клієнтом – противником дорівнює 52, то зловмисник може ідентифікувати вхідний вузол входу HS. Потім на вхідний вузол HS здійснюється снайперська атака, і HS змушений обрати інший вузол. Цей процес повторюється до тих пір поки HS не обере ворожий вузол. Щоб перевірити обрав HS скомпрометований вузол чи ні, противник – клієнт надсилає 50 додаткових осередків в HS. Якщо лічильник осередків в захисті противника дорівнює 53, то атакуючий може ідентифікувати приховану службу. [1]

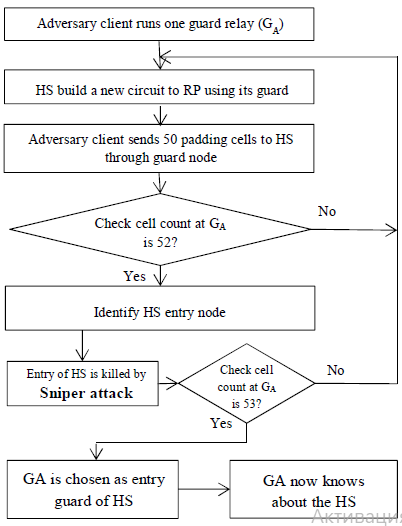


Рисунок 2.3.2.1

1. Метод заповнення осередків

У цій атаці прихована служба змушена встановлювати контрольовані зловмисником ланцюги зустрічі до RP. Після отримання осередку RELAY\_COMMAND\_RENDEZVOUS1 з файлом cookie зловмисника, RP генерує трафік зі спеціальною сигнатурою. Ця сигнатура може бути ідентифікована середнім вузлом нападника. Зауважте, що спеціальний механізм осередку PADDING у Tor спрощує генерацію трафіку сигнатури, який відкидається на стороні одержувача, і, таким чином, не помітний для HS.

Кроки атаки як на рисунку 2.4.1

1. Спочатку зловмисник надсилає осередок RELAY\_COMMAND- \_INTRODUCE1 до IP HS із зазначенням адреси RPO.
2. IP передає вміст у осередку RELAY-COMMAND INTRODUCE2 для прихованої служби.
3. HS встановлює трипроменеву схему до RPO, надсилаючи осередок RELAY\_COMMAND\_RENDEZVOUS1.
4. Коли RPO, керований зловмисником, отримує осередок RELAY- \_COMMAND\_RENDEZVOUS1, він надсилає 50 осередків PADDING до HS тим же шляхом.
5. Коли HS отримує цей осередок PADDING, ці осередки HS мовчки скидаються.
6. RPO посилає осередок DESTROY через ланцюг рандеву, що веде до закриття ланцюга.

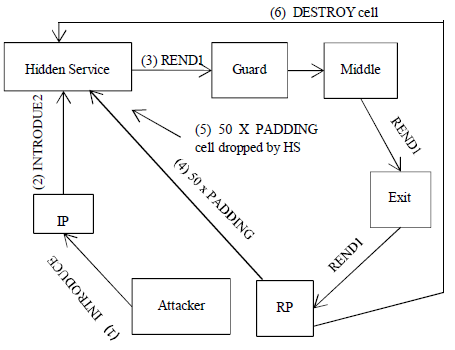


Рисунок 2.4.1

Охоронний вузол зловмисника стежить за переміщенням трафіку на ланцюгах, що проходять через нього. Щоразу, коли він отримує осередок DESTROY по ланцюгу, він перевіряє: (1) чи був осередок отриманий одразу після того, як RPO отримав осередок RELAY\_COMMAND\_REN DEZVOUS1, (2) кількість перенаправлених осередків: 3 осередка вгору по ланцюгу і 53 осередка вниз по ланцюгу. Якщо ці умови виконуються, то зловмисник розуміє, що для прихованої служби був обраний його охоронний вузол. [1]

1. Website fingerprinting атака

Ця атака привертає велику увагу дослідників, оскільки дозволяє деанонімізувати користувачів володіючи досить скромними ресурсами, а саме - корумпованим вхідним вузлом (рисунок 2.5.1). WF атака - окремий випадок атак аналізу трафіку, під час яких локальний спостерігач намагається з'ясувати інформацію про контент (відвіданий веб-сайт), оглядаючи потоки трафіку всередині анонімного з'єднання. Атакуючий збирає метаінформацію: розмір пакету, напрямок трафіку. Збір відбувається без порушення шифрування. [4]

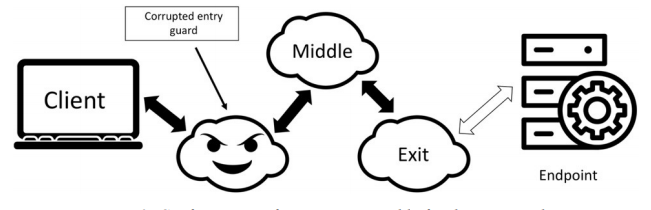


Рисунок 2.5.1

Стратегія атакуючого досить проста: він збирає мережевий трафік через вхідний вузол, а потім посилає його в класифікатор, який навчається на цих даних. Після того, як класифікатор навчився, він готовий до експлуатації. Класифікатор може бути навчений як мультикласовий (розпізнавати сайти), так і в якості бінарного: переглядає користувач сайт зі списку атакуючого або з зовнішнього світу.

Однак, потрібно зробити кілька припущень, за допомогою яких можна спростити атаку:

* Існує тільки k сторінок, які користувач може відвідати. Це дуже сильне припущення, оскільки це число завжди дуже маленьке, в порівнянні з дійсним числом існуючих сторінок.
* Поведінка користувача досить специфічна. Наприклад, він відвідує одну сторінку за однією в одній вкладці браузера.
* Атакуючий може зрозуміти, де початок і кінець завантаження сторінки в послідовності трафіку. Це досить важка задача, враховуючи трафік з реального світу.
* Атакуючий може фільтрувати весь фоновий трафік, який виробляють сторонні додатки або інші сполуки через ланцюг Tor.
* Атакуючий може тренувати класифікатор в тих же умовах, що і жертва. Наприклад, у них повинна бути однакова версія браузера Tor.

Основоположним принципом атаки є використання алгоритмів машинного навчання. Тому постає питання, що використовувати в якості навчальних даних для алгоритму, оскільки до кінця не зрозуміло, що означає «послати трафік в класифікатор». Розглянемо, що ж використовується в якості вхідних даних для алгоритму.



Рисунок 2.5.2

Найбільш важливим компонентом для будь-якого алгоритму машинного навчання є ознаки об'єктів. Дістати ознаки для них можна на різних рівнях: рівень осередків Tor, TLS, TCP. На рівні додатку, Tor передає всі дані в осередках які потім перетворюються в TLS записи як показано на рисунку 2.5.2. Примітно, що багато осередків може бути упаковано в однин TLS запис. І останній - транспортний рівень: TLS записи, як правило, фрагментируются в кілька TCP пакетів, розмір яких обмежений максимальним розміром пакету даних. Крім того, кілька TLS записів можуть бути поміщені в один TCP пакет. Однак, постає дуже важливе питання: який з шарів містить найбільше інформації, з точки зору атаки? Більшість дослідників схиляється до того, що найбільш інформативним є рівень осередків.

Однак, існує ряд проблем, які ускладнюють WF атаки:

* Розмір реального світу дуже великий, тому необхідно збирати гігантську кількість даних, щоб покривати хоча б невелику частину інтернету.
* Оскільки вихідні вузли розкидані по всьому світу, а у більшості сайтів є хороша локалізація, то можна зібрати під однією назвою кілька версій одного і того ж сайту на різних мовах і з різними контентом, що істотно знизить точність класифікатора.
* Точного способу виділяти послідовність пакетів, що стосуються тільки одного запиту на сторінці досить проблематично, завдання залишається невирішеним. Якщо користувач не використовує фіксований набір вхідних вузлів, то він може просто припинити використовувати корумпований вхідний вузол

Підводячи підсумки, WF атака може бути використана як в якості моніторингового, розвідувального спостерігача (чи відвідує користувач ресурси зі списку чи ні), так і в розпізнавальному варіанті (який портал відвідує користувач зі списку).

1. Звукові маячки

Василіос Маврудіс, один з дослідників, які виявили цю проблему, в своєму виступі на 33-му Chaos Communication Congress докладно описав метод деанонімізація користувачів Tor, який дозволяє вирахувати їх справжню IP-адресу і деяку іншу інформацію (рисунок 2.6.1). [5]

Успіх цього способу вимагає, щоб користувач Tor зайшов на сторінку, яка містить рекламу яка генерує ультразвук, або ж прихований Javascript-код, який змушує браузер робити те ж саме через HTML5 Audio API.

Якщо телефон користувача Tor знаходиться десь поблизу, і на ньому встановлені деякі програми, його телефон пошле одній або декільком рекламним компаніям повідомлення, що містить інформацію про його пристрої, щоб рекламна компанія могла доповнити наявний у неї профіль користувача, зв'язавши його комп'ютер з його телефоном .

За словами Маврудіса, щоб це спрацювало, на мобільному пристрої має бути встановлено додаток, в який вбудований один з безлічі рекламних комплектів для розробки програмного забезпечення.

Під час своїх дослідів, Маврудісу вдалося перехопити частину інформації, якою ці ультразвукові маячки обмінюювалися через телефон. Вона містила такі подробиці, як справжню IP-адресу користувача, геолокаційні координати, номер телефону, ID пристрою Android, серійний номер IMEI і MAC-адресу пристрою.

Даний спосіб деанонімізація може застосовуватися з різних сторін. За словами Маврудіса, способи деанонімізація людини по даній методиці не обмежуються лише тим, щоб обманом змусити користувачів Tor відвідувати сайти, що містять такі ультразвукові маячки. Дослідники стверджують, що зловмисники можуть використовувати уразливості в XSS (Міжсайтовий Скриптинг), щоб вставити свій JavaScript-код на веб-сторінку, схильну до злому через XSS.

Ще більш простий спосіб - сховати ультразвуковий запис, нечутний для людського вуха, в відео або аудіо файли, які переглядають або прослуховують певні користувачі Tor.



Рисунок 2.6.1

ВИСНОВОК

Приховані сервіси всередині мережі Tor - це двосічний меч. З одного боку, він зберігає анонімність постачальника послуг, а з іншого - технічно захищає зловмисних користувачів та організації, які розміщують нелегальний вміст, наприклад, інформацію про торгівлю наркотиками та зброєю, дитячу порнографію.

В роботі були розглянуті деякі методи, які дозволяють успішно провести деанонімізацію користувачів Tor. Різноманітні типи атак направлені на деанонімізацію користувачів, уже давно відомі і їх теоретична основа детально описана. Тобто повна анонімність головного тіньового браузера це міф.

Із розглянутих методів можна зробити однозначний висновок: атакуючому, який має багато ресурсів, наприклад державним службам, приватним корпораціям і т.д., не складе великих труднощів деанонімізувати велику кількість користувачів Tor. Проте важливим фактором є зацікавленість в тому, щоб знайти того чи іншого користувача, а також готовність профінансувати цей пошук.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Deanonymizing schemes of hidden services in tor network

URL:<https://www.researchgate.net/publication/278021842_Deanonymizing_schemes_of_hidden_services_in_tor_network_A_survey> (дата звернення 30.04.2020)

1. Архитектура Tor или как на самом деле работает анонимная сеть Tor

URL: <http://www.spy-soft.net/arxitektura-tor-ili-kak-na-samom-dele-rabotaet-anonimnaya-set-tor/> (дата звернення 30.04.2020)

1. Tor: полная деанонимизация. Фингерпринтим пользователей с помощью системы активного мониторинга и не только

URL: <https://xakep.ru/2015/06/25/tor-197/> (дата звернення 05.5.2020)

1. ТЕХНОЛОГИИ ДЕАНОНИМИЗАЦИИ ПОЛЬЗОВАТЕЛЕЙ TOR

URL: <https://cyberleninka.ru/article/n/tehnologii-deanonimizatsii-polzovateley-tor> (дата звернення 07.05.2020)

1. Звуковые маячки - угроза анонимности

URL:<https://www.pf.team/articles/zvukovye-maiachki---ugroza-anonimnosti__bmDHMhBF> (дата звернення 09.05.2020)