

ESCUELA TÉCNICA SUPERIOR DE INGENIEROS  
INDUSTRIALES Y DE TELECOMUNICACIÓN

UNIVERSIDAD DE CANTABRIA



***Trabajo Fin de Grado***

**Estudio y análisis de vulnerabilidades de  
la Deep Web mediante la implementación  
de un nodo Tor**

**(Study and vulnerability analysis of the  
Deep Web by implementing a Tor node)**

Para acceder al Título de

***Graduado en  
Ingeniería de Tecnologías de  
Telecomunicación***

Autor: Raúl González Gómez

Marzo – 2018



E.T.S. DE INGENIEROS INDUSTRIALES Y DE TELECOMUNICACIÓN  
**GRADUADO EN INGENIERÍA DE TECNOLOGÍAS DE TELECOMUNICACIÓN**

**CALIFICACIÓN DEL TRABAJO FIN DE GRADO**

**Realizado por:** Raúl González Gómez

**Director del TFG:** Roberto Sanz Gil

**Título:** “Estudio y análisis de vulnerabilidades de la Deep Web mediante la implementación de un nodo Tor”

**Title:** “Study and vulnerability analysis of the Deep Web by implementing a Tor node”

**Presentado a examen el día:** 7/03/2018

**Para acceder al Título de:** GRADUADO EN INGENIERÍA DE TECNOLOGÍAS DE  
TELECOMUNICACIÓN

**Composición del Tribunal:**

**Presidente (Apellidos, Nombre):** Roberto Sanz Gil

**Secretario (Apellidos, Nombre):** Alberto Eloy García Gutiérrez

**Vocal (Apellidos, Nombre):** Marta García Arranz

**Este Tribunal ha resuelto otorgar la calificación de:** .....

**Fdo.:** El Presidente

**Fdo.:** El Secretario

**Fdo.:** El Vocal

**Fdo.:** El Director del TFG

(sólo si es distinto del secretario)

Vº Bº del Subdirector

Trabajo Fin de Grado Nº (a asignar por  
Secretaría)



## **Agradecimientos**

Quiero dar las gracias a Roberto, mi tutor en el proyecto, por el apoyo y la ayuda continua que me ha otorgado durante todo el camino recorrido con este proyecto.

Por otra parte, agradecer a mis compañeros Martin, Garrido y Jorge por estar unidos día tras día en la facultad y en especial a Alberto por apoyarme y permitirme usar su red doméstica para la realización del proyecto.

Para acabar, mi familia y mi novia, los cuales no han perdido nunca la confianza en mí y me han ayudado en todo.

Raúl González Gómez, marzo 2018

## Resumen

El proyecto trata de analizar los diferentes ataques existentes que pueden sufrir un nodo de salida (Exit Relay) de la Tor Network y que pueden hacer que la vulnerabilidad de dicha red se vea comprometida. Se implementará un Exit Relay, y se realizarán diferentes tipos de ataques, como por ejemplo, el conocido como The Sniper Attack, que consiste en que un atacante consiga inutilizar un Exit Relay (o un grupo de ellos) de manera remota.

Durante los experimentos desarrollados por el Laboratorio de Investigación Naval de los Estados

Unidos, en Washington DC, se ha probado que el ataque puede reducir la memoria del nodo víctima con un consumo de ancho de banda de subida muy bajo. Es decir, podría inutilizar los nodos de salida más importantes de la red en tan solo unos minutos, reduciendo el ancho de banda total de la Tor Network considerablemente.

Otros tipos de ataques a estudiar en este trabajo podrían ser: modelos probabilísticos; ataques basados en análisis de tráfico y tiempo; AS con ataques a nivel global; y ataques basados en vulnerabilidades en los protocolos.

## Abstract

The project tries to analyze the different existing attacks that an Exit Relay of the Tor Network can suffer and that can make the network vulnerability compromised. An Exit Relay will be implemented, and different types of attacks will be carried out, for example, known as The Sniper Attack, which means that an attacker can disable an Exit Relay (or a group of them) remotely.

During the experiments carried out by the Naval Research Laboratory of the States

United States, in Washington DC, it has been proven that the attack can reduce the memory of the victim node with a consumption of very low rise bandwidth. That is, it could disable the most important network output nodes in just a few minutes, reducing the total bandwidth of the Tor Network considerably.

Other types of attacks to be studied in this work could be: probabilistic models; Attacks based on traffic and time analysis; AS with global attacks; And vulnerability-based attacks on protocols.

# Índice general

## 1. Contenido

Capítulo 1 .....	- 12 -
1. Introducción.....	- 12 -
1.1. Motivación .....	- 15 -
1.2. Objetivos .....	- 15 -
Capítulo 2 .....	- 16 -
2. Estado del arte .....	- 16 -
2.1. ¿Deep Web?, ¿Qué es? .....	- 17 -
2.2. Niveles de profundidad de la Deep Web .....	- 18 -
2.3. Crawling, ¿Qué son los crawlers y qué hacen? .....	- 20 -
2.4. ¿Acceso a Deep Web? .....	- 22 -
2.5. ¿Qué es Tor? .....	- 23 -
2.6. Funcionamiento de Tor .....	- 23 -
2.7. Tipos de Relays .....	- 25 -
2.8. Composición de la red Tor .....	- 27 -
2.9. Vulnerabilidades de Tor .....	- 30 -
2.10. Ataques cibernéticos, cultura criminal moderna.....	- 34 -
Capítulo 3 .....	- 40 -
3. El exit relay .....	- 40 -
3.1. Vidalia .....	- 42 -
3.2. Tor anonymizing relay monitor ARM .....	- 43 -
3.3. Atlas Tor .....	- 51 -
Capítulo 4 .....	- 56 -
4. Seguridad de un nodo Tor .....	- 56 -
4.1. Niveles inferiores de criptografía en Tor .....	- 58 -
Capítulo 5 .....	- 62 -
5. Implementación de un Exit Relay dentro de Tor .....	- 62 -
5.1. Instalación de un nodo Tor.....	- 62 -
5.2. Primer contacto con Tor-arm.....	- 68 -
5.3. Ataques y vulnerabilidades de nuestro relay .....	- 69 -
5.3.1. Tortazo .....	- 70 -
5.3.2. Tor's Hammer.....	- 73 -
5.4. Problemas técnicos en el desarrollo .....	- 74 -
Capítulo 6 .....	- 76 -
6. Conclusiones y líneas futuras .....	- 76 -
6.1. Conclusiones.....	- 76 -
6.2. Líneas futuras.....	- 76 -

## Índice de figuras

1.1	Estadísticas sobre Internet.....	12.-
1.2	Número de usuarios en Internet.....	12.-
1.3	Estadísticas consumo de internet siglo XXI.....	13.-
1.4	Simbología sobre internet.....	14.-
2.1	Niveles conocidos en la red.....	19.-
2.2	Representación del entramado de la red.....	20.-
2.3	Funcionamiento de recopilación de páginas web.....	21.-
2.4	Descripción simbólica de la estructura de la Deep Web.....	22.-
2.5	Logotipo "The Onion Router".....	23.-
2.6	Esquema de funcionamiento de Tor.....	24.-
2.7	Camino de transmisiones en Tor.....	24.-
2.8	Ataques más comunes de 2016.....	36.-
2.9	Representación mundial ciberataques.....	38.-
2.10	Visualización de "Wannacry".....	38.-
3.1	Composición de Tor.....	40.-
3.2	Interfaz principal de Vidalia.....	42.-
3.3	Interfaz Mapa Tor en Vidalia.....	42.-
3.4	Página 1 "Tor-ARM".....	44.-
3.5	Página 2 "Tor-ARM".....	45.-
3.6	Página 3 "Tor-ARM".....	46.-
3.7	Página 4 "Tor-ARM".....	46.-
3.8	Interfaz relay individual con "Atlas Tor".....	51.-
3.9	Interfaz sobre relay en "Tor Status".....	54.-
3.10	Segunda parte interfaz relay en "Tor status".....	55.-
4.1	Camino en Tor.....	56.-
4.2	Intercambio de claves Diffie Hellman .....	57.-
4.3	Cifrado Tor .....	57.-
4.4	Algoritmo AES.....	58.-
4.5	Algoritmo asimétrico RSA.....	59.-
5.1	Raspberry Pi 3 modelo B.....	62.-
5.2	Origen de Raspbian.....	63.-
5.3	Interfaces de red de Raspberry.....	64.-
5.4	Interfaz software "Putty".....	65.-

## Índice de tablas

2.1	Comparación Tor y VPN.....	-31-
-----	----------------------------	------



## Acrónimos

3DES	—	Triple Data Encryption Standard
AES	—	Advanced Encryption Standard
ARM	—	Anonymizing Relay Monitor
CPU	—	Central Processing Unit
DNS	—	Domain Name System
DH	—	Diffie-Hellman
DHE	—	Ephemeral Diffie-Hellman
EFF	—	Electronic Frontier Foundation
FTP	—	File Transfer Protocol
HDMI	—	High Definition Intermedia Interface
HTML	—	HyperText Markup Language
HTTP	—	Hypertext Transfer Protocol
HTTPS	—	Hypertext Transfer Protocol Secure
IP	—	Internet Protocol
IPv4	—	Internet Protocol version 4
IPv6	—	Internet Protocol version 6
ISP	—	Internet Service Provider
KML	—	Keyhole Markup Language
LIFO	—	Last In, First Out
MAC	—	Media Access Control
NAT	—	Network Address Translation
OP	—	Onion Proxy
OR	—	Onion router
OS	—	Operating system
OSI	—	Open System Interconnection
RAM	—	Random Access Memory
RSA	—	Rivest-Shamir-Adleman
SHA	—	Secure Hash Algorithm
SSH	—	Secure Shell
SSL	—	Secure Service Layer
TLS	—	Transport Level Security

TCP	—	Transmission Control Protocol
Tor	—	The Onion Router
UDP	—	User Data Protocol
URL	—	Uniform Resource Locator
USB	—	Universal Serial Bus
UTF-8	—	8-bit Unicode Transformation Format
VPN	—	Virtual Private Network
WiFi	—	Wireless Fidelity
WWW	—	World Wide Web

## Palabras clave

- Red Tor
- Deep Web
- Relay
- Internet
- Anonimato
- Ataques cibernéticos
- Ciberseguridad
- Raspberry

# Capítulo 1

## 1.Introducción

En la actualidad, la tecnología en todos sus ámbitos está expuesta a un continuo cambio, el cual puede llegar a ser abrumador hasta para el ser humano familiarizado con esta rama de la ciencia.

La sociedad avanza, hasta el punto en el que toda persona vive desde sus inicios rodeada de una gran variedad de dispositivos conectados a la red. El tiempo de uso de estos dispositivos aumenta a un ritmo frenético día a día y damos cada vez mayor importancia a su uso para las actividades cotidianas. [17] [18]

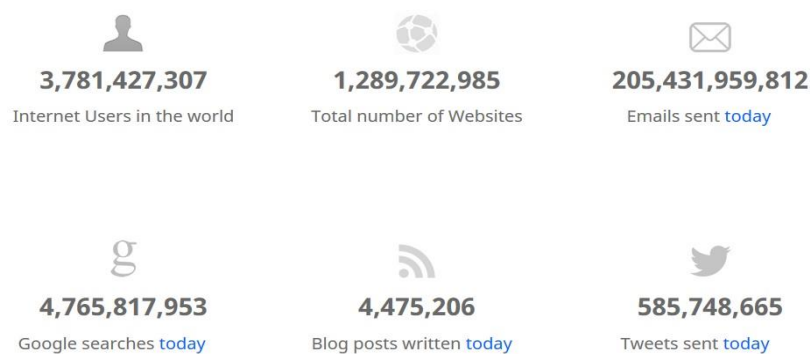


Figura 1.1: Estadísticas sobre Internet

En el año 2017, estudios realizados por Internet Live Stats afirman que el 40% de la población mundial (sobre 3.636 millones de usuarios) están conectados unos con otros mediante internet, cifra que aumenta constantemente. [19]

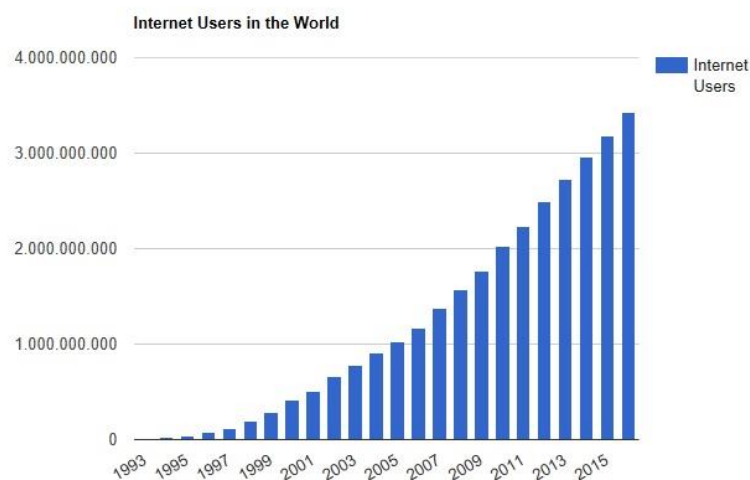


Figura 1.2: Número de usuarios en Internet

Year	Internet Users**	Penetration (% of Pop)	World Population	Non-Users (Internetless)	1Y User Change	1Y User Change	World Pop. Change
2016*	<b>3,424,971,237</b>	46.1 %	7,432,663,275	4,007,692,038	7.5 %	238,975,082	1.13 %
2015*	<b>3,185,996,155</b>	43.4 %	7,349,472,099	4,163,475,944	7.8 %	229,610,586	1.15 %
2014	<b>2,956,385,569</b>	40.7 %	7,265,785,946	4,309,400,377	8.4 %	227,957,462	1.17 %
2013	<b>2,728,428,107</b>	38 %	7,181,715,139	4,453,287,032	9.4 %	233,691,859	1.19 %
2012	<b>2,494,736,248</b>	35.1 %	7,097,500,453	4,602,764,205	11.8 %	262,778,889	1.2 %
2011	<b>2,231,957,359</b>	31.8 %	7,013,427,052	4,781,469,693	10.3 %	208,754,385	1.21 %
2010	<b>2,023,202,974</b>	29.2 %	6,929,725,043	4,906,522,069	14.5 %	256,799,160	1.22 %
2009	<b>1,766,403,814</b>	25.8 %	6,846,479,521	5,080,075,707	12.1 %	191,336,294	1.22 %
2008	<b>1,575,067,520</b>	23.3 %	6,763,732,879	5,188,665,359	14.7 %	201,840,532	1.23 %
2007	<b>1,373,226,988</b>	20.6 %	6,681,607,320	5,308,380,332	18.1 %	210,310,170	1.23 %
2006	<b>1,162,916,818</b>	17.6 %	6,600,220,247	5,437,303,429	12.9 %	132,815,529	1.24 %
2005	<b>1,030,101,289</b>	15.8 %	6,519,635,850	5,489,534,561	12.8 %	116,773,518	1.24 %
2004	<b>913,327,771</b>	14.2 %	6,439,842,408	5,526,514,637	16.9 %	131,891,788	1.24 %
2003	<b>781,435,983</b>	12.3 %	6,360,764,684	5,579,328,701	17.5 %	116,370,969	1.25 %
2002	<b>665,065,014</b>	10.6 %	6,282,301,767	5,617,236,753	32.4 %	162,772,769	1.26 %
2001	<b>502,292,245</b>	8.1 %	6,204,310,739	5,702,018,494	21.1 %	87,497,288	1.27 %
2000	<b>414,794,957</b>	6.8 %	6,126,622,121	5,711,827,164	47.3 %	133,257,305	1.28 %

Figura 1.3: Estadísticas consumo de internet siglo XXI

Podemos encontrar actividades muy variadas, las cuales precisan de una conexión con internet para poder ser realizadas. Hoy en día ya no existe un sector que no trabaje apoyándose en la red.

Dependiendo de la labor que se quiera realizar dentro de la red cada usuario emplea una de las características presentes en internet como, por ejemplo: universalidad, interdisciplinariedad laboral, facilidad de uso y gran accesibilidad, libertad de expresión, anonimato, etc.

La sociedad de la información en la que nos encontramos hoy en día se apoya en todas las características anteriormente citadas, pero hay una que está cobrando gran relevancia en los últimos tiempos, es la opción de ser anónimo en la red pudiendo ocultar nuestra identidad frente a terceros.

Para lograr el ansiado anonimato en la red podemos decantarnos por una red VPN o en su defecto acceder mediante Tor.

En el caso que nos ocupa profundizaremos en el sistema Tor, haciendo hincapié en su funcionamiento interno, características principales y resultados alcanzables.

Internet siempre ha tenido una imagen generalizada de iceberg, el cual se divide en una zona situada por encima de la superficie y otra profunda de un volumen mucho mayor.

La primera parte es accesible para todos los usuarios corrientes de la red y se correspondería con el internet “visible” englobando únicamente un 4% del volumen total de la red, lo que conocemos como “Surface Web”.

La segunda en cambio es la parte profunda que engloba un porcentaje del 96% sobre el tamaño total de internet y es la zona que necesita de unas condiciones especiales para su acceso, engloba tanto la Deep Web como la Dark Web. [16]

Además del volumen tan bien diferenciado entre las dos áreas que encontramos en internet, el internet tradicional y el profundo poseen multitud de diferencias que explicaremos más adelante.

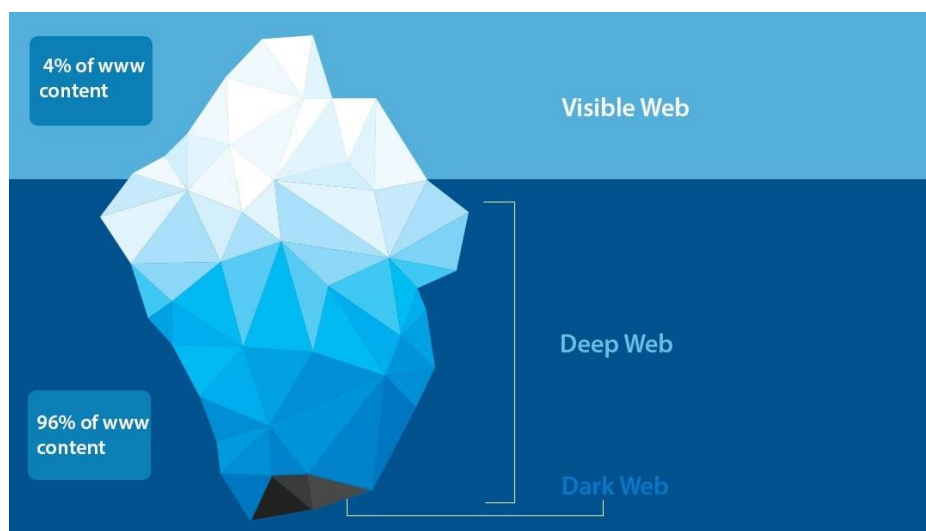


Figura 1.4: Simbología sobre internet

Desde su aparición, la Deep Web se rodea de un aura de misterio por todas las historias relacionadas con las actividades que se realizan en su interior como el material o contenido visible que circulan por la sociedad en la que vivimos.

El morbo por lo ilegal o lo moralmente incorrecto está muy presente en la comunidad del siglo XXI, lo que es cierto es que dentro del internet profundo hay dos clases de usuarios, los que buscan mejorar el mundo apoyándose en las ventajas que presenta esta red y los que solo quieren lucrarse o realizar actos ilegales y en ocasiones hasta enfermizos.

En este proyecto vamos a aclarar el funcionamiento de esta tecnología y desmentir los mitos que aparecen sobre ella, adentrémonos en las profundidades de la red.

## **1.1. Motivación**

La motivación de mayor peso para enfocarnos en la realización de este proyecto es la de aclarar el funcionamiento de la Deep Web desde dentro a la vez que podemos ir desmintiendo o afirmando los mitos que la rodean.

Para ello realizaremos la presentación de un exit relay a la red Tor y su asentamiento en la misma con todo lo que ello implica.

A continuación, y con todo lo anterior resuelto, comprobaremos el nivel de seguridad y las formas de ataque que existen sobre los repetidores de esta red.

## **1.2. Objetivos**

Como objetivo tenemos el aprendizaje teórico y práctico del funcionamiento y mantenimiento tanto de la red Tor como una única red, así como de todos los niveles que encontramos por debajo de ésta. Entre ellos encontramos todos los distintos dispositivos y miembros que albergan su propia información de configuración dentro de Tor y con ello sus formas únicas de funcionamiento junto al resto de miembros.

A nivel teórico, buscamos conocer la forma de búsqueda dentro de la Deep Web ya que las páginas dentro de este sitio no están indexadas al contrario que en el internet normal.

Los buscadores de la Surface Web no pueden encontrar las direcciones de los espacios web situados en la Deep Web por lo que es preciso usar otras herramientas para tal propósito.

En el enfoque práctico el objetivo será el entendimiento de la transmisión de la información por la red mediante los relays existentes, el encaminamiento de los datos, la seguridad que precisan los usuarios de esta red y, por el contrario, la flaqueza del sistema y los ataques realizables sobre éste.

# Capítulo 2

## 2. Estado del arte

Para poder entender el grado de importancia que tiene internet en nuestras vidas debemos deshojar lo que llamamos “La red de redes” en sus diferentes capas explicando en profundidad cada una de ellas.

Internet es la red por excelencia, se compone de páginas estáticas o fijas y posee un alcance mundial que conecta al resto de redes existentes. Esto es relevante para la experiencia de cada usuario en la red ya que le concede la posibilidad de contactar con personas situadas en cualquier punto del planeta y compartir con ellos recursos de cualquier índole.

Sabemos que el tamaño de internet es desbordante, además de esta realidad tenemos que dar a conocer el resto de sus asombrosas características. **[20]**

### 1. Universal

Como hemos dicho anteriormente, esta tecnología abarca el mundo entero y gracias a ello podemos acceder a contenido y servicios generados en otros países al instante.

### 2. Eliminó las barreras de tiempo y espacio

Podemos acceder a eventos en tiempo real, aunque se encuentren a miles de kilómetros.

### 3. Económicamente accesible

Obviamente dependiendo de las políticas de cada país esto puede variar razonablemente, pero en líneas generales el coste del uso de internet es mínimo.

Hoy en día, tanto buscar información como comprar productos pasando por el resto de las actividades accesibles digitalmente están al alcance de la mano de la gran mayoría de las personas.

### 4. Facilita la interdisciplinariedad laboral

Gracias a la fácil conectividad que nos brinda internet somos capaces de colaborar en proyectos vía web y así coordinar grupos de trabajo.

### 5. Trajo profundos cambios sociales

La evolución de internet y sus consecuencias como la creación de las redes sociales y otro tipo de redes nos dan la oportunidad de interactuar con personas que nunca hemos tenido frente a frente.

### 6. Facilidad de uso

Cualquier persona puede acceder de manera estable y continua a internet sin demasiados problemas. Esta sencillez es la que ha llevado a internet a ser uno de los mayores éxitos en la historia de la humanidad.



## **7. Posibilita el anonimato**

Como luego explicaremos en detalle, internet posee sitios que obligan a una identificación por parte del usuario. Por otra parte, podemos encontrar lugares donde predomina el anonimato de los usuarios. Esto puede ser a la vez un punto positivo o una gran desventaja.

## **8. Masificación de contenido**

La posibilidad de llegar a muchas personas en poco tiempo es algo muy preciado para compañías y campañas de solidaridad.

## **9. Otorga libertad de expresión**

Lograr la regulación total y efectiva de internet es prácticamente imposible, esto provoca que los usuarios puedan subir a la red cualquier contenido en cualquier momento. Este sin duda es uno de los puntos más peligrosos de esta tecnología.

## **10. Dependencia o adicción**

La invención de internet es una de las cosas que ha marcado el camino a seguir en el futuro por la humanidad. El potencial de internet es asombroso a la vista de todos, demasiado para algunas personas que se ven sobrepasadas por el gran abanico de posibilidades que ofrece la red. Esto puede llevar a adicciones u obsesiones relacionadas con problemas físicos y psíquicos.

Internet posee un número inmenso de datos representativos como pueden ser:

- El 40% de la población mundial está actualmente conectada a la red, este porcentaje aumenta a diario.
- La primera página web fue puesta en línea en el año 1991, en 2017 la red alcanza la cifra de 1,3 billones de páginas, aunque el 75% de ellas están inactivas.
- Cada segundo se realizan búsquedas en la red con cualquier temática y se envían emails que recorren todo el mundo.

Estas impresionantes cifras se basan en la red que el usuario de a pie conoce, aunque como ya sabemos esto aproximadamente se basa en el 4% del total por lo que la red engloba mucho más en sus oscuros rincones, la llamada “Deep Web”. [15]

## **2.1. ¿Deep Web?, ¿Qué es?**

El 96% restante posee infinidad de datos, lo que conocemos como la Internet profunda o Internet invisible. [13] [14]

La existencia de la Internet profunda se basa esencialmente en la imposibilidad de indexar o poder localizar un gran porcentaje de la información existente mediante los motores de búsqueda (Google, Yahoo, Bing, etc.).

Dentro de este Internet invisible podemos encontrar diferentes niveles. A mayor profundidad la seguridad del usuario se diluye a la vez que aumenta el misticismo y la ilegalidad del contenido.

## **2.2. Niveles de profundidad de la Deep Web**

El número de niveles existentes en la red no está del todo claro ya que a partir de cierta profundidad los mitos crecen y la realidad sobre estos sitios se difumina, aun así, vamos a realizar la clasificación más detallada posible. [24] [25]

Podemos afirmar que hay cinco niveles reales y bien asentados dentro de la red, cruzando este límite ya nada está claro ni es verificado por la mayoría de la humanidad.

### **Nivel 1**

La conocida por todos "Internet visible". El nivel más superficial que encontramos en la red y la zona más segura para el usuario. Como ya sabemos aquí se sitúan las páginas de acceso común, las redes sociales, periódicos digitales e infinidad de páginas con temáticas variadas.

### **Nivel 2**

Bautizado como "Bergie Web". Aquí encontramos el resto de la internet conocida e indexada pero un poco más profunda, es decir, servidores FTP, pornografía, buscadores independientes como por ejemplos los reconocidos Ares y Emule, etc.

Las cosas empiezan a cambiar a partir de aquí.

### **Nivel 3**

Es el nivel donde ya encontramos la famosa "Deep Web".  
Requerimos el uso de un proxy para poder sumergirnos de una manera anónima.

Las páginas albergadas en este nivel no son alcanzables por los buscadores comunes ya que emplean el pseudo-dominio .onion y no alguno común como .com o .org. Además, y como dato significativo, estas páginas no se relacionan entre ellas como sí lo hacen las que encontramos en el internet tradicional.

Encontraremos aquí sitios abandonados a lo largo del tiempo, redes Torrent, foros con temáticas interesantes y otros con dudosa integridad (venta de drogas y armas, pornografía...), cibercriminales, etc.

### **Nivel 4**

Este piso se llama "Charter Web" y precisa del uso de Tor para poder navegar de forma segura y anónima. El peligro aumenta en este nivel de manera alarmante y los ataques pueden llegar de cualquier sitio en forma de robo y malversación de datos.

La parte más accesible de este nivel es un mercado negro de todo lo relacionado con la ilegalidad en el que encontramos pedofilia, libros y vídeos prohibidos, asesinos a sueldo, tráfico de personas, etc.

Dentro del nivel 4 hay una parte aún más profunda a la que no se puede llegar por medios convencionales, se requiere una modificación de hardware llamada "closed shell system". Aquí se pueden encontrar las cosas más sórdidas de la red.

Sobrepasar este punto requiere tener algo prácticamente imposible de lograr, por no decir completamente inviable. Esto es tener un hardware con capacidad de computo de algoritmos al nivel de mecánica cuántica, lo que solo es posible para el gobierno.

## Nivel 5

Es el nivel llamado “Marianas Web” por el punto más profundo del planeta “Fosa de las Marianas” con su profundidad de 10.934 metros. Hay poca información relacionada sobre este nivel. Documentación clasificada, pruebas y experimentos del gobierno y todo lo que tenga que ver con masonería.

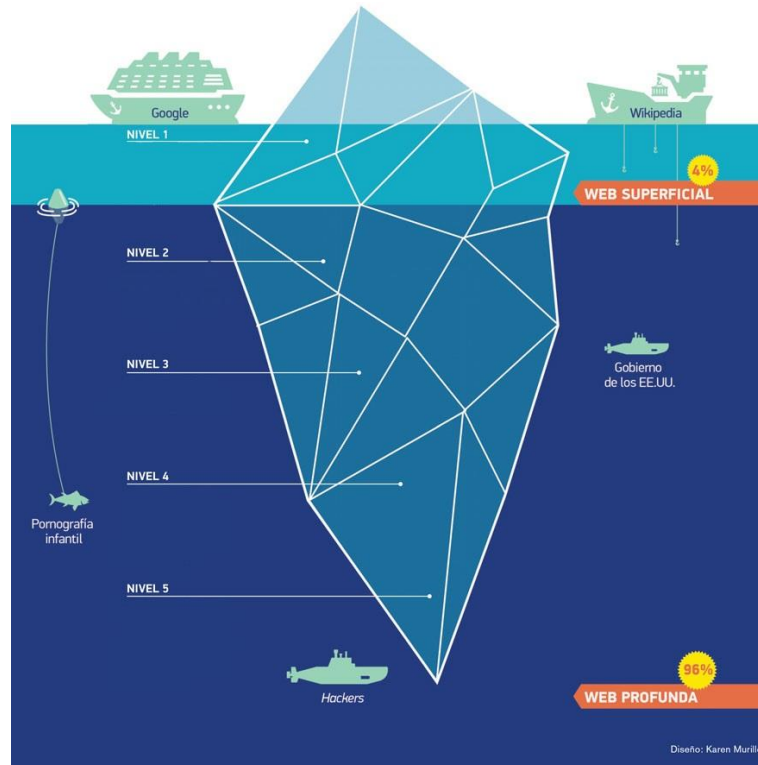


Figura 2.1: Niveles conocidos en la red

Lo que queda de la red es un misterio y solo podemos aventurar conjeturas sobre el resto de los niveles ya que no es posible encontrar pruebas de alguien que haya logrado entrar.

## Nivel 6

Llamado “The fog”. Aquí los hackers que consiguen entrar utilizan la computación cuántica para poder superar la encriptación y así poder navegar por la inmensidad de datos en este nivel.

## Nivel 7

Se puede nombrar como “Virus Soup”. Es una zona de guerra donde se encuentran los hackers de más alto nivel (todos ellos desconocidos), los cuales están en constante conflicto para llegar al siguiente nivel.

## Nivel 8

Se especula que este nivel es el control primario de internet. Quien sea capaz de acceder aquí podrá controlar totalmente la red a su antojo.

## 2.3. Crawling, ¿Qué son los crawlers y qué hacen?

Como hemos dicho anteriormente, una de las diferencias sustanciales entre la web superficial y la profunda es la manera de alcanzar una página web. Esto depende totalmente de los crawlers existentes, si son capaces o no de alcanzar estas páginas. [23]

La Deep Web se puede imaginar como una zona oscura y desamparada dentro de la red a la que no consiguen acceder estos crawlers y por conclusión, toda página incluida en la Deep Web no va a estar nunca indexada. Por suerte, esto no significa que sea imposible acceder a ellas ya que hay métodos para alcanzar lo que buscamos como son:

- Comprar links a usuarios de la Deep Web.
- Usar buscadores diseñados específicamente para trabajar dentro de la Deep Web como son: DuckDuckGo, directorio propio de Tor, The Hidden Wiki, Not Evil, etc.

Sabido lo anterior podemos decir que un crawler, también conocido como araña de la web, es un software o webbot que recorre los enlaces automáticamente de las páginas webs.

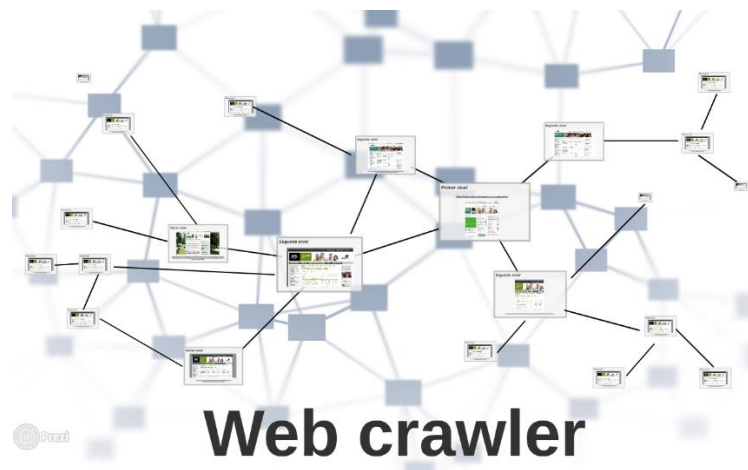


Figura 2.2: Representación del entramado de la red

Un crawler dispone de un conjunto inicial de URLs, conocidas como semillas, y va descargando las páginas web asociadas a las semillas, añadiendo éstas a la lista de URLs y buscando dentro de éstas URLs otras direcciones. [26]

Cada nueva URL encontrada se añade a la lista de URLs que la araña web debe visitar conocida como la frontera de rastreo. Es decir, recolecta URL's para posteriormente descifrar su código y almacenar datos sobre ellas en una base de datos.

Los archivos se almacenan por lo general de tal manera que se puedan ver, leer y navegar como lo fueron en la web en directo, pero se conservan como "instantáneas". Así, el motor de búsqueda creará un índice de las páginas descargadas para proporcionar búsquedas más rápidas.

Cuando un crawler visita un sitio web opta por una de estas dos alternativas:

- Buscar el archivo robots.txt que contiene direcciones para los crawlers y links del sitio web y busca también la meta etiqueta robots para ver las reglas que se han estipulado.  
Con este archivo, los servidores HTTP de la Deep Web controlan el acceso a determinados directorios y archivos del servidor.

- Elaborar un índice de las páginas web que hay en su sitio explorando el contenido del texto visible, de varias etiquetas HTML y los hipervínculos en listados en la página.

Como dato curioso Google posee el crawler más famoso del mundo llamado Googlebot. La cadencia del Googlebot para acceder a un sitio web depende del Ranking de éste, cuanto mayor valor tenga, más veces accederá el robot a sus páginas. En este sentido, los medios de comunicación son visitados cada día por Googlebot, mientras que hay sitios a los que no accede en semanas.

Para saber si Googlebot ha visitado a nuestra página, sólo tenemos que revisar la caché y observar los logs de nuestro servidor para ver si el rastreador ha entrado. [27]

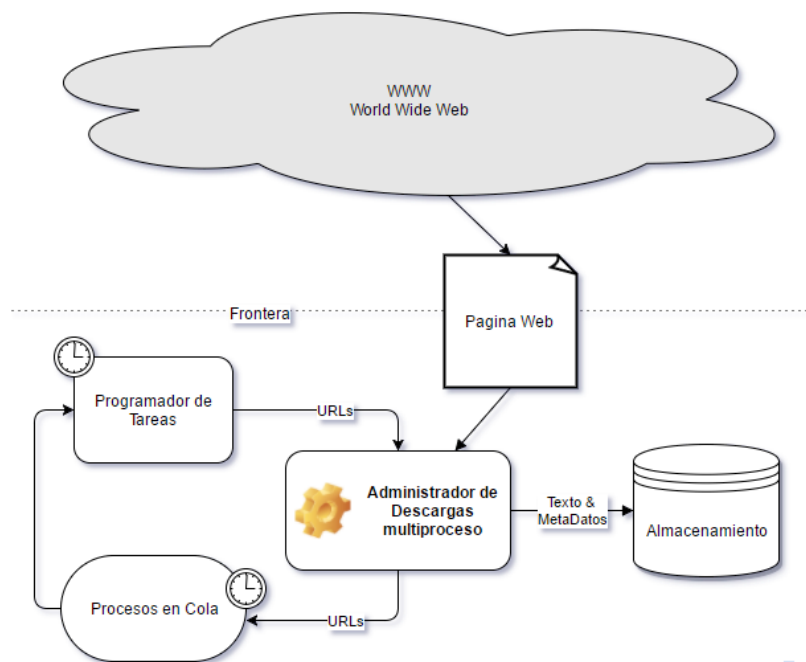


Figura 2.3: Funcionamiento de recopilación de páginas web

Decimos que las páginas no indexadas se encuentran únicamente en el internet profundo, esto es relativamente cierto hasta un punto. En el internet superficial hay métodos con los que se puede evitar la indexación de un sitio web:

- Sitios web los cuales requieren de una autenticación normalmente con un usuario y una contraseña para su acceso.
- Sitios dinámicos fácilmente editables.
- Páginas web no indexadas deliberadamente a petición de su gestor o creador.

## 2.4. ¿Acceso a Deep Web?

Los distintos niveles de la Deep Web están íntimamente ligados con los modos de acceso a la red, es decir, cada paso que avanzamos en la profundidad del Internet profundo nos obliga a emplear herramientas de acceso más complejas para tener la certeza de avanzar con seguridad por estos sitios web.

La Deep Web puede ser vista como una cueva con múltiples niveles.

Los inicios son sencillos, los cuales no requieren de unos conocimientos avanzados ni de un material especializado para su exploración. El problema viene cuando se quiere continuar y encontramos numerosas dificultades como fosos, caminos estrechos y terrenos escarpados los cuales nos obligan a tener a nuestra disposición las herramientas adecuadas para su avance, ya que sin ellas podemos encontrarnos en peligro si decidimos continuar nuestra aventura.

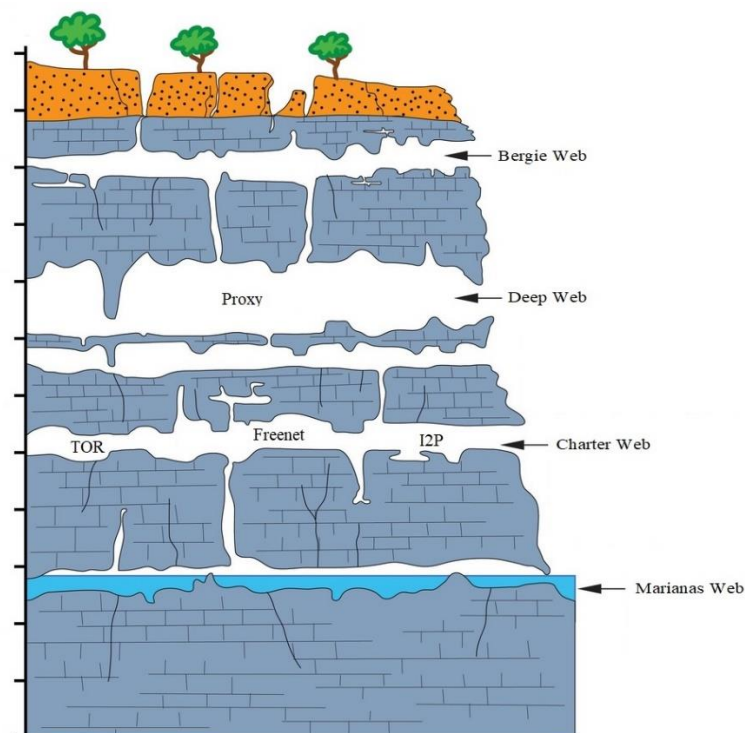


Figura 2.4: Descripción simbólica de la estructura de la Deep Web

El interior de la Deep Web se asemeja mucho a este tipo de estructura natural.

Las primeras capas no albergan demasiadas complicaciones y con emplear un proxy podemos navegar relativamente seguros.

Cuando en cambio queremos avanzar en nuestra investigación y continuar bajando de nivel las cosas se ponen serias. Estas capas albergan a hackers cada vez de mayor nivel y usuarios con dudosa moralidad que pueden provocarnos una muy negativa experiencia en esta red por lo que debemos acceder con sumo cuidado.

Para conseguir esto, "ANONIMATO" es nuestro término ya que sin ello estaremos expuestos a que hagan con nosotros lo que quieran.

Esta característica solo la podremos conseguir usando una red Tor, Freenet o I2P que son capaces de otorgarnos un anonimato frente al resto de usuarios.

En este proyecto vamos a adentrarnos en profundidad en Tor.

## 2.5. ¿Qué es Tor?

The Onion Router (Tor) es un software anti espionaje con enrutamiento cifrado implementado por la marina de los Estados Unidos creado aproximadamente a finales del año 2002. EFF (Electronic Frontier Foundation), una organización creada para luchar por nuestros derechos digitales continuó el legado de Tor patrocinando dicho programa. [12]

Actualmente The Onion Router está gestionado por “Tor Project”, organización sin ánimo de lucro que trabaja para que cada usuario tenga libertad de acceso y expresión en Internet manteniendo su privacidad y anonimato. [29]

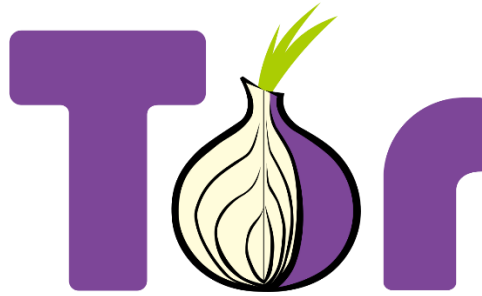


Figura 2.5: Logotipo “The Onion Router”

Tor traducido a nuestro idioma quiere decir “El encaminamiento cebolla” y se refiere a una red compuesta por servidores de usuarios que donan su ancho de banda y capacidad de procesamiento por el bien global. Esta red nos permite navegar por las profundidades de la Deep Web sin revelar nuestra identidad a ningún usuario de forma privada y segura. Más concretamente, Tor oculta el origen y destino del tráfico de Internet, haciendo que otros no puedan averiguar tan fácilmente quién eres y qué estás viendo en línea.

## 2.6. Funcionamiento de Tor

Cuando un usuario desea emplear Tor para realizar una búsqueda anónima u otro servicio, esta red crea un túnel virtual donde se emplea el protocolo TLS (Transport Layer Security), por encima de TCP/IP en la capa de transporte del modelo OSI.

La red Tor posee relays o nodos voluntarios situados por todo el mundo.

El siguiente paso en la comunicación origen-destino es el cifrado de los datos por capas. Los datos son transmitidos por una ruta que cruza tres relays entre el usuario de la red y el servidor, esta configuración es la óptima para el anonimato en Tor y no se debe variar como más adelante explicaremos.

Como método para garantizar la seguridad, cada relay en el camino solo sabe de quien recibe los datos y a quien debe mandárselos. Añadido a esto, cada salto de la información en el camino provoca una nueva negociación de las llaves a emplear en los cifrados, esto asegura que Tor sea un sistema robusto en la comunicación.

Lógicamente, el enrutamiento de cebolla tiene un origen. Su nombre lo debe al sistema en el que cada relay al recibir la información, la encripta con sus datos privados y lo transmite al siguiente provocando que los datos logren en el proceso varias capas de encriptación. La información transmitida solo va a ser descryptada con las llaves del cliente respecto a cada relay que entra en juego.

En toda transmisión realizada dentro de la red Tor podemos ver un esquema idéntico compuesto por un usuario de origen, un servidor de destino y siempre tres relays trabajando como enlace entre los dos anteriores.

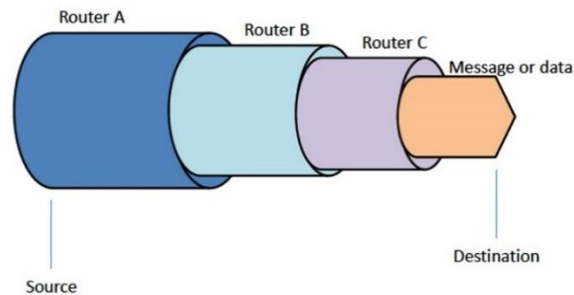


Figura 2.6: Esquema de funcionamiento de Tor

La pregunta que se puede hacer cualquier persona en este momento es, ¿por qué deben ser siempre tres relays los que aparezcan en la ruta?, vamos a despejar la incógnita a continuación.

Si la transmisión origen-destino cuenta solo con dos relays el usuario percibirá una velocidad de navegación mayor y un mejor rendimiento del sistema, pero nada de esto compensa ya que el usuario estará desprotegido y verá como su anonimato estará expuesto. Esta grave falta de seguridad se debe a que si un atacante acecha un relay de salida automáticamente conoce el relay que actúa como guarda de entrada y por consiguiente al usuario que se conecta a la red Tor y sus datos.

Entonces, sería lógico pensar que si añadimos más relays a la ecuación y tenemos una ruta con cuatro o cinco nodos la seguridad del cliente de Tor sería infranqueable, y nada más lejos de la realidad.

Con esta configuración la seguridad se verá muy perjudicada igual o de mayor manera que al emplear únicamente un par de relays. Podemos añadir que el cómputo general del sistema es mayor y como resultado la velocidad de navegación para el usuario caería drásticamente al igual que el rendimiento general.

Vemos entonces que esta elección no otorga nada positivo a la navegación del usuario con Tor dando como resultado la elección de tres relays como la opción ganadora y más funcional.

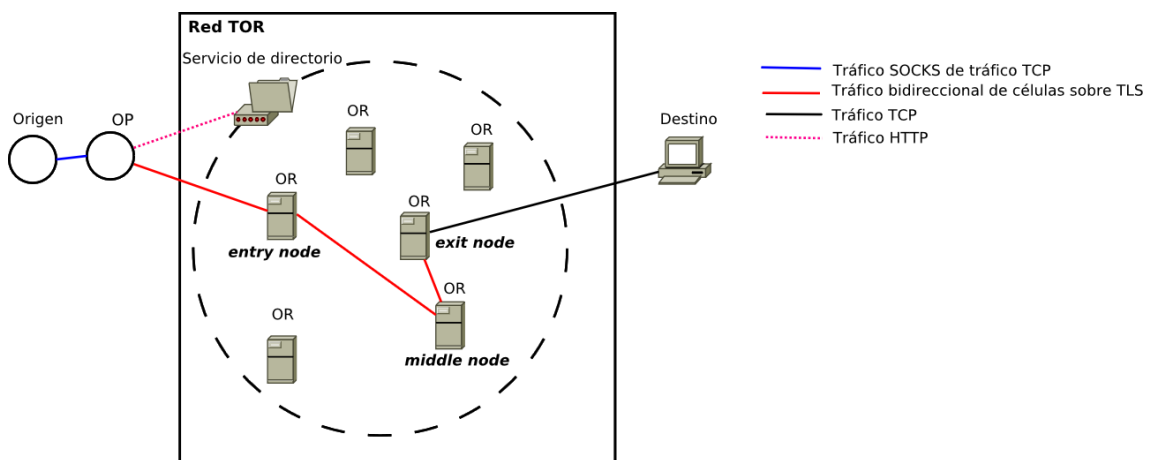


Figura 2.7: Camino de transmisiones en Tor



Por último, gracias a la imagen que encontramos anteriormente vamos a poder explicar una comunicación y transmisión completa mediante el enrutamiento cebolla. Los pasos a seguir son:

1. El OP (Onion Proxy) del cliente solicita al servicio de directorio una lista con los nodos Tor.
2. El Proxy del cliente “elige” de manera aleatoria un camino para alcanzar al servidor destino.
3. Se envía la información al nodo de entrada y este lo encripta con su clave.
4. El entry node transmite el paquete encriptado al nodo intermedio para que lo encripte de nuevo con su clave.
5. El exit relay recibe los datos y vuelve a realizar una encriptación con su clave (añade otra capa más a la cebolla).
6. Se realiza la desencriptación de la información en orden LIFO para que el servidor de destino pueda recibir los datos de manera legible.

## **2.7. Tipos de Relays**

Quien desee apoyar la causa por la libertad de expresión en países altamente censurados y ayudar a los usuarios a alcanzar el anonimato en la red puede añadir a la red un nuevo nodo para mejorar el servicio.

Para ello lo primero será elegir el tipo de nodo que queremos que acceda a la red y esto pasa por conocer todos los tipos que existen de relays y la labor que realiza cada uno.

Podemos clasificar los relays de Tor en cuatro categorías: Entry Guards, Middle Relays, Exit Relays y por último bridges.

### **Entry Guard**

En toda transmisión Tor, este relay es el encargado de recibir la información del cliente en el primer salto y asegurar la privacidad de la comunicación. Un Entry Guard es simplemente un Middle Relay que ha logrado la autorización para trabajar como guard gracias a poseer un ancho de banda necesario, un cierto uptime garantizado y un tiempo mínimo de buen trabajo como Middle en la red. Al cumplirse todos estos requisitos, los cuales no son sencillos, se obtiene la flag de Entry Guard.

Los Entry Guard son imprescindibles en la seguridad de la red ya que son relativamente complejos de conseguir como hemos dicho anteriormente. Aseguran además una variedad de nodos dentro de la red, es decir, en una comunicación Tor cada nodo es diferente y tiene una labor distinta, esto hace que a un cibercriminal le sea más difícil poseer dos relays en una comunicación y con ello poder extraer información del cliente.

Cada cliente tiene asignado unos Entry Guards con los que trabaja siempre como si fuese una suscripción temporal la cual caduca. A la hora de renovar la lista de Guards para cada usuario, estos cambian hasta que vuelvan a caducar y con ello se reasignan.

## **Middle Relay**

Es el tipo de relay más sencillo de obtener ya que no tiene unos requisitos específicos ni una configuración compleja al rechazar cualquier política de salida.

Se puede decir que este relay pasa desapercibido ya que recibe información encriptada, añade su cifrado, y lo transmite al relay de salida. Es un relay que no maneja información en claro por lo que nadie se fija en él para realizarle ataques y así sacar datos de interés.

## **Exit Relay**

Es el tipo de relay más peligroso y a la vez al que más valor hay que dar desde el punto de vista del propietario del nodo y de los usuarios. Al realizar el último salto hacia el servidor web, todos los ojos de los atacantes están puestos sobre ellos para poder extraer información importante.

Como vemos, poseer un exit relay es peligroso y complejo. Pero hay más complicaciones con este repetidor ya que es muy exigente en su funcionamiento. Un nodo de salida funcionando en la red Tor precisa de un ancho de banda simétrico de gran capacidad para poder ofrecer un funcionamiento estable y con buenos resultados.

Para poseer un nodo de salida debemos configurar un middle relay y cambiarle las políticas de salida, éstas se compondrán de las direcciones IP y los puertos a los que daremos permisos de acceso o a los que se los denegaremos. Esto dependerá del objetivo y la seguridad que tenga el poseedor del relay.

## **Bridge**

Es el nodo que dista más de los anteriores tipos de nodos ya que éste es el único que no aparece en el directorio principal de Tor al resto de la red cuando se sitúa en línea.

Al no haber una lista pública de todos los bridges, aunque el ISP del cliente filtre conexiones a los relays de la red Tor, este no podrá bloquear todos los bridges en línea. Como resultado, si el acceso a la red Tor está siendo bloqueado, se pueden emplear bridges, dando así un papel crucial a los bridges en la lucha contra la censura que someten algunos países a sus ciudadanos. [5]

Cualquier usuario se conectará siempre a un entry guard, a un middle relay y a un exit relay en su camino hasta alcanzar el destino establecido. Esto viene preestablecido hasta que cambiemos la configuración que tenemos a nuestro alcance en el buscador de Tor.

Para emplear bridges en nuestra conexión debemos incorporarlos a nuestro Tor Browser Bundle. Para ello debemos seguir unos sencillos pasos:

1. Obtenemos los bridges existentes en la página: <https://bridges.torproject.org/bridges>
2. Seleccionamos el botón “torbutton” (representado por una cebolla) del navegador Tor y abrimos las preferencias de red.
3. Marcamos la opción “Mi proveedor de servicios de internet (ISP) bloquea las conexiones Tor” y copiamos la dirección de uno de los bridges disponibles.

Últimamente algunos ISPs han logrado encontrar maneras de bloquear el tráfico usando una técnica llamada DPI (“Inspección profunda de paquetes”). Para seguir sorteando la censura se emplean herramientas como “obfsproxy” que transforman o camuflan el tráfico entre el cliente y el bridge simulando una conexión http más “sencilla” o hasta logrando una simulación de una sesión por Skype.

## **2.8. Composición de la red Tor**

La mala fama lograda por la red Tor sobre su contenido lleno de documentación ilegal y contenido multimedia sensible está verificada, esto no quiere decir que todo lo que podamos encontrar dentro de esta red sea ilegal.

Dentro de Tor cualquier usuario puede encontrar el contenido más miserable de internet, así por el contrario también información útil y precisa que se creía desaparecida de la red.

Tor ofrece realizar multitud de actividades en línea con la posibilidad de hacerlo de forma anónima al resto de la red, esto provoca la atracción por parte de más de un millón de usuarios. Cada una de estas personas llega a Tor con un objetivo muy distinto, pudiendo conectarse un periodista buscando anonimato en sus noticias, un usuario en busca de un remedio para alguna dolencia de la que se pueda avergonzar o por el contrario alguien en busca de actividades ilícitas.

La variedad en Tor no se encuentra únicamente en el contenido existente, la enorme cantidad de usuarios que acceden a Tor diariamente nos da una pista de la cantidad de actividad que se realiza y con ello la gran diversidad de usuarios entre los que encontramos: [49]

### **Usuarios normales**

Personas que buscan proteger su privacidad de los proveedores de servicio los cuales pueden llegar a ofrecer sus registros de navegación y de ladrones de identidades. Los ISP dicen que anonimizan los datos de usuario lo cual no es del todo cierto.

Personas que quieren protegerse frente a empresas u organizaciones que infringen las normas sobre las documentaciones privadas de cada usuario.

Padres en busca de una protección extra para sus hijos en línea ya que cada vez es más sencillo realizar una identificación personal en internet y descubrir la ubicación exacta de un usuario.

Usuarios que por necesidad o curiosidad quieren realizar una búsqueda de material sensible. Dependiendo del país en el que nos encontremos podemos estar sometidos a censura de información.

### **Usuarios con perfil alto**

Personas con un puesto de trabajo de gran nivel y reconocimiento es seguido por un gran número de personas las cuales juzgan sus creencias políticas o religiosas. Ya que estas personas no quieren que esto afecte a su carrera profesional y a la vez no desean permanecer en silencio y poder así expresar su opinión emplean la red Tor.

### **Periodistas**

Multitud de reporteros sin fronteras han sido encarcelados por todo el mundo. Se recomienda a blogueros, periodistas y a sus fuentes el empleo de Tor para proteger su seguridad y privacidad.

Los periodistas requieren esta protección ya que su trabajo les dirige en muchas ocasiones a temas controvertidos sobre democracia, economía y religión.

Tor participa en SecureDrop, plataforma de software de código abierto destinada a la comunicación segura entre periodistas y fuentes. Muchas organizaciones de noticias usan este sistema de envío, incluyendo Associated Press, The Washington Post, The New York Times, The CBC, etc.

## **Grupos del orden**

Grupos telemáticos de la policía pueden vigilar sitios web cuestionables sin dejar pistas de su investigación o realizar operaciones encubiertas en la red.

Por otra parte, la policía puede trabajar fácilmente con sus fuentes de información sin que éstas tengan que temer por su seguridad ya que no pueden ser identificadas.

## **Grupos activistas y denunciantes**

Los activistas de derechos humanos usan Tor para denunciar anónimamente los abusos de las zonas en guerra.

Internacionalmente, los trabajadores de los derechos laborales emplean Tor y otras formas de anonimato en línea para organizar a los trabajadores de acuerdo con la Declaración Universal de los Derechos Humanos.

Global Voices recomienda Tor, especialmente para blogs anónimos, a través de su sitio web.

En el este de Asia, algunos organizadores laborales usan el anonimato para revelar información sobre talleres clandestinos que producen bienes para los países occidentales y para organizar la mano de obra local.

## **Empresas**

Se emplea Tor como centro de intercambio de información sobre violaciones de seguridad entre empresas.

Es un repositorio que requiere a los miembros que informen sobre las infracciones a un grupo central, que correlaciona los ataques para detectar patrones coordinados y enviar alertas. De esta manera y gracias al anonimato que otorga Tor las empresas pueden mejorar sus sistemas de seguridad sin que los posibles atacantes consigan información alguna.

Por otro lado, el entorno empresarial requiere mantener una confidencialidad de las estrategias de negocio donde vemos inversiones o próximos pasos evolutivos dentro de cada empresa.

## **Ejército**

Los soldados de campo desplegados fuera de su país usan Tor para enmascarar los sitios que visitan, protegiendo los intereses y operaciones militares, así como también protegiéndose de daños físicos.

Cuando Internet fue diseñado por DARPA, su objetivo principal era poder facilitar comunicaciones distribuidas y sólidas en caso de huelgas locales. Sin embargo, algunas funciones deben estar centralizadas, como son los sitios de comando y control. La naturaleza de los protocolos de Internet es revelar la ubicación geográfica de cualquier servidor. Tor permite que el comando y el control militar estén físicamente seguros contra el descubrimiento y el derribo.

A la hora de realizar una reunión de inteligencia el ejército precisa que no se guarden las direcciones ni la información trascendental en el servidor web situado en la zona insurgente.

## **Profesionales del sector de las telecomunicaciones**

Un firewall puede tener políticas que solo permitan ciertas direcciones IP o rangos de direcciones. Tor se puede utilizar para verificar esas configuraciones mediante el uso de una IP fuera del bloque de IPs asignado de la empresa.

Se puede emplear Tor para realizar actividades profesionales delicadas que en un principio están apartadas por las políticas estrictas y los sistemas de seguridad de cada empresa.

Un ingeniero de red puede usar Tor para volver a conectarse de forma remota a los servicios implementados de un proyecto, sin la necesidad de una máquina externa y una cuenta de usuario, como parte de las pruebas operacionales.

Si en algún momento nuestro ISP tiene problemas de enrutamiento, Tor puede poner a nuestra disposición recursos de Internet.

## Hackers

El principal objetivo de un hacker en la red Tor es la de conseguir alcanzar nuevas metas, evolucionar recopilando información útil que se encuentra esperando en las profundidades.

Además, el white hat hacker puede emplear el anonimato de la red Tor para poner al descubierto a usuarios que realizan actividades ilegales o éticamente dudosas.

## Ciberdelincuentes

Los delincuentes emplean la red Tor por la protección y seguridad que el ser anónimo en la red les brinda para realizar sus actividades ilegales. En sus negocios dentro de la red tienen la tranquilidad de no necesitar un contacto cara a cara con sus clientes para realizar la transacción lo que conlleva un mayor porcentaje de éxito a la vez que se reduce la posibilidad de ser capturado.

Como vemos el abanico de posibilidades que Tor ofrece al usuario que busca pasar desapercibido en la red es enorme. Si en Tor decimos que hay una cantidad elevada de diferentes tipos de usuarios, aún es mayor el número de servicios ocultos en la red:

- **Comercio:** drogas, armas particulares y armamento militar, documentación de alto valor, órganos humanos, contenido sensible, contratación para servicios ilegales...
- **Comunicaciones:** chats online encriptados, correos electrónicos privados extremo a extremo, red descentralizada de mensajería, mensajería anónima con destinatario en Tor o fuera de esta red, listados de correos electrónicos...
- **Almacenamiento de archivos:** sistema descentralizado y seguro de almacenamiento, alojamiento web, directorio de archivos BitTorrent y enlaces magnet para el intercambio punto a punto...
- **Finanzas:** monedero de criptomonedas bitcoin, información completa sobre la economía de los bitcoins, envío y recepción de criptocrédito, mezcladores o blanqueadores de bitcoins "contaminados" con fondos identificables para eliminar rastros...
- **Portales de información:** dentro de Tor podemos encontrar varias wikis resistentes a la censura donde se puede editar anónimamente una vez registrado en el sitio, incluyendo Wikipedia.
- **Noticias y documentación:** podemos encontrar prensa original de la internet superficial, enumeración y rastreo de actividades de ciertas industrias, sitios web para compartir y publicar documentos que invitaba a los usuarios a contribuir con información de identificación personal, o de cualquier persona de interés, sitios de noticias dedicados a eventos sobre la web profunda con entrevistas y reseñas sobre mercados de la darknet, servicios ocultos de Tor, acciones legales, privacidad, bitcoin y noticias relacionadas.

- **Sistemas operativos:** directorios donde encontramos sistemas operativos de código abierto como son Debian o Qubes OS al igual que versiones reducidas e infinidad de packs de actualización.
- **Denuncias:** softwares para la habilitación de plataformas para informantes, redes globales de periodistas independientes que informan sobre temas políticos y sociales, revistas electrónicas...
- **Organizaciones sin ánimo de lucro:** plataformas para recaudación de fondos por la ayuda de personas en situaciones difíciles, organizaciones que financian y apoyan la libertad de expresión y de prensa, organización de defensa de los derechos y libertades de los ciudadanos en Internet, activistas de la red...
- **Pornografía:** en Tor se ofrecen todos los niveles existentes de pornografía y del negocio del sexo.
- **Motores de búsqueda:** rastreo de ficheros BitTorrent, metabúsqueda segura, repositorios de artículos académicos, motores que se apoyan en wikis para mejorar la búsqueda del usuario manteniendo su privacidad...
- **Redes sociales y foros:** sitios web donde encontramos casi cualquier contenido, desde foros de discusión o de contenido muy similar a cualquier foro que podemos encontrar en el internet superficial hasta foros destinados al cibercrimen.

## 2.9. Vulnerabilidades de Tor

Tor, al ser una red que otorga anonimato de baja latencia, posee una parte donde transmite la información de manera encriptada y otra en donde los datos se encuentran en claro sin ningún tipo de protección.

Tor no puede proteger contra la supervisión del tráfico en los límites de la red Tor, es decir, el tráfico que ingresa y sale de la red. Si bien Tor proporciona protección contra el análisis del tráfico, no puede evitar la confirmación de éste.

Aun conociendo esta realidad podemos afirmar que trabajar sobre esta red es hoy en día una de las formas más seguras de navegar en la red sin identificarse.

En el año 2009 se demostró que Tor es más resistente a las técnicas de huellas digitales de los sitios web que otras alternativas en la creación de túneles, como son los protocolos convencionales de VPN. La razón se debe a la reconstrucción de datos en los paquetes, las VPN de un único salto no precisan de esta acción mientras que Tor sí.

La huella dactilar del sitio web otorga una precisión por encima del 90% para identificar paquetes HTTP en protocolos VPN en comparación con Tor, el cual dio una precisión del 2,96%.**[22]**

Estos datos no quieren decir que Tor sea una tecnología superior a la de VPN, cada una de ellas funciona de mejor manera en unos entornos específicos. **[50]**

Vamos a deshojar un poco más cada una de estas tecnologías con el fin de conocer cuál es la más eficaz en cada situación que se nos presente en la red.

Lo primero que debemos tener claro es que tanto VPN como Tor pueden usarse para eludir la censura de Internet y proteger su privacidad.

También se puede usar VPN o Tor para superar filtros de contenido geo-restringido al conectarse a un nodo que se encuentre en una ubicación que tenga acceso sin restricciones al contenido que se desee.

Tor		VPN	
✓ Ventajas	✗ Desventajas	✓ Ventajas	✗ Desventajas
<ul style="list-style-type: none"> <li>-Nadie puede rastrear los sitios que visita de regreso a su dirección IP.</li> <li>-Al ser una red distribuida, hace que sea extremadamente difícil para cualquier gobierno u organización cerrarla.</li> <li>-Tecnología de acceso gratuito.</li> </ul>	<ul style="list-style-type: none"> <li>-Es lento debido a que sus datos se enrutan a través de varios relevos.</li> <li>-Algunos ISP bloquean activamente los relays Tor, lo que dificulta la conexión de algunos usuarios.</li> <li>-Como el tráfico en el nodo de salida no está encriptado, cualquiera que esté ejecutando el exit relay puede ver su tráfico. Además, cualquiera puede configurar un nodo de salida Tor y espiar a los usuarios.</li> </ul>	<ul style="list-style-type: none"> <li>-La velocidad de conexión es mucho más rápida respecto a la de Tor ya que solo existe el servidor VPN que se encuentra entre el ordenador y el sitio solicitado.</li> <li>-Algunos proveedores de VPN incluyen protección contra malware en el software del cliente.</li> <li>-VPN proporciona una gran privacidad y seguridad.</li> </ul>	<ul style="list-style-type: none"> <li>- Un buen servicio VPN resultado costoso.</li> <li>-Si se busca seguridad es obligatorio un servicio de VPN de calidad que no almacene datos o registros de las comunicaciones.</li> </ul>

Tabla 2.1: Comparación Tor y VPN

Con toda la información acerca del funcionamiento de los anteriores sistemas podemos decir que no hay un ganador claro, como casi todo en la vida hay matices a valorar, en este caso será el objetivo que busquemos.

Si solo se desea navegar en la red de forma anónima, Tor es una opción más que efectiva. Pero si se busca una mejor protección de la privacidad, seguridad mejorada de piratas informáticos, ladrones de identidad y malware, y sobre todo un aumento de la velocidad, entonces un servicio de VPN es probablemente una mejor opción.

Sabemos que Tor ofrece mucho y normalmente con seguridad si se emplea HTTPS, pero no es un sistema a prueba de balas.

Existe una gran cantidad de ataques que pueden vulnerar la seguridad de este sistema como son:

- **Escucha de nodos de salida**

En septiembre de 2007, Dan Egerstad, un consultor de seguridad sueco reveló que había interceptado nombres de usuario y contraseñas para cuentas de correo electrónico al monitorear los exit relays de Tor.

Como Tor no puede encriptar el tráfico entre un nodo de salida y el servidor de destino, cualquier nodo de salida está en posición de capturar el tráfico que pasa a través de él que no usa encriptación de extremo a extremo como Secure Sockets Layer (SSL) o Transport Layer Security (TLS).

- **Ataques directos sobre relays**

El personaje principal de este ataque es siempre un relay. Se puede buscar la inhabilitación o en caso contrario el uso obligatorio de un nodo como entry guard o exit relay en una ruta de la red Tor.

En un primer plano se busca la generación de bucles o ciclos infinitos entre relays provocándoles una inutilización al no poder ser empleados en la creación de nuevos circuitos. Al reducir el número de nodos disponibles, las posibilidades de empleo de un nodo instalado por un atacante aumentan considerablemente.

Por otra parte, se puede lograr jugar con los circuitos que se van a crear en el futuro en Tor editando las características de funcionamiento de los relays existentes.

En un caso ideal para el atacante, si lograra reducir al mínimo las funcionalidades de todos los relays excepto de los que se sitúen en su propiedad, logrará que todos los usuarios que se conecten a la red busquen emplear sus nodos para tener un mejor servicio dentro de Tor y por consiguiente realizar un acto delictivo a mayor escala.

- **Ataque de análisis de tráfico**

Un atacante que logra observar ambos extremos del canal de comunicación intenta encontrar patrones en el tráfico para hacer coincidir los datos entrantes y salientes con el fin de desanonimizar a los usuarios.

Esto puede hacerse correlacionando el volumen de datos transmitidos o comparando los tiempos en los que se transmiten los paquetes.

Los ataques de correlación son un problema difícil de resolver en redes de anonimato de baja latencia como es Tor.



- **Espionaje del sistema autónomo (AS)**

Si existe un sistema autónomo (AS) en ambos segmentos de la ruta establecida, éste puede correlacionar estadísticamente el tráfico en los segmentos de entrada y salida de la ruta y atacar al destino con el que el cliente se comunica.

En 2012, LASTor propuso un método para predecir grupos de ASes potenciales en estos dos segmentos y luego evitar elegir esta ruta durante el algoritmo de selección de ruta en el lado del cliente.

- **Ataque del francotirador “Sniper attack”**

Este ataque consume muy pocos recursos para el atacante y se utiliza para inhabilitar relays de la Tor Network arbitrariamente.

El ataque funciona utilizando un cliente y servidor que poseen un acuerdo, y completando las colas del nodo de salida hasta que el nodo se quede sin memoria, y por lo tanto no puede dar servicio a otros clientes.

Al atacar una proporción significativa de los nodos de salida de esta manera, un atacante puede degradar la red y aumentar la posibilidad de que los objetivos usen nodos controlados por el atacante.

**Ataque “Bad Apple”**

Ataque capaz de revelar las direcciones IP de los usuarios de BitTorrent en la red Tor.

Este ataque contra Tor consta de dos partes:

1. Explotar una aplicación insegura para revelar la dirección IP de origen de un usuario Tor.
2. Explotar Tor para asociar el uso de una aplicación segura con la dirección IP de un usuario (revelado por la aplicación insegura). Como no es un objetivo de Tor protegerse contra los ataques a nivel de aplicación, Tor no se hace responsable de la primera parte de este ataque. Sin embargo, debido al diseño de Tor, es posible asociar las secuencias provenientes de una aplicación segura con usuarios rastreados, la segunda parte de este ataque es de hecho un ataque contra Tor.  
La segunda parte de este ataque es lo que conocemos como ataque “Bad Apple” que viene del dicho “una manzana podrida arruina el montón”.

- **Ataque de huellas dactilares del circuito**

Ataque en el que el atacante correlaciona la actividad realizada por un ratón dentro del navegador de Tor o en un navegador convencional mediante mediciones temporales con javascript.

- **Ataque de anonimato Firefox / JavaScript**

En agosto de 2013, se descubrió que los navegadores Firefox en muchas versiones anteriores del Tor Browser Bundle eran vulnerables a un ataque de JavaScript, ya que NoScript no estaba habilitado por defecto.

Los atacantes usaron esta vulnerabilidad para extraer las direcciones MAC e IP de los usuarios y los nombres de los ordenadores.

- **Modelos probabilísticos**

Mediante estudios matemáticos y conociendo ciertos datos seguros como son que un cliente siempre va a poseer una salida o que siempre se realizan saltos sobre tres relays en la red se puede llegar a dejar sin anonimato al usuario mediante estadística.

- **Ataque cellflood**

Este ataque se sitúa en el grupo de los DoS en donde se busca inhabilitar relay situados dentro de Tor mediante el constante envío de solicitudes y por consiguiente haciendo que el nodo no pueda aceptar nuevos usuarios.

“Cellflood Attack” fue descubierto en 2013 por cuatro investigadores de la Universidad de Sapienza de Roma y de la Universidad de Columbia de Nueva York.

El ataque se aprovecha del hecho de que encriptar mensajes con una clave pública es 20 veces más sencillo que abrir mensajes con una clave pública. La primera vez que se crea el circuito, las claves de sesión se distribuyen utilizando la clave pública del relay con las CREATE cells. El coste de enviar grandes cantidades de CREATE cells a un relay es menos costoso para un atacante que para el relay atacado.

## **2.10. Ataques cibernéticos, cultura criminal moderna**

En pleno 2017 la tecnología de internet está tan arraigada en la sociedad que cualquier usuario común es capaz de realizar actividades cotidianas en la red ya sean labores de tramitación legal o simplemente una compra de alimentación o ropa. [44]

El mayor problema de internet es su parte oscura y por desgracia aún para una gran parte de la sociedad también la parte desconocida de la red, el negocio del cibercrimen.

Mucha gente tiene una imagen de internet errónea donde solo se pueden encontrar lugares web útiles que nos facilitarán la vida. Esto no es así ya que el mundo del crimen informático crece diariamente y los ciberdelincuentes poseen mayores conocimientos y mejores herramientas para perpetrar los delitos contra el ciudadano de a pie y empresas.

Poco a poco las personas se van dando cuenta de lo que está sucediendo en la red gracias a noticias sobre ciberataques y toman las precauciones básicas para tener un cierto nivel de seguridad ante las posibles estafas del siglo XXI. Algunas de las más importantes y conocidas son: [42]

- No revelar información personal por Internet.
- Poseer un antivirus y un firewall. Es recomendable el uso de software antispyware.
- Solo rellenar formularios de registro de sitios web ligados a empresas u organizaciones conocidas y fiables.
- Confiar solo en páginas web conocidas y si es posible que incluyan el protocolo seguro de transferencia de hipertexto (HTTPS).
- Tener actualizado al día el sistema operativo y el navegador desde los que nos conectamos a la red.
- Proteger la información de usuario empleando contraseñas robustas.

- Al descargar programas desconocidos, revisa que tengan licencias. Un software con procedencia desconocida puede llevar incluido malware u otros programas maliciosos.
- No abrir nunca los ficheros adjuntos de mensajes que nos lleguen de usuarios desconocidos.

Con estos sencillos pasos un usuario normal puede garantizar su seguridad en la red y proteger su información privada.

Aunque la sociedad aun confunda términos a la hora de hablar del crimen informático, debemos diferenciar claramente el término hacker y cibercriminal ya que son dos tipos de usuarios con fines totalmente opuestos.

En la cultura de la seguridad informática a los primeros se les denomina White Hat Hackers (Hackers de sombrero blanco), a los segundos Black Hat Hackers (Hacker de sombrero negro).

La palabra “Hacker” en la cultura popular se suele asociar a la figura de un delincuente. Algo que no puede estar más lejos de toda realidad como veremos a continuación. **[43]**

Un hacker es una persona con conocimientos altos sobre computación que persigue constantemente el aumento de conocimientos y el descubrimiento de vulnerabilidades en sistemas informáticos sin fines ilegales. En la industria de la seguridad informática a esta actividad se le denomina hacking ético.

Muchas empresas de gran importancia en sus respectivos sectores contratan a hackers para el mantenimiento de la seguridad de sus sistemas y redes.

Ahora bien, un cibercriminal es una persona que se aprovecha de las vulnerabilidades de las redes y sistemas de información para llevar a cabo actos tipificados por ley como criminales como son:

- Robo de información.
- Extorsión.
- Divulgación de información confidencial.
- Distribución de pornografía infantil.
- Envío de correo basura.
- Enaltecimiento del terrorismo.
- Fraudes y robo de identidad.
- Falsificación de información.
- Piratería.

Cada vez hay más personas y dispositivos conectados en el mundo, así que no resulta sorprendente que tanto la frecuencia como la gravedad del cibercrimen se hayan disparado durante los últimos cinco años. **[39]**

Los tipos de cibercrimen han evolucionado en los últimos años, los correos spam han dado paso a ataques capaces de dejar expuestos los datos de millones de personas a grupos malintencionados. **[41]**

Los ciberataques han aumentado exponencialmente gracias a un despliegue de herramientas de ataque automatizadas de fácil acceso. Ha llegado el punto en el que el cibercrimen ha superado en ingresos al tráfico de drogas.

La empresa de ciberseguridad Ona Systems ha dictaminado los ataques más frecuentes que hemos sufrido el año 2016. **[40]**

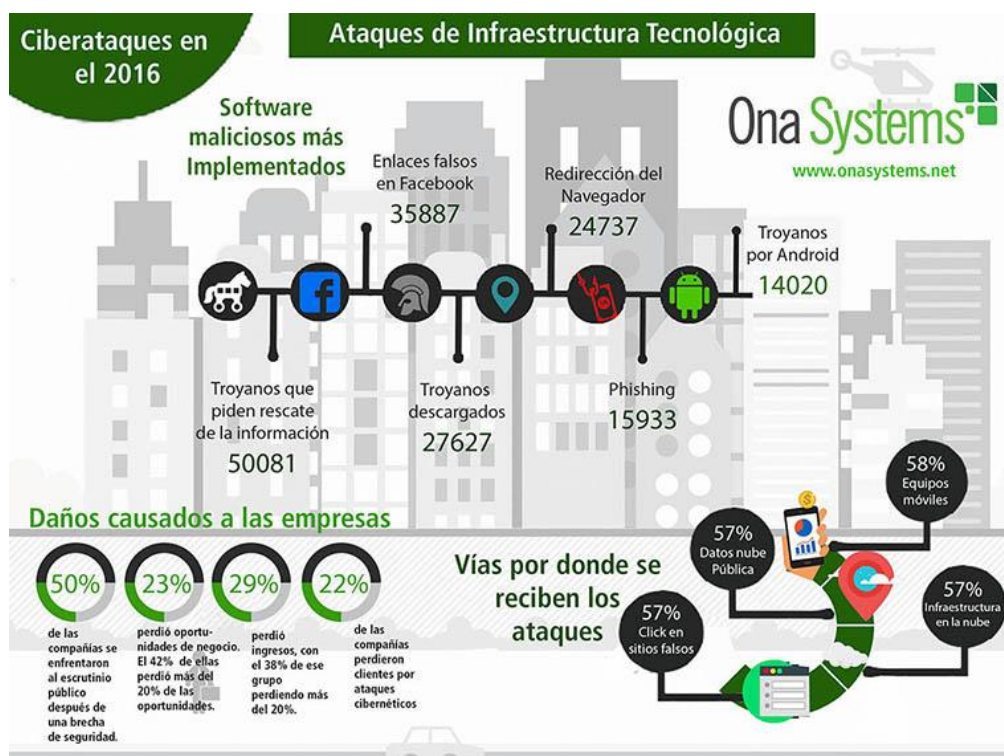


Figura 2.8: Ataques más comunes de 2016

Podemos catalogar los ataques cibernéticos en: [46] [47]

- **Puertas traseras:** el atacante emplea una puerta trasera para instalar un software de keylogging, permitiendo así un acceso ilegal al sistema.
- **Ataque de denegación de servicio DDoS:** ataques con tráfico inútil que provoca una pérdida de conectividad por el consumo del ancho de banda de la red de la víctima o sobrecarga de los recursos computacionales del sistema atacado.
- **Ataque de acceso directo:** acceso físico al sistema y con ello la posibilidad de instalación de software con gusanos informáticos.
- **Eavesdropping:** espionaje de conversaciones entre dos usuarios de una red.
- **Spoofing:** el atacante se hace pasar por otra persona mediante la creación de datos falsificados para obtener acceso a un sistema o información relevante.
- **Manipulación:** cambios en los parámetros de una página web por el atacante manteniendo la estética de la página original. Con ello se roba la identidad del cliente.
- **Ataque "Privilege Escalation":** permite al usuario tener un acceso elevado a la red, que principalmente no se le dió. El atacante tiene la ventaja de los errores de programación y permite un acceso elevado a la red.
- **Exploits:** programa o código que se aprovecha de un agujero de seguridad en una aplicación o sistema, de forma que un atacante podría usarla en su beneficio.

- **Ingeniería Social:** el ataque de una persona cercana al sistema objetivo, la cual conoce los programas empleados, el firewall y demás configuraciones que ayudan al ciberataque.
- **Ataque indirecto:** ataque realizado desde un tercer equipo en la ecuación atacado-atacante. Esto provoca dificultades en el rastreo del origen del ataque.
- **Malware:** software malintencionado diseñado para dañar o realizar acciones no deseadas en el sistema como la eliminación de archivos o recogida de datos del usuario atacado. Virus, gusanos, troyanos, etc. son tipos de malware, que pueden causar graves daños en el disco duro de un ordenador.
- **Adware:** programa que automáticamente muestra u ofrece publicidad no deseada, ya sea incrustada en una página web mediante gráficos, carteles, ventanas flotantes, o durante la instalación de algún programa al usuario, con el fin de generar lucro a sus autores.
- **Ransomware:** este ataque restringe el acceso a un sistema informático o simplemente a diferentes archivos con el objetivo de obtener un rescate por parte del usuario atacado.
- **Rootkits:** software malicioso que esconde cierto proceso o programas de detección de escaneo antivirus normal y continúa disfrutando de un privilegio de acceso a su sistema.
- **Spyware:** espía y recopila información de un ordenador y después transmite estos datos a una entidad externa sin el consentimiento del propietario del ordenador.
- **Scareware:** engaña a los usuarios de una computadora para que visiten sitios infestados de malware. Se presentan como advertencias legítimas de compañías de software antivirus que afirman que los archivos de su computadora se han infectado.
- **Troyano:** códigos maliciosos escondidos detrás de los programas genuinos que pueden permitir el acceso completo al sistema y pueden causar daños o la corrupción/pérdida de datos.
- **Virus:** programa inserta copias de sí mismo en otro archivo de ordenador e infecta las zonas afectadas.
- **Gusano:** programa que realiza acciones maliciosas y se propaga en sí a otras redes informáticas.
- **Phishing:** es una suplantación de identidad. Con este ataque se intenta adquirir información confidencial de forma fraudulenta mediante el envío de un correo electrónico aparentemente oficial, algún sistema de mensajería instantánea o incluso utilizando también llamadas telefónicas.
- **Robo de Identidad:** se roban datos personales y se emplean para cometer un fraude.
- **Robo de Propiedad Intelectual:** robo de material con derechos de autor en el que se violan los derechos de autor y las patentes.
- **Ataques contraseña:** programa automatizado para encontrar la contraseña correcta de un sistema y lograr así entrar en él.
- **Bluesnarfing:** obtener acceso a la información y datos de un teléfono mediante el uso de la tecnología bluetooth.

- **Keylogger:** software espía que tiene la capacidad de espiar a los acontecimientos en el sistema informático. Tiene la capacidad para registrar cada golpe en el teclado, sitios web visitados y cada información disponible en el sistema. Este registro se graba y envía a un receptor especificado.

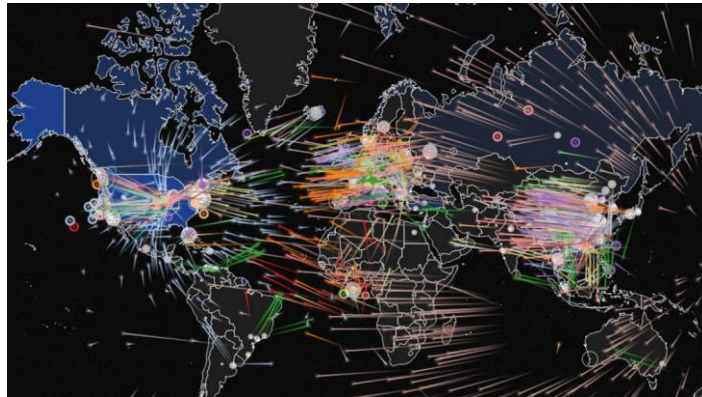


Figura 2.9: Representación mundial ciberataques

Aunque los ciberataques más comunes siguen siendo a pequeña escala, en 2017 hemos registrado un número sorprendente de ciberataques a gran escala y amenazas contra la seguridad informática global. [45]

### Filtraciones de Shadow Brokers

El grupo de hackers conocido como Shadow Brokers hizo su primera aparición en agosto de 2016.

Divulgaron vulnerabilidades de Windows como es EternalBlue, el cuál aprovecharon después cibercriminales para realizar ataques ransomware.

### Ransomware WannaCry

El 12 de mayo un ataque masivo de ransomware llamado WannaCry afectó al servicio nacional de salud de Gran Bretaña (NHS), a la empresa Telefónica de España, FedEx, Deutsche Bahn y muchos otros blancos a nivel mundial.

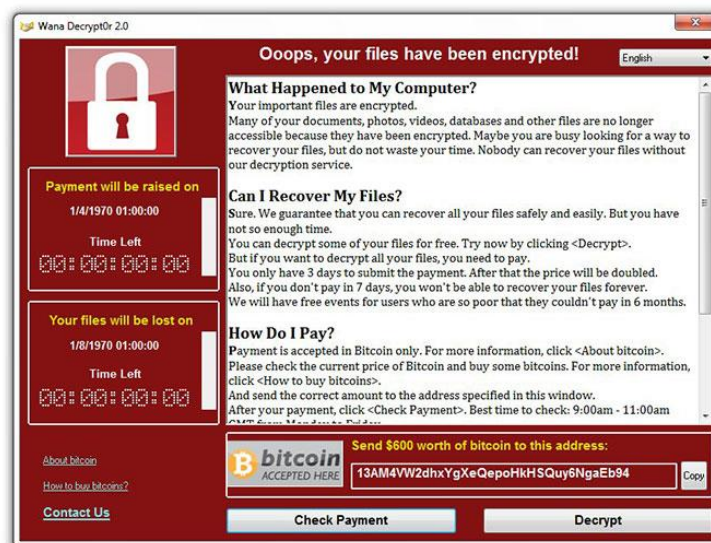


Figura 2.10: Visualización de "Wannacry"

Durante el ataque, los datos de las víctimas son encriptados, y se pide un rescate económico pagado con la criptomoneda que se emplea en la Deep Web conocida como Bitcoin, para permitir el acceso a los datos.

WannaCry fue un ataque sin precedentes infectando ordenadores personales en más de 150 países. Aunque este ataque tenía una potencia muy por encima de la mayoría de los ataques cibernéticos, por suerte presentaba defectos de diseño con los que se logró inutilizar el malware y detener así el ataque.

### **Petya**

Al mes de recibir el mundo la embestida del ransomware WannaCry, se produjo un nuevo ataque del mismo tipo aprovechando las vulnerabilidades de Windows expuestas por el grupo Shadow Brokers.

Este ransomware fue una evolución de WannaCry, aun así, presentó deficiencias en el sistema de pago.

Petya afectó a la infraestructura de Ucrania atacando compañías eléctricas, transporte público, etc.

### **Filtración Vault 7 de Wikileaks**

El 7 de marzo de 2017, Wikileaks hizo público más de 8.000 documentos que poseían: información confidencial, debilidades de Windows e iOS, herramientas de hacking y más documentación relevante.

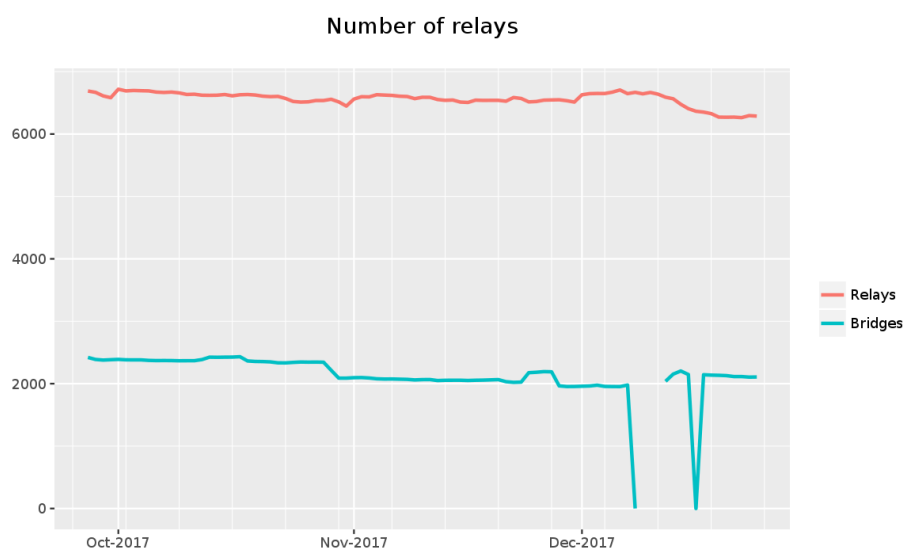
# Capítulo 3

## 3.El exit relay

La red Tor está formada por multitud de nodos conectados creando una malla resistente y descentralizada, a prueba de fallos, y la cual permite a la sociedad disfrutar de un anonimato frente al resto de usuarios.

Actualmente existen más de 6.000 nodos trabajando continuamente y dispuestos a otorgar un camino a cada usuario de Tor.

Estos nodos son únicos y con configuraciones personalizadas para la labor que van a realizar, tanto como nodo de entrada, intermedio o de salida conectándose al servidor que precisa el cliente.



The Tor Project - <https://metrics.torproject.org/>

Figura 3.1: Composición de Tor

Un nodo TOR es un tipo de proxy público, el cual conlleva una serie de riesgos a la hora de su ejecución. Se puede configurar de dos maneras, como un nodo de salida o como un nodo intermedio.

Un nodo intermedio solo retransmite el tráfico cifrado a otros nodos TOR, y no permite que los usuarios anónimos se comuniquen directamente con sitios fuera de la red TOR.

Operar cualquiera de estos nodos es útil para la red TOR en su conjunto, en especial un nodo de salida porque son particularmente escasos.

Sin embargo, ejecutar un nodo intermedio es menos peligroso ya que la dirección IP nunca aparece en los registros de acceso.

Ser propietario de un nodo en la red puede ocasionar que nuestro proveedor de servicios de internet se oponga por diferentes razones.

Sabiendo esto, la persona que decida continuar y apoyar la lucha contra el espionaje podrá configurar un relay que colabore en la causa Tor.

La cuestión es, ¿qué necesitamos para poner en línea un relay o un bridge?

La mayor parte de la sociedad piensa que se necesita material muy costoso y unos conocimientos expertos para trabajar con herramientas como podría ser un relay, y nada más lejos de la realidad ya que se precisa de muy pocos prerequisites para lograrlo.



Cualquier futuro poseedor de un relay debe cumplir con los siguientes requisitos:

- La conexión necesita un ancho de banda de por lo menos 20 kilobytes/segundo en ambas direcciones, además se necesita estar en línea constantemente cuando encendamos el dispositivo que albergue el relay.
- Necesitará una conexión con una dirección IP que sea enrutable públicamente.
- Si el dispositivo se encuentra tras un cortafuego NAT (Network Address Translation) y no tiene acceso público o dirección IP pública, necesitaremos configurar una regla de reenvío de puertos en nuestro router. Esto se puede realizar mediante “Tor Universal Plug & Play” o manualmente en el panel de control de nuestro router.

Teniendo en cuenta lo anterior, el futuro poseedor de un nodo Tor tiene la ventaja de acceder a un enorme abanico de posibilidades de configuración e instalación.

Cada relay Tor es capaz de trabajar sobre la mayoría de los sistemas operativos existentes entre los que encontramos algunos muy reconocidos y empleados a nivel global y otros menos conocidos y elegidos para labores muy específicas.

La lista completa de S.O. empleados por los relays destinados a Tor es:





-Windows XP -Windows 7 -Windows 8 -Windows 10		
-Red Hat -Debian -Fedora -OpenSUSE	-Ubuntu -Kali Linux -BlackArch Linux -Arch Assault	 <b>Linux</b>
-FreeBSD -SunOS -OpenBSD	-Darwin -NetBSD	
-Darwin		

Tabla 3.1: Sistemas operativos disponibles agrupados por distribuciones

El siguiente paso a tener en cuenta es el proceso de instalación y configuración del relay Tor. La persona colaboradora con Tor puede optar por diversas alternativas las cuales van desde un simple uso de un software específicamente diseñado para este propósito hasta la configuración completa mediante línea de comandos.

### 3.1. Vidalia

Una de las opciones clásicas y más sencillas conocidas es el software “Vidalia”. [2] [30]



Figura 3.2: Interfaz principal de Vidalia

Además de lograr colaborar con Tor rápidamente, esta conocida interfaz gráfica de usuario multiplataforma otorga una lista muy amplia de características sobresalientes relacionadas con Tor:

- Permite al usuario iniciar, detener y ver el estado de Tor.
- Permite monitorizar el uso de ancho de banda, filtrar y buscar mensajes de registro y configurar multitud de aspectos de Tor.
- Otra característica destacada es el mapa de la red Tor, permitiendo al usuario ver la ubicación geográfica de los servidores Tor, así como por donde está pasando el tráfico de la aplicación del usuario.

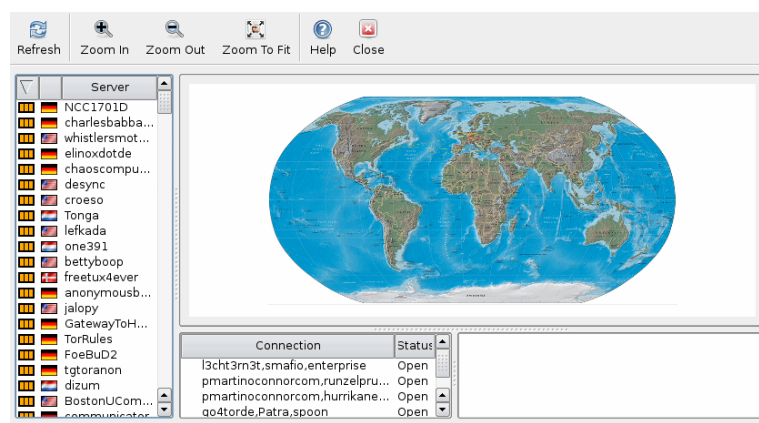


Figura 3.3: Interfaz Mapa Tor en Vidalia

- Y por último y no menos importante, con Vidalia el usuario puede configurar de la manera que desee un relay Tor o un bridge dependiendo las necesidades que se tengan.

Para nosotros como usuarios de Tor y posibles voluntarios de esta red nos interesa por encima de todo el apartado configuración de Vidalia donde encontramos la configuración relacionada con la compartición. En este punto el software nos ofrece tres alternativas:

### **Run as a client only**

La opción más simple de todas, ser cliente en busca de seguridad a la hora de navegar por la red. Aquí podemos profundizar en internet de forma anónima mediante el empleo de la red Tor.

### **Relay traffic for the Tor network**

Este punto es el necesario para todo usuario que pretenda convertirse en voluntario Tor. Al seleccionar esta opción pasamos a formar parte de la red Tor como un relay trabajando en cualquiera de sus posibles configuraciones (entry, middle o exit).

### **Help censored users reach the Tor network**

Si deseamos apoyar a usuarios sometidos a algún tipo de censura en la red podemos apoyar su causa pro derecho de la información y la comunicación mediante la instalación de un bridge Tor.

La otra gran alternativa para la creación de un relay pasa por el uso del prompt específico del sistema operativo que empleemos.

Esta opción se realiza mediante el uso de comandos en donde deberemos actualizar y mantener instaladas siempre las versiones al día para poseer las últimas actualizaciones de seguridad críticas, así como configurar al completo y mantener en funcionamiento el sistema que vamos a comunicar con Tor.

Para realizar la configuración y conseguir que nuestro futuro relay trabaje como deseemos debemos acceder al archivo torrc que guarda todos los parámetros que necesitaremos.

Una vez tengamos el nodo conectado podemos decantarnos por activar servicios como por ejemplo "Secure Shell SSH" para la posible visualización del estado del relay en cualquier dispositivo fuera de nuestra red.

Cuando situemos el nodo siempre al alcance de nuestra mano, es de vital importancia para nuestra investigación y el buen funcionamiento del sistema el empleo de una herramienta de monitorización que nos otorgue la información más completa y precisa del estado de nuestra comunicación con el resto de la red Tor. La opción mundialmente elegida y más reconocida es "Tor anonymizing relay monitor ARM".

## **3.2. Tor anonymizing relay monitor ARM**

Es un monitor de estado para Tor, el cual muestra estadísticas en tiempo real de una infinidad de elementos relacionados con Tor entre los que podemos encontrar:

- Uso de recursos como: memoria RAM, uso de CPU, ancho de banda, etc.
- Tráfico de subida y bajada por segundo y valores totales.
- Información del relay: nickname, fingerprint, puertos, dirección IP, etc.
- Configuración establecida en archivo torrc.
- Conexiones establecidas entrantes y salientes.

ARM se puede dividir en 5 páginas con información bien diferenciada entre ellas.

Lo primero que vamos a encontrar en el inicio de cada página es el tipo de sistema operativo en el que esta corriendo el relay y su versión, así como la versión de Tor.

Siempre es recomendable tener las actualizaciones al día para mantenernos más seguros frente a ataques cibernéticos y para obtener una mejor experiencia de usuario. Junto a esto vemos información esencial sobre el usuario como es: su nombre, la dirección IP empleada junto a los puertos por donde circula el tráfico, y por último las banderas que se encuentran activas en nuestro relay.

A continuación, vemos datos de funcionamiento y estado: porcentaje de trabajo de la aplicación ARM y de Tor, memoria RAM empleada y de CPU, huella única y por último las políticas de salida con las que diferenciaremos el tipo de relay dentro de la red.

## Página 1 ARM (Principal)

Es la parte principal de arm y la que más información otorga. Podemos dividir la página en dos grandes apartados.

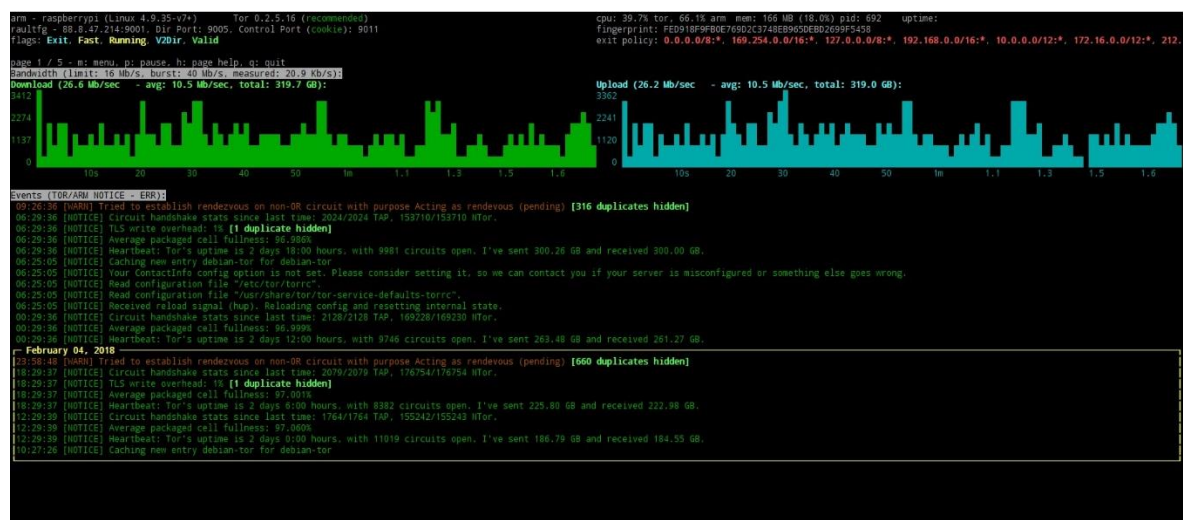


Figura 3.4: Página 1 “Tor-arm”

En la primera parte vemos dos gráficas que aportan una gran cantidad de información al usuario. La primera está destinada para representar el tráfico de descarga que posee el nodo mientras que la segunda se encarga de todo el tráfico que envía el repetidor de Tor. Decimos que este apartado otorga información muy valiosa porque es capaz de decirnos de qué manera se transmiten los datos (a ráfagas o de manera continua), y si está conectado a la red o de manera contraria, si no hay comunicación alguna.

Por último, la página primaria posee una zona de eventos donde el monitor nos informa en tiempo real de noticias sobre el estado del relay y posibles errores que se producen.

**Página 2 ARM**

Esta página va a describirnos todas las conexiones existentes de nuestro relay.

Existe un error común de visualización relacionado con estas comunicaciones dentro de ARM, el cuál produce una pérdida de conexiones dentro de la segunda página. Esto se soluciona añadiendo el comando “DisableDebuggerAttachment 0” dentro del archivo de configuración torrc.

[illegible]

Figura 3.5: Página 2 “Tor-arm”

Subsanado este pequeño inconveniente, podemos ver todas y cada una de las conexiones que maneja el relay de nuestra propiedad entre las que encontramos: conexiones entrantes y salientes con sus respectivas direcciones IP y puertos y los circuitos Tor con sus tres saltos desde el origen al destino en donde actúa nuestro relay.

Cada fila posee la dirección IP pública del relay siguiente al que nos conectamos en el circuito, las direcciones IP de nuestro propio relay, la huella única, el nombre de usuario del dueño del relay que va a continuación en la conexión, el tiempo en el cual está activa la conexión y por último el tipo de comunicación.

El caso de los circuitos es ligeramente diferente al de las otras conexiones. Al iniciar nuestro relay, Tor nos incluye siempre en cuatro circuitos como método de seguridad, los cuales no duran en línea demasiado tiempo ya que son de confirmación para la propia red.

## Página 3 ARM

Esta página es realmente útil para el propietario del relay.

```
arm - raspberrypi (Linux 4.9.35-v7+) Tor 0.2.5.16 (recommended) cpu: 60.0% tor: 14.4% arm mem: 166 MB (18.0%) pid: 692 uptime:
raulfig - 88.0.47.214:9001 Dir Port: 9005, Control Port (cookie): 9011 fingerprint: FED918F9F0E789D2C3748E0645DCB02639F545B
Flags: Exit Fast Running V2Dir Valid exit policy: 0.0.0.0/8:* 169.254.0.0/16:* 127.0.0.0/8:* 192.168.0.0/16:* 10.0.0.0/12:* 212.

page 3 / 5 - m: menu, p: pause, h: page help, q: quit
Tor Configuration (press h to show all options)
Value: 1 GB (default, dataSize, usage: # bytes|Kbytes|Mbytes|Gbytes|Kbits|Mbits|Gbits)
Description: Limit the maximum token bucket size (also known as the burst) to the given number of bytes in each direction. (Default: 1 GByte)

BandwidthRate 1 GB Average bandwidth usage limit
BandwidthBurst 1 GB Average bandwidth usage limit
RelayBandwidthRate 2 MB Average bandwidth usage limit for relaying
RelayBandwidthBurst 5 MB Maximum bandwidth usage limit for relaying
ControlPort 9011 Port providing access to tor controllers (arm, vidalia, etc)
HashedControlPassword <none> Hash of the password for authenticating to the control port
CookieAuthentication True If set, authenticates controllers via a cookie
DataDirectory /var/lib/tor Location for storing runtime data (state, keys, etc)
Log notice file /var/log/tor/notices.log Runlevels and location for tor logging
RunAsDaemon True Toggles if tor runs as a daemon process
User debian-tor UID for the process when started
ControlSocket <none> Socket or socket name to be used to connect
ControlPort <none> Port for connecting to the control port
SocksPort 0 Port for using tor as a Socks proxy
BridgeRelay False Act as a bridge
ContactInfo <none> Contact information for this relay
ExitPolicy reject 0.0.0.0/8:* reject 169.254.0.0/16:* reject 127.0.0.0/8:* reject 192.168.0.0/16:* reject 10.0.0.0/12:*... Traffic destinations that can exit from this relay
Nickname raulfig Your alias this identifier identifies the relay
ORPort 9001 Port used to accept relay traffic
PortForwarding False Use UDP or NAT-net if needed to relay
AccountingInterval 0 h Amount of traffic before hibernating
AccountingStart <none> Duration of an accounting period
SuperfloodPage <none> Publish this html file on the DirPort
DirPort 9005 Port for directory connections
HiddenServiceDir <none> Directory contents for the hidden service
HiddenServicePort <none> Port the hidden service is provided on
```

Figura 3.6: Página 3 “Tor-ARM”

A la hora de actualizar y mejorar algún parámetro de funcionamiento del relay la tercera página nos otorga una lista con todos los valores de configuración activos dentro del documento torrc.

## Página 4 ARM

En la cuarta página vemos el archivo de configuración al completo de la misma manera que se vería entrando en el directorio que alberga torrc.

```
arm - raspberrypi (Linux 4.9.35-v7+) Tor 0.2.5.16 (recommended) cpu: 40.0% tor: 14.2% arm mem: 166 MB (18.0%) pid: 692 uptime:
raulfig - 88.0.47.214:9001 Dir Port: 9005, Control Port (cookie): 9011 fingerprint: FED918F9F0E789D2C3748E0645DCB02639F545B
Flags: Exit Fast Running V2Dir Valid exit policy: 0.0.0.0/8:* 169.254.0.0/16:* 127.0.0.0/8:* 192.168.0.0/16:* 10.0.0.0/12:* 212.

page 4 / 5 - m: menu, p: pause, h: page help, q: quit
Tor Configuration (press h to show all options)
1 ## Configuration file for a typical tor user
2 ## last updated 9 October 2011 for tor 0.2.3.2-alpha
3 ## (may or may not work for much older or much newer versions of Tor.)
4 ##
5 ## Lines that begin with "## " try to explain what's going on. Lines
6 ## that begin with just "# " are disabled commands; you can enable them
7 ## by removing the "# " symbol.
8 ##
9 ## See "run tor" or https://www.torproject.org/docs/tor-manual.html
10 ## for more options you can use in this file.
11 ##
12 ## Tor will look for this file in various places based on your platform:
13 ## https://www.torproject.org/docs/faq#torrc
14 ##
15 ## Tor opens a socks proxy on port 9050 by default -- even if you don't
16 ## configure one below. Set "SocksPort 0" if you plan to run Tor only
17 ## as a relay, and not make any local application connections yourself.
18 ##
19 ##
20 ##
21 SocksPort 0 # Default: Bind to localhost:9050 for local connections.
22 ##
23 ##
24 ##
25 ##
26 SocksPort 192.168.0.1:9100 # Bind to this address:port too.
27 ##
28 ## Entry policies to allow/deny SOCKS requests based on IP address.
29 ## First entry that matches wins. If no SocksPolicy is set, we accept
30 ## all (and only) requests that reach a SocksPort. Untrusted users who
31 ## can access your SocksPort may be able to learn about the connections
32 ## you make.
33 SocksPolicy accept 192.168.0.0/16
34 SocksPolicy reject *
35 ##
36 ## Logs go to stdout at level "notice" unless redirected by something
37 ## else, like one of the below lines. You can have as many Log lines as
38 ## you want.
39 ##
40 ## We advise using "notice" in most cases, since anything more verbose
41 ## may provide sensitive information to an attacker who obtains the logs.
42 ##
43 ## Send all messages of level "notice" or higher to /var/log/tor/notices.log
44 ##
45 ##
46 ##
47 Log notice file /var/log/tor/notices.log
48 ##
49 ##
50 ##
51 ##
52 ##
53 ## Send every possible message to /var/log/tor/debug.log
```

Figura 3.7: Página 4 “Tor-ARM”

ARM permite modificar cualquiera de los valores que se encuentran dentro del archivo sin necesidad de abrirlo con un editor de texto.

## **Página 5 ARM**

Aquí se sitúa el intérprete de control, el cual no es más que un prompt incluido dentro de la propia aplicación arm que nos permite trabajar sin necesidad de salir de la monitorización.

Mediante el comando /help podemos ver una lista con los comandos disponibles y una breve descripción de los mismos.

Si introducimos /info, el prompt nos devolverá información sobre el relay especificado.

## **Fases en el establecimiento del relay**

En este punto, como poseedores de un relay Tor que se encuentra en línea y funcionando correctamente solo nos queda esperar a que la red dé el siguiente paso.

Todo relay o bridge que se prepare para trabajar en Tor debe pasar por las mismas fases, el tiempo entre éstas es independiente para cada relay pero el orden viene predefinido por la red.

Las posibles variaciones temporales entre fases dependerán del tipo de labor que va a realizar el nodo en la red y la comunicación de éste con Tor.

Todo relay que se conecte a Tor para colaborar en el encaminamiento cifrado de información será expuesto a cuatro fases diferentes en las que deberá cumplir ciertos parámetros. [7]

### **Primera fase**

En sus inicios, todo relay realiza una fase de auto-test de ancho de banda.

Tor crea cuatro circuitos en los que accedemos como nuevo relay y enviamos un tráfico por cada uno.

Con este método Tor estima el ancho de banda del relay así como las ráfagas con más información que ha hecho en un periodo de diez segundos.

Conociendo estos datos podemos deducir que nuestro relay deberá publicitar a la red un ancho de banda de:

$$4x(\text{tráfico relay}) / 10 = X \text{ Kbyte por seg}$$

Al cumplir esto, ya formamos parte de Tor y somos aceptados por los Directory Authorities.

Dependiendo del ancho de banda que ofrezcamos a la red, nuestra popularidad hacia los clientes y el uso que harán de nuestro relay aumentará o disminuirá.

Actualmente dentro de Tor podemos ver relays con un ancho de banda casi inexistente y en desuso y relays que han alcanzado la cima en la red con valores cercanos a los 100 MBytes/seg.



Esta fase inicial suele durar los tres primeros días desde la conexión, esto se debe a que la red compara nuestro ancho de banda publicitado con el del resto de relays similares hasta lograr, mediante continuas contrastaciones, una aproximación lo más exacta posible de los valores de trabajo del relay en nuestra propiedad.

## **Segunda fase**

El funcionamiento interno de la red Tor se basa en un sistema de tiempo-confianza.

En los inicios de nuestro relay no tendremos apenas peso en Tor y pasaremos prácticamente desapercibidos exceptuando por los compontes de la red que se encuentren en una situación similar, esto provocará que no recibamos apenas tráfico.

Cuando aumente el tiempo de funcionamiento de nuestro relay, la imagen que tendrá la red sobre nosotros mejorará. Se nos otorgará a continuación nuevos circuitos en los que colaborar lo que provocará una mejora en nuestro ancho de banda pasivo.

Una vez logrado esto, los bwauths (analizando nuestros parámetros de funcionamiento) nos unirán a circuitos de mayor nivel donde conectaremos con relays de mayores prestaciones lo que volverá a otorgarnos un mayor peso en Tor. Constantemente mientras pasamos el tiempo como relay voluntario de Tor, mejoramos y crecemos cíclicamente frente al resto de colaboradores haciéndonos ascender en el escalafón del sistema.

Nuestros primeros pasos y los de cualquier futuro relay Tor es mejorar nuestras prestaciones trabajando inicialmente como middle relay ya que de esta manera se evitan posibles ataques maliciosos en la red.

Los Entry Guards son clave para asegurar la privacidad de los datos en Tor.

Si en lugar de existir Entry Guards, se emplean Middle Relay como nodo de entrada, existiría una gran probabilidad de que los dos primeros saltos de nuestra ruta estuvieran en posesión de un posible atacante, poniendo la información que pase por el circuito en grave peligro.

Sabido esto, cualquier voluntario no puede ejercer como nodo de entrada.

Los requisitos para realizar esta importante labor son:

- Poseer una estabilidad en sentido tanto funcional como de seguridad en la red.
- Estar integrado el tiempo suficiente en el sistema para dar confianza al sistema.

## **Tercera fase**

Al superar las dos fases iniciales, las cuales engloban los siete primeros días de funcionamiento, nuestro objetivo en la red cambia. Ahora ya formamos parte plenamente de Tor con lo que se nos abren dos caminos a seguir.

En el primero podemos continuar trabajando como middle relay mientras que la alternativa será cambiar de tercio y comenzar labores distintas como son las de los nodos de entrada o salida.

La mejor forma para saber el estado en el que se encuentra el relay y el tipo de relay en el que se ha convertido es viendo los flags otorgados.



Los Directory Authorities son quienes asignan los flags a los relays. Éstos se otorgan mediante tres valores a evaluar:

- El ancho de banda logrado por el relay.  
A mayor bandwidth mayor será nuestra importancia dentro de Tor.
- El tiempo de actividad dentro de la red.  
El relay debe mantenerse en línea un porcentaje muy alto del tiempo consiguiendo así la estabilidad requerida.
- Tiempo de operación.  
El tiempo que lleva nuestro nodo en línea es crucial para lograr una posición de importancia dentro del sistema.

El tipo de relay influye directamente sobre la cantidad de tráfico que movemos por la red.

Tanto nosotros como los clientes que conectan con nuestro relay podemos ver los flags que tenemos otorgados. Cada uno de ellos nos define de una manera única, entre los que podemos obtener encontramos:

- **Bad exit:** se otorga al relay que posee una mala configuración, posiblemente maliciosa, provocando un mal funcionamiento de la red.
- **Fast:** los relay que poseen un ancho de banda con un considerable volumen obtienen esta bandera. La poseen la mayoría de los nodos que se encuentran actualmente activos en Tor.
- **Guard:** todo nodo de entrada recibe esta bandera. Cada cliente ve esta bandera activada en los nodos de entrada y se conecta a uno de ellos dependiendo el ancho de banda de cada uno y otros factores de funcionamiento.
- **HSDir:** el relay poseedor es un directorio de servicio oculto v2.
- **Named:** se le otorga a cada relay que posee un nickname.
- **Running:** el relay que se encuentra en línea en los últimos 45 minutos está considerado en pleno funcionamiento y se le inyecta esta bandera.
- **Stable:** se otorga cuando un relay trabaja de manera estable, sin fallos ni pérdida de servicio.
- **V2Dir:** si un relay admite el protocolo de directorio v2 tendrá esta bandera en su lista.
- **Valid:** para obtener esta bandera, un relay debe estar ejecutando una versión de Tor que no esté defectuosa y además la autoridad de directorio no debe tenerlo en su lista de posibles nodos sospechosos.
- **Unnamed:** el alias que emplea el nodo poseedor de esta bandera es empleado por otro relay de la red.
- **Exit:** esta bandera informa que el relay trabaja como último salto en cada circuito creado en Tor.

## **Cuarta fase**

La última fase es en la que el relay ya queda asentado en la red al terminar un periodo cíclico completo en el sistema de Tor.

Una vez que nuestro relay forme parte completamente de Tor podemos comenzar a investigar sobre el tráfico circulante y sobre el estado y avance del nodo.

La monitorización del relay y la visualización de su estado en tiempo real se realiza mediante la herramienta ARM.

Gracias a las aplicaciones web como son las empleadas por “Atlas Tor” y “Tor status” podemos ver nuestro posicionamiento en la red, así como gráficas totalmente actualizadas que nos informan de múltiple información de funcionamiento.

### 3.3. Atlas Tor

Atlas Tor es una aplicación web de código abierto que otorga información relacionada con los usuarios, servidores, tráfico, rendimiento y aplicaciones relacionadas con la red Tor.

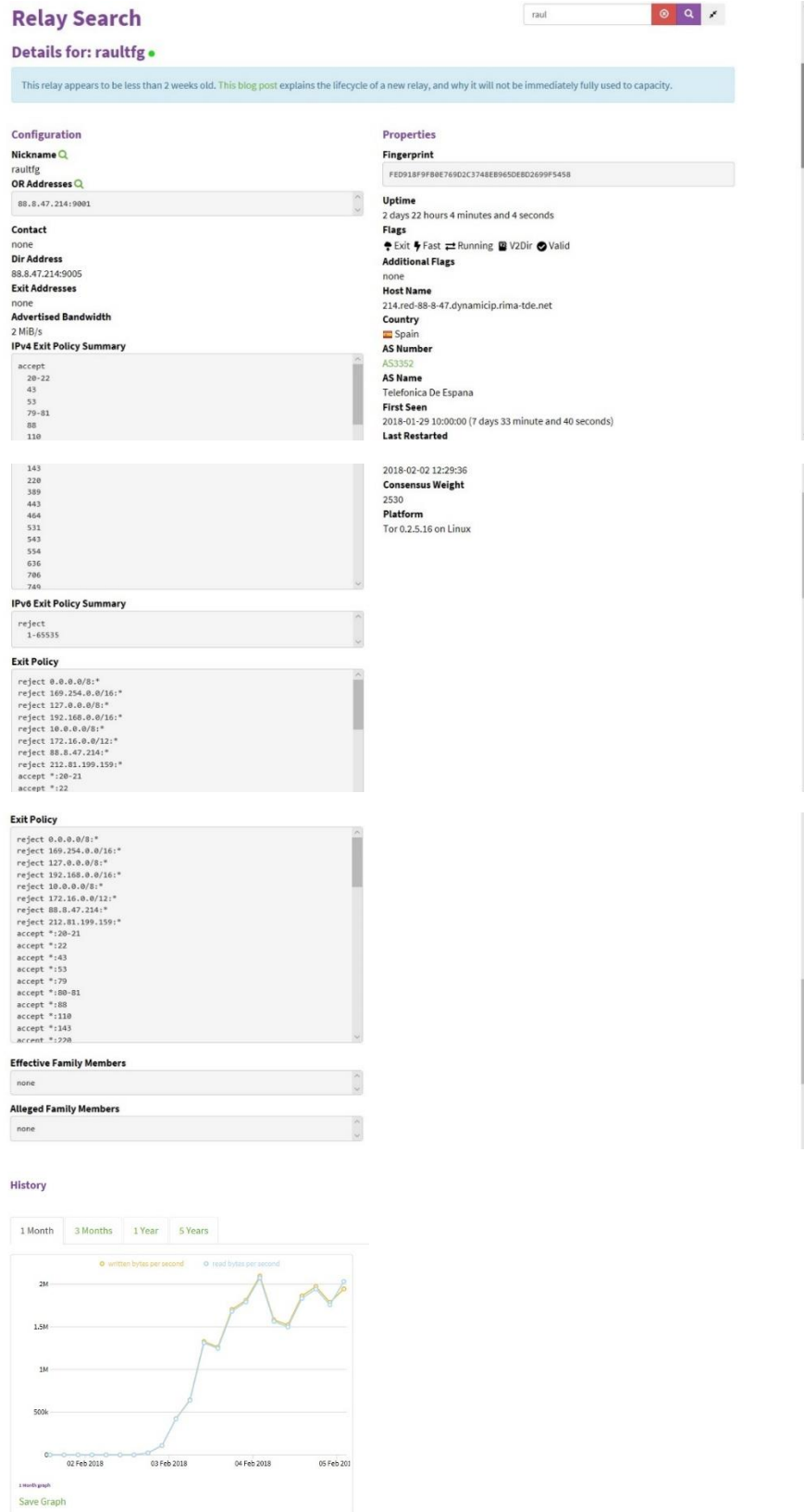


Figura 3.8: Interfaz relay individual con “Atlas Tor”

En el apartado de usuarios podemos ver información relevante como es:

- Número de clientes Tor a nivel mundial.
- Relays y bridges instalados por países.
- Países en cabeza en usuarios Tor, relays y en eventos de censura.

El apartado de servidores es uno de los más importantes y útiles en la aplicación de Atlas. Posee información como:

- Número de relays y de bridges en todo el mundo.
- Número de relays definido por banderas.
- Clasificación de relays por sistemas operativos, versiones de Tor y por direcciones IP.

Otra gran parte que encontramos en Atlas es la relacionada con el tráfico que circula por Tor:

- Ancho de banda total de Tor.
- Ancho de banda consumida y anunciada por banderas o direcciones IP.
- Ancho de banda anunciado por los relays más rápidos.

Esta aplicación Tor es la más popular entre los usuarios y voluntarios de la red favorita para el anonimato. Atlas tiene una potencia única que da una lista de servicios única e inigualable. Estos servicios se conocen como:

### **Exonera Tor**

Muestra si una IP fue utilizada por un relay Tor en una fecha determinada.

### **Relay Search**

Muestra datos sobre todos los relays y puentes situados en Tor.

### **Consensus Health**

Muestra información sobre el directorio actual de consenso.

### **Tor Map**

Muestra un mapa interactivo de los relays Tor y proporciona archivos KML.

### **OrNetStats**

Muestra estadísticas para monitorear la diversidad en la red Tor.

## DuckDuckGo

Muestra detalles del nodo Tor al incluir las palabras clave "tor node" en una búsqueda.

## Onionite

Aplicación web progresiva para ver información sobre los nodos individuales que componen la red Tor.

## Consensus Issues

Envía correos electrónicos a los operadores de las autoridades de los directorios sobre problemas de consenso.

Atlas posee datos informativos de todos y cada uno de los relays y bridges existentes en el mundo.

Vemos tres grandes grupos en cada página de usuario de Tor: configuración, propiedades e historia.

Configuración
<p><b>Nickname:</b> nombre de usuario que verá el resto de la red.</p> <p><b>OR Addresses:</b> direcciones IP v4 e IP v6 junto con el puerto por donde escucha a cada cliente.</p> <p><b>Contact:</b> detalles de contacto del operador del relay.</p> <p><b>Dir Address:</b> dirección IP v4 junto con el puerto por donde el relay escucha las solicitudes de directorio.</p> <p><b>Exit Address:</b> direcciones IP que emplea el relay para salir a internet.</p> <p><b>Advertised Bandwidth:</b> el volumen de tráfico, tanto entrante como saliente, que el relay está dispuesto a mantener.</p> <p><b>IPv4 Exit Policy Summary:</b> políticas de salida IP v4 del relay.</p> <p><b>IPv6 Exit Policy Summary:</b> políticas de salida IP v6 del relay.</p> <p><b>Exit Policy:</b> política de conexiones de salida que el relay acepta o deniega.</p> <p><b>Effective Family Members:</b> lista de relays que forman parte de un grupo o familia.</p> <p><b>Alleged Family Members:</b> lista de relays que forman parte de un grupo o familia sin considerar al relay principal incluido en el grupo.</p>

En la siguiente zona vemos gráficas con información leída y escrita y con las probabilidades que tenemos de convertirnos en un tipo u otro de relay. Podemos ver el historial del último mes o los últimos tres, o si precisamos un estudio más exhaustivo podemos llegar a ver información de hasta cinco años atrás.

Propiedades
<b>Fingerprint:</b> identificador único de 20 bytes para cada relay.
<b>Uptime:</b> tiempo que lleva en línea el relay.
<b>Flags:</b> listado de banderas que nos otorgan los “directory authorities”.
<b>Additional Flags:</b> banderas que no aparecen en el consenso de directories. Se generan por el buscador de relays.
<b>Host Name:</b> Nombre de host encontrado en una búsqueda dns inversa de la dirección IP primaria del relay. Cambia como máximo cada 12 horas.
<b>Country:</b> país donde se localiza el relay mediante una búsqueda en una base de datos de GeoIP.
<b>AS Number:</b> número de sistema autónomo.
<b>AS Name:</b> nombre de sistema autónomo.
<b>First Seen:</b> primera fecha en la que se presenta el relay de manera online.
<b>Last Restarted:</b> última vez en la que el relay fué reinicializado.
<b>Consensus Weight:</b> peso asignado a cada relay por los “directory authorities”. Este valor es empleado por los clientes en su algoritmo de selección.
<b>Platform:</b> el sistema operativo y la version Tor empleada en el relay.

## Tor status

A nivel global y como imagen de grupo Tor, la aplicación Tor status es la indicada.

Esta aplicación publica principalmente una lista completa y bien detallada de todos y cada uno de los relays existentes en el mundo clasificados por: país, nickname, ancho de banda en KB/seg, tiempo en línea, hostname, banderas conseguidas, ORPort y DirPort, fecha de primera aparición en la red, ASName y ASNumber, ancho de banda de consenso y OrAddress.

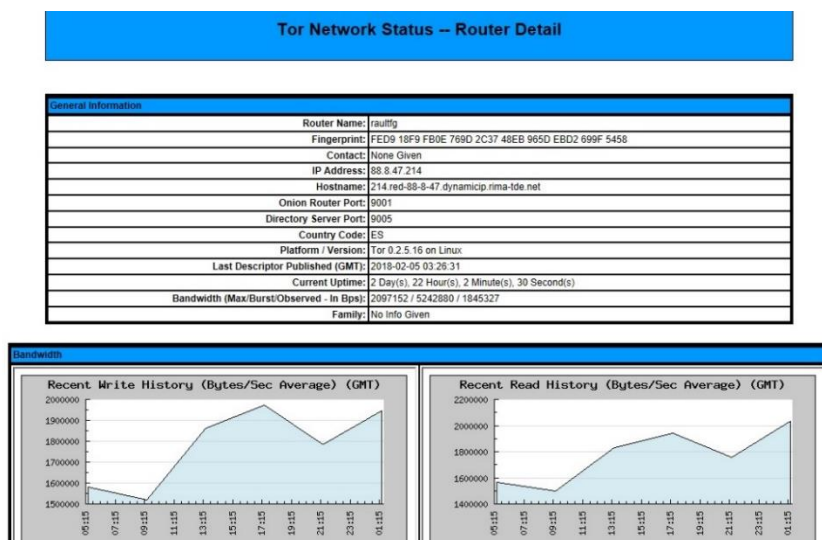


Figura 3.9: Interfaz sobre relay en “Tor Status”

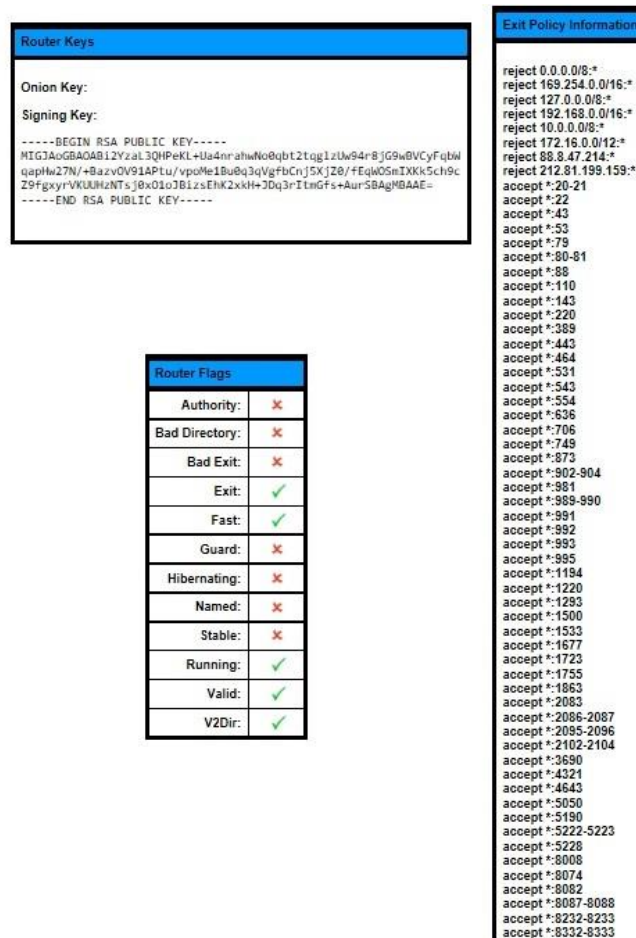


Figura 3.10: Segunda parte interfaz relay en “Tor status”

Cada relay posee en Tor status una página propia donde se detalla extensamente cada característica que posee mediante información separada en tablas y en dos gráficas las cuales indican el tráfico escrito y leído.

La gran ventaja de Tor status respecto a sus competidores es la inmensidad de opciones de clasificación dentro de Tor. Es capaz de ofrecernos búsquedas por direcciones IP específicas o puertos, así como gráficas que relacionan datos de todo tipo basados en relays individuales o grupales.

Si por cualquier motivo necesitamos llevarnos con nosotros cierta información presentada en la aplicación, ésta nos ofrece la opción de la exportación de archivos .csv para nuestro uso.

Por último, demostrando la capacidad de información que es capaz de albergar la aplicación Tor network status, aún posee opciones de visualización para que el usuario que precise encontrar información relacionada con la red Tor pueda satisfacer sus necesidades.

Tor status es capaz de publicar cuatro listas completas con todos los AS que trabajan sobre Tor basándose en el número de relays totales o únicamente de salida que poseen y en el ancho de banda de cada uno de ellos.

# Capítulo 4

## 4. Seguridad de un nodo Tor

Los últimos años están saliendo a la luz numerosos casos de espionaje gubernamental y empresarial sobre los usuarios de internet. Las personas ya no sienten seguridad a la hora de navegar por la red y buscan una alternativa para poder seguir conectado con el resto del mundo sin dar información privada.

La alternativa que elige la mayoría de los usuarios es la red Tor la cual hemos visto que es una de las opciones más eficientes y seguras existentes para obtener un anonimato frente al resto de la red.

La sociedad cree que Deep Web hace referencia a lo que conocemos como internet oscura o Dark Web, pero esto no refleja la realidad de la red profunda.

La Deep Web representa una capa única en las profundidades de internet donde precisamos de un acceso anónimo, sin embargo, ésta necesita el menor número de sistemas de seguridad y anonimidad otorgando una inmensa cantidad de datos que nunca podríamos encontrar en el internet normal.

Ya sabemos cómo funciona el sistema Tor Web para acceder a la internet profunda y al resto de niveles existentes desde el punto de vista de la propia red y sus componentes: relays, bridges y comunicaciones entre ellos, pero ¿qué ocurre cuando un cliente quiere adentrarse en las profundidades más oscuras de la red empleando Tor?

A continuación, vamos a explicar las fases que se suceden cuando un usuario busca conectarse con la red profunda mediante Tor browser.

El primer paso que va a seguir un cliente es la creación de un circuito seguro que lo proteja hasta que alcance su destino.

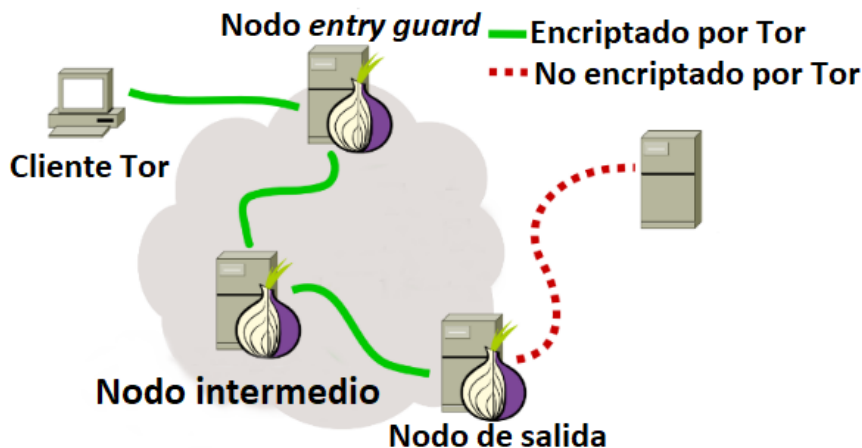


Figura 4.1: Camino en Tor

Cuando un usuario inicia su Tor Browser para acceder a todo lo que la red oculta puede ofrecerle, lo primero que hace el dispositivo que maneja es realizar un establecimiento de un enlace con uno de los relays de entrada de la red que tiene asignados en su lista de usables.



La lista de relays de entrada que puede emplear cada cliente desde un inicio es algo de lo que se encargan los “Onion Proxy (OP)” analizando la información de funcionamiento y estado de los relays de la red que proporcionan los “Directory Authorities” con lo que se pueden realizar asignaciones de la manera más correcta posible y con ello elegir los caminos que pueden ofrecer una mejor experiencia de usuario.

Cada dispositivo que se va a conectar a Tor posee un proxy de onion que conoce el relay de entrada y de salida que se va a emplear en cada futuro circuito Tor.

Esta lista la podemos encontrar en el archivo guiado por la ruta de acceso /Data/Tor/state siendo este último el nombre del fichero a examinar.

El siguiente paso en el establecimiento de un circuito Tor es la negociación de las claves para realizar el cifrado de tipo cebolla en cada salto, del cual se encarga el “Onion Proxy”.

Las claves simétricas son generadas a partir de una clave compartida la cual se logra mediante el protocolo de establecimiento de claves de Diffie-Hellman.

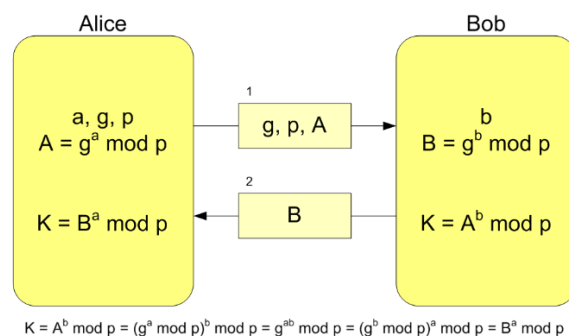


Figura 4.2: Intercambio de claves Diffie Hellman

El protocolo criptográfico Diffie-Hellman, es el protocolo de establecimiento de claves entre partes que no han tenido contacto previo, utilizando un canal inseguro y de manera anónima (no autenticada).

Se emplea como medio para acordar las claves simétricas que serán empleadas para el cifrado de una sesión. Siendo no autenticado, sin embargo, provee las bases para varios protocolos autenticados.

Su seguridad radica en la extrema dificultad de calcular logaritmos discretos en un cuerpo finito.

Dichas claves simétricas se crean a partir de un intercambio de mensajes. Cuando la sesión termina, se destruyen las claves singulares de la sesión y por consiguiente será imposible para un atacante recuperar las claves de la sesión y descifrar la información transferida por el circuito.

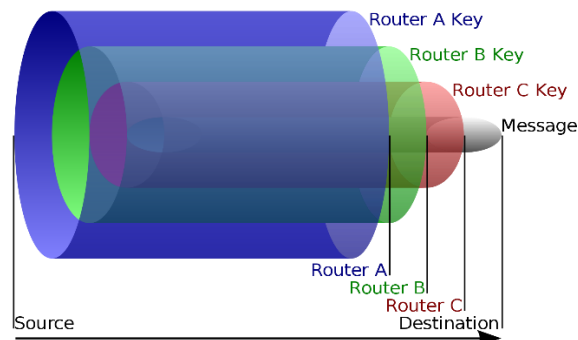


Figura 4.3: Cifrado Tor

Ya solo queda que un cliente pueda descargar información de la red de manera segura mediante la encriptación por capas que realiza Tor en cada transmisión.

Los datos transmitidos al cliente se encriptan con la clave única del nodo de salida como vemos en la última representación. [51]

Cada relay debe soportar una lista de algoritmos de cifrado con los que va a trabajar en cada encriptación de la información que pasa por él. Los algoritmos necesarios en cada relay dependen de las labores a realizar por estos en el path: [53]

- SSL DHE RSA WITH 3DES EDE CBC SHA.
- SSL DHE DSS WITH 3DES EDE CBC SHA.
- TLS DHE DSS WITH AES 128 CBC SHA.
- TLS DHE DSS WITH AES 256 CBC SHA.
- TLS DHE RSA WITH AES 128 CBC SHA.
- TLS DHE RSA WITH AES 256 CBC SHA.
- TLS DHE RSA WITH 3DES EDE CBC SHA.

Continuando el método de cifrado se encripta el mensaje resultante con la clave del nodo intermedio y por último con la del relay de entrada. [52]

De esta manera cuando el cliente se conecte con el servidor de destino, cada relay descryptará la capa que le corresponda permitiendo leer el mensaje final de manera clara.

#### 4.1. Niveles inferiores de criptografía en Tor

El grado de aceptación que posee Tor respecto a los procesos criptográficos es muy alto como hemos visto. Cada transmisión y recepción segura no podría darse sin estos algoritmos de encriptación, los cuales son responsables de otorgar la seguridad a cada usuario en su andadura dentro de la red.

El algoritmo principal empleado a la hora de encriptar cada dato es AES (Advanced Encryption Standard), el algoritmo simétrico más utilizado hoy en día.

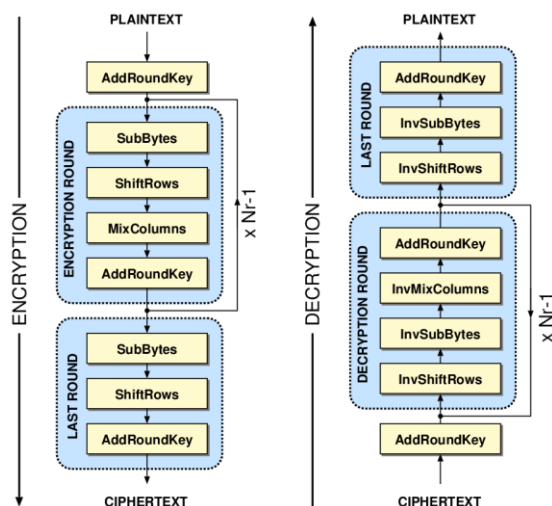


Figura 4.4: Algoritmo AES

La longitud de clave puede ser de 128 bits (10 rondas), 192 bits (12 rondas) o 256 bits (14 rondas) basado en sustituciones, permutaciones y transformaciones lineales, ejecutadas en varias veces en bloques de datos de 16 bytes.

En el caso de la encriptación en Tor, AES siempre empleará las claves más cortas, de 128 bits.

El gran problema del algoritmo AES es su descripción matemática ordenada y la escasa cantidad de rondas de cifrado que posee, lo que le sitúa como uno de los algoritmos más predecibles, provocando pérdida de seguridad.

El algoritmo empleado para la negociación de claves públicas es el RSA, nombrado así por sus tres creadores: Rivest, Shamir y Adleman.

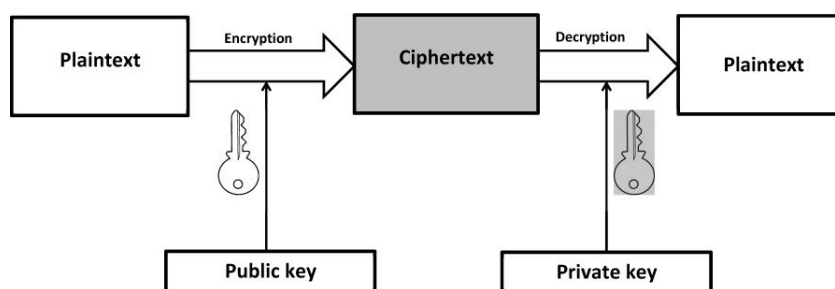


Figura 4.5: Algoritmo asimétrico RSA

Es un sistema de cifrado asimétrico desarrollado en 1977, que trabaja con dos claves diferentes: una clave “pública”, y otra “privada”. Ambas son complementarias entre sí, así que un mensaje cifrado con una de ellas sólo puede ser descifrado por su contraparte.

La longitud de clave puede ser de: 128, 256, 1024, 2048 y 4096 bits.

La elección para las claves empleadas dentro de Tor es 1024 bits.

El establecimiento de las claves las realiza el algoritmo Diffie Hellman. DH no es un algoritmo de cifrado como los vistos anteriormente, sino que se cataloga como de intercambio de claves.

Permite negociar una clave secreta entre dos nodos, a través de un canal inseguro y enviando únicamente dos mensajes (uno en cada sentido). La clave secreta que dan como resultado no puede ser descubierta por ningún atacante, aunque consiga los mensajes enviados anteriormente.

En la actualidad se ha demostrado que, aunque un atacante no puede sonsacar información relevante una vez realizado el intercambio de claves, es capaz de realizar un ataque “Man in the middle” en su inicio. Éste podría situarse entre ambas máquinas y acordar una clave simétrica con cada una de las partes, haciéndose pasar por el nodo A de cara al nodo B y, al contrario.

Una vez establecidas las 2 claves simétricas, el atacante hace de puente intermedio, descifrando la comunicación que le llega del host emisor y volviéndola a cifrar para enviársela al host receptor.

Esto se puede solucionar empleando conjuntamente un sistema de autenticación de los mensajes.

Por último, Tor emplea como función resumen SHA (Secure Hash Algorithm), más específicamente la versión SHA-1. [54]

Es una familia de funciones hash de cifrado publicadas por el Instituto Nacional de Normas y Tecnología. SHA-1 es la primera actualización de SHA publicada en 1995.

Hasta 2005 fue considerado un algoritmo seguro. Se vió comprometido después de que en 2004 se demostrase que MD5 no era seguro por parte de un equipo de investigadores chinos, a partir de aquí el tiempo de vida de SHA-1 quedó visto para sentencia.

Ya sabida la manera en que Tor otorga seguridad a cada uno de los usuarios, solo queda saber cómo se alcanza cada destino web sin el uso de crawlers web como realiza la web tradicional y el contenido que podemos encontrar dentro de las profundidades de la red.

Tanto en la Deep Web como en la Dark Web y en el resto de los niveles profundos de internet no podremos encontrar nunca direcciones web corrientes.

Todo sitio web localizado dentro de estos niveles de la red tienen en común el uso del pseudo-dominio .onion, un pseudo dominio de nivel superior genérico que indica una dirección IP anónima accesible por medio de la red Tor. [37]

Los buscadores web pueden acceder a sitios .onion usando proxys y enviando la solicitud a través de servidores de Tor. [38]

El objetivo de usar este sistema es hacer que tanto el distribuidor de la información que es usada como el receptor no se puedan rastrear, ni entre ellos, ni por un tercero.

Las direcciones con el pseudo-dominio .onion se forman mediante una combinación de 16 caracteres alfanuméricos generados automáticamente basándose en una clave pública cuando Tor es configurado. Esa combinación de 16 caracteres puede ser creada con cualquier letra del alfabeto y con dígitos decimales entre el 2 y el 7 representando un número de 80-bit en base32. [48]

Ejemplos representativos de páginas que emplean .onion son:

- **The Hidden Wiki:**  
[http://zqktlwi4fecvo6ri.onion/wiki/index.php/Main\\_Page](http://zqktlwi4fecvo6ri.onion/wiki/index.php/Main_Page)
- **The Uncensored Hidden wiki:**  
[http://uhwikih256ynt57t.onion/wiki/index.php/Main\\_Page](http://uhwikih256ynt57t.onion/wiki/index.php/Main_Page)
- **Vault 43:**  
<http://vault43z5vxy3vn3.onion/>
- **Foro "The Hub":**  
<http://thehub7dnl5nmcz5.onion/>
- **Onion soup:**  
<http://soupksx6vqh3yddda.onion/>

Existen servicios como The Duck Duck Go que pueden ser utilizados como buscador en esta "red oculta dentro de la red".

La red dispone además de buscadores como "Torch" (<http://xmh57jrznw6insl.onion/>) o "DeepSearch" (<http://xycpusearchon2mc.onion>) con los que llegar de forma directa a la información que buscamos.

La otra alternativa es ir a directorios y foros que guardan listas con direcciones .onion organizadas por temas específicos. [36]

Una vez conocida la forma en la que la red Tor crea los circuitos para cada usuario y le da a éste la suficiente seguridad y anonimato para navegar de manera fiable, y por otro lado la manera en la que se encuentra cada sitio web podemos hablar del contenido existente en la Deep Web y en el resto de los niveles de profundidad.

Sabemos que este mundo paralelo es inmenso en tamaño y en cantidad de información que alberga.

Cada nivel existente posee información de diferente índole, siendo directamente proporcional la profundidad en la que nos situemos con la sordidez y peligrosidad del contenido que podemos encontrar. [35]

La Deep Web y el resto de la “web secundaria” albergan contenido de todo tipo, algunos de los servicios e información que se encuentran en este completo y oscuro directorio son:

- Drogas y su comercialización.
- Servicios de hacking, espionaje, asesinos a sueldo, etc.
- Venta de dispositivos electrónicos liberados y robados.
- Venta de armas ilegales.
- Documentación identificativa falsa.
- Venta de información de cuentas bancarias y servicios.
- Foros con información para cometer delitos y confesiones acerca de crímenes cometidos.
- Pornografía de todo tipo.
- Información gubernamental y confidencial.
- Trucos financieros.
- Violencia explícita.
- Documentación ideológica extremista.
- Documentación para crear explosivos y otros artilugios ilegales.

## Capítulo 5

### 5. Implementación de un Exit Relay dentro de Tor

A continuación, vamos a ver la parte más práctica en la instalación y posterior configuración de cualquier relay que podemos encontrar en la red Tor actualmente. [11]

Debemos saber que no todos los tipos de relay aportan la misma seguridad a sus poseedores. Como ya hemos explicado antes hay relays con una importancia mayor en la red los cuales son en los que ponen su objetivo todos los atacantes.

El montaje de un relay intermedio es algo sencillo, tanto en su creación como en su mantenimiento.

Es el relay más básico y del cual encontramos mayor cantidad en la “Tor Network”.

Al ser un relay que se sitúa en mitad de cada circuito establecido en la red no maneja información en claro por lo que ningún usuario con malas intenciones se fijará en él, lo que hace que no peligre la red local del propietario del middle relay.

El siguiente paso en dificultad y peligrosidad para el creador de un relay es la posesión de un relay de entrada. Este tipo de relay precisa de un gran ancho de banda y un cierto tiempo establecido en la red Tor para su creación desde un nodo intermedio.

Las complicaciones aumentan con este nodo respecto a un middle relay.

Es el relay que conoce los datos del cliente que emplea el circuito Tor y asegura su privacidad, lo que le da cierta importancia en la red. Un atacante puede atacar a la conexión entre el cliente y este tipo de relay para sonsacar información sin encriptar.

Por último, tenemos el nodo de salida. El relay conocido como el más peligroso para los voluntarios de Tor y “famoso” para los ciberdelincuentes ya que descripta la información que circula por el circuito establecido para el cliente y el servidor.

Como nos gustan los desafíos, el propósito de este proyecto será establecer un “exit relay” dentro de la red y probar la imagen de atacante buscando debilidades mediante los ataques más usados sobre los relays instalados en la red.

#### 5.1. Instalación de un nodo Tor

Lo primero que debemos tener claro es la plataforma donde queremos que trabaje nuestro nodo. La opción por la que nos decantamos es el computador de placa simple Raspberry Pi (Versión 3, modelo B), el cual otorga una gran versatilidad que permite al usuario emplearla en actividades y proyectos tecnológicos de todo tipo.



Figura 5.1: Raspberry Pi 3 modelo B

Esta placa posee características de hardware muy interesantes como son:

- **Procesador:** 1.2GHz 64-bit quad-core ARMv8.
- **Memoria RAM:** 1GB SDRAM.
- **Puertos USB:** 4 puertos USB 2.0 en los que conectar periféricos o dispositivos de almacenamiento.
- **Salida de vídeo:** conector HDMI rev 1.3 y 1.4.
- **Ranura tarjeta almacenamiento:** ranura para tarjeta micro SD donde instalaremos el sistema operativo.
- **Conectividad de red:** conectividad 10/100 Ethernet mediante cable RJ-45 y WiFi 802.11n. Soporta además conexiones bluetooth 4.1.
- **Alimentación:** mediante conector micro USB 5V.

Una vez establecido lo anterior, pondremos el punto de mira sobre el sistema operativo que albergará al nodo de salida durante su vida en línea.

Teniendo en cuenta lo que es capaz de ofrecer la Raspberry con sus prestaciones, la opción lógica es la instalación de un SO liviano que no precise de un rendimiento elevado.

Por lo tanto, la elección es sencilla, Raspbian, el sistema operativo GNU/Linux de distribución libre y código abierto es perfecto para este trabajo.

La versión que emplearemos será “Raspbian Jessie”, una versión actualizada la cual cuenta con todas las herramientas que necesitamos y un entorno de escritorio gráfico amigable para el usuario. [28]



Figura 5.2: Origen de Raspbian

Empleando el software “Win32DiskImager” incluiremos el sistema operativo dentro de la tarjeta micro SD, lugar desde el que se cargará la información de arranque dentro de la placa Raspberry.

Una vez tengamos el escritorio de Raspbian preparado para su uso deberemos dar al futuro relay una dirección ip estática (privada) para poder crear unas políticas de entrada y salida con el router que empleemos.

Para ello abriremos la ventana de comandos y escribiremos:

```
sudo nano -w /etc/network/interfaces: nos permite acceder a la configuración de red de la Raspberry.
```

Una vez dentro de la configuración de las interfaces de red vemos los apartados destinados a las señales inalámbricas y por cable. [9]

Por mantener una comunicación más estable y en la que podamos garantizar un cierto nivel de seguridad siempre elegiremos conexión por cable ethernet.

Añadiremos las siguientes líneas de configuración y guardaremos los cambios: [4]

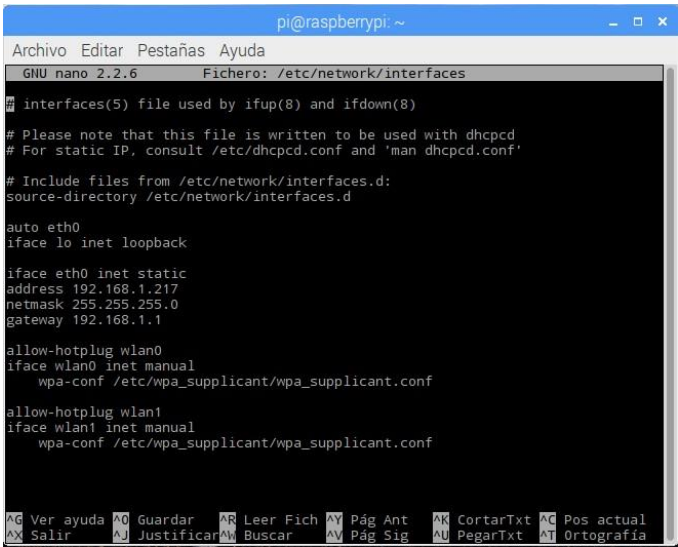


Figura 5.3: Interfaces de red de Raspberry

Dada una comunicación estable a la Raspberry, el siguiente paso nos lleva al acceso del router que emplearemos como puente de comunicación entre nuestro nodo y la red Tor. Debemos acceder a la configuración avanzada de nuestro router:

Server Name	External Port Start	External Port End	Protocol	Internal Port Start	Internal Port End	Server IP Address	WAN Interface	Remove
rasp	22000	22000	TCP/UDP	22	22	192.168.1.217	ppp0.1	<input type="checkbox"/>
orport	9001	9001	TCP/UDP	9001	9001	192.168.1.217	ppp0.1	<input type="checkbox"/>
dirport	9005	9005	TCP/UDP	9005	9005	192.168.1.217	ppp0.1	<input type="checkbox"/>

Figura 5.4: configuración router

Al permitir que nuestro router se comunique con el exterior y tengamos realizada la configuración ssh en el router y dentro del propio relay como se hará a continuación, podremos continuar su instalación desde cualquier dispositivo conectado a la red, se encuentre donde se encuentre. [10]

A continuación, nos dirigimos al terminar de Raspbian donde comenzaremos la creación de nuestro relay.

Realizaremos mediante los dos siguientes comandos una actualización completa de sistema y aplicaciones, desembocando esto en un futuro mejor funcionamiento y una mayor seguridad del sistema frente a posibles ataques:

**sudo apt-get update:** actualiza la lista de paquetes disponibles y sus versiones, pero no instala o actualiza ningún paquete.

**sudo apt-get upgrade:** una vez descargada la lista de software disponible y la versión en la que se encuentra, podemos actualizar dichos paquetes con este comando.

Vamos a terminar la configuración ssh antes presentada en la configuración del router por el que pasará toda la información.

**sudo apt-get install ssh:** instalamos el paquete ssh desde los repositorios disponibles.

**sudo /etc/init.d/ssh start:** inicializamos el servicio ssh en la Raspberry.

**sudo update-rc.d ssh defaults:** iniciamos el servicio ssh pero esta vez desde el arranque del Sistema.



Ya estamos en posición de continuar la configuración de nuestro relay desde un dispositivo distinto conectado en nuestra propia red local o desde otro dispositivo situado fuera de ella, todo ello gracias al software “Putty”. [8]

“Putty” es lo que conocemos como un cliente SSH o Telnet de código abierto disponible para una gran variedad de sistemas operativos. Nos permite controlar de manera remota otro dispositivo conectado a la red. [1]

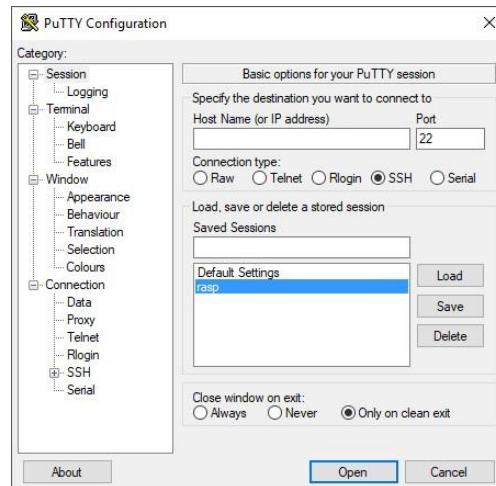


Figura 5.4: Interfaz software “Putty”

Solo falta centrarnos específicamente en la instalación y configuración del propio relay. Para ello, el siguiente paso será la instalación del servicio Tor en nuestra Raspberry mediante el comando:

**sudo apt-get install tor:** instalamos Tor mediante la búsqueda del paquete en los repositorios.

Una vez instalado Tor, tendremos todos los archivos necesarios en nuestro dispositivo. Debemos buscar a continuación el archivo que posee todos los parámetros de funcionamiento de nuestro relay.

La ruta que debemos seguir para encontrarlo es /etc/tor/ y el documento se conoce como torrc. [6]

En él encontramos toda la configuración necesaria para que el relay realice la labor en la red que nosotros buscamos. [3]

Los parámetros que precisamos editar son:

**SocksPort 0:** parámetro para configurar un proxy. Dando el valor 0 decimos que no queremos trabajar de ninguna otra manera que como un relay de la red Tor.

**Log notice file /var/log/Tor/notices.log:** ruta donde se encuentra el documento que realiza las labores de registro donde se almacenan todas las noticias y cambios que se dan lugar en nuestro relay.

**RunAsDaemon 1:** este parámetro configurado a valor 1 hace que nuestro relay se mantenga corriendo en todo momento (primer y segundo plano).

**ControlPort 9011:** es el puerto en el que Tor escuchará conexiones locales.

**ORPort 9001:** es el puerto que se anunciará a las conexiones de Tor entrantes.

**Nickname raultfg:** aquí incluiremos el nombre de usuario con el que nos verá el resto de la red Tor.

**RelayBandwidthRate 2MB:** elegimos el ancho de banda que poseerá nuestro relay.

**RelayBandwidthBurst 5MB:** valor máximo por ráfaga que permitimos al relay.

**DirPort 9005:** indicamos el puerto destinado al servicio de directorio.

**ExitPolicy:** con este parámetro indicamos como debe trabajar nuestro relay, las políticas de entrada y salida que forzarán a nuestro nodo a trabajar como guard, middle o exit relay.

**DisableDebuggerAttachment 0:** añadir esta línea a la configuración solventa problemas visuales en la segunda página de la aplicación arm, la encargada de las conexiones de nuestro relay. Sin este valor no podemos visualizar todas las conexiones existentes que posee nuestro nodo.

El apartado “ExitPolicy” variará completamente si nuestro objetivo pasa por tener controlado un middle relay con posibilidad de ser un nodo de entrada o si en cambio buscamos trabajar en la red como exit relay.

Si elegimos la opción del montaje de un relay intermedio solo necesitamos crear una política de salida cerrada en la que rechazamos cualquier política de salida.

Nuestro relay en cambio busca ser uno de los pocos nodos de salida que trabajan dentro de Tor. Para su creación deberemos decidir a qué direcciones IP y puertos otorgaremos acceso y a cuáles se los denegaremos. Nuestra configuración para asegurar solo conexiones seguras y evitar en la medida de lo posible ataques no deseados es:

#### **Direcciones IP:**

```
ExitPolicy reject 0.0.0.0/8:*
ExitPolicy reject 169.254.0.0/16:*
ExitPolicy reject 127.0.0.0/8:*
ExitPolicy reject 192.168.0.0/16:*
ExitPolicy reject 10.0.0.0/12:*
ExitPolicy reject 172.16.0.0/12:*
ExitPolicy reject 212.81.199.159:*
```

#### **Puertos:**

```
ExitPolicy accept *:20-21 => FTP (File Transfer Protocol)
ExitPolicy accept *:22 => SSH (Secure Shell)
ExitPolicy accept *:43 => WHOIS
ExitPolicy accept *:53 => DNS (Domain Name System)
ExitPolicy accept *:79 => Finger
ExitPolicy accept *:80-81 => HTTP (Hypertext Transfer Protocol)
ExitPolicy accept *:88 => Kerberos
ExitPolicy accept *:110 => POP3 (Post Office Protocol v3)
ExitPolicy accept *:143 => IMAP (Internet Message Access Protocol)
ExitPolicy accept *:220 => IMAP3 (Internet Message Access Protocol v3)
ExitPolicy accept *:389 => LDAP (Lightweight Directory Access Protocol)
ExitPolicy accept *:443 => HTTPS (Hypertext Transfer Protocol over TLS/SSL)
ExitPolicy accept *:464 => Kpasswd (Kerberos password)
ExitPolicy accept *:531 => IRC/AIM (AOL Instant Messenger)
ExitPolicy accept *:543-544 => Klogin (Kerberos login)
ExitPolicy accept *:554 => RTSP (Real Time Streaming Protocol)
ExitPolicy accept *:636 => LDAP (Lightweight Directory Access Protocol over TLS/SSL)
ExitPolicy accept *:706 => SILC (Secure Internet Live Conferencing)
ExitPolicy accept *:749 => Kerberos administration
ExitPolicy accept *:873 => rsync
ExitPolicy accept *:902-904 => VMware
ExitPolicy accept *:981 => Remote HTTPS management
ExitPolicy accept *:989-990 => FTP over TLS/SSL (File Transfer Protocol)
```

```

ExitPolicy accept *:991 => Netnews Administration System
ExitPolicy accept *:992 => Telnet protocol over TLS/SSL
ExitPolicy accept *:993 => IMAP over SSL (Internet Message Access Protocol over
TLS/SSL)
ExitPolicy accept *:995 => POP3 over SSL (Post Office Protocol v3)
ExitPolicy accept *:1194 => OpenVPN (Virtual Private Network)
ExitPolicy accept *:1220 => QT Server Admin (QuickTime Streaming Server
administration)
ExitPolicy accept *:1293 => PKT-KRB-IPSec (Internet Protocol Security)
ExitPolicy accept *:1500 => VLSI License Manager
ExitPolicy accept *:1533 => Sametime
ExitPolicy accept *:1677 => GroupWise
ExitPolicy accept *:1723 => PPTP (Point-to-Point Tunneling Protocol)
ExitPolicy accept *:1755 => RTSP (Media Services)
ExitPolicy accept *:1863 => MSNP (MS Notification Protocol)
ExitPolicy accept *:2083 => Secure Radius Service
ExitPolicy accept *:2086-2087 => GNUUnet, ELI
ExitPolicy accept *:2095-2096 => NBX
ExitPolicy accept *:2102-2104 => Zephyr
ExitPolicy accept *:3690 => SVN (Subversion version control system)
ExitPolicy accept *:4321 => RWHOIS (Referral Who is Protocol)
ExitPolicy accept *:4643 => Virtuozzo
ExitPolicy accept *:5050 => MMCC (Yahoo Messenger)
ExitPolicy accept *:5190 => ICQ and AOL Instant Messenger
ExitPolicy accept *:5222-5223 => XMPP, XMPP over SSL (Extensible Messaging and
Presence Protocol)
ExitPolicy accept *:5228 => Android Market
ExitPolicy accept *:8008 => HTTP alternate
ExitPolicy accept *:8074 => Gadu-Gadu (instant messaging client)
ExitPolicy accept *:8082 => HTTPS Electrum Bitcoin
ExitPolicy accept *:8087-8088 => Simplify Media SPP Protocol, Radan HTTP
ExitPolicy accept *:8232-8233 => Zcash
ExitPolicy accept *:8332-8333 => Bitcoin
ExitPolicy accept *:8443 => PCsync HTTPS
ExitPolicy accept *:8888 => HTTP Proxies, NewsEDGE
ExitPolicy accept *:9418 => Git pack transfer service
ExitPolicy accept *:10000 => Network Data Management Protocol
ExitPolicy accept *:11371 => OpenPGP hkp
ExitPolicy accept *:19294 => Google Voice TCP
ExitPolicy accept *:19638 => Ensim control panel
ExitPolicy accept *:50002 => Electrum Bitcoin SSL
ExitPolicy accept *:64738 => Mumble (voIP)

```

#### **Rechazamos el resto de conexiones:**

```
ExitPolicy reject *.*
```

Para que la anterior configuración tenga efecto sobre nuestro relay deberemos reiniciar el servicio relacionado con Tor:

**sudo /etc/init.d/tor restart:** reiniciamos el servicio Tor en nuestra raspberry.

**cat /var/log/tor/log:** ejecutamos el registro de Tor para comprobar si esta funcionando correctamente. Si esto es así, podemos afirmar finalmente que poseemos un nodo operative dentro de la red Tor.

Terminando con la configuración del relay, cerramos el círculo instalando la aplicación nombrada anteriormente “Tor-arm” (Tor anonymizing relay monitor) para controlar en todo momento el estado de nuestro relay y así monitorizar y estudiar el funcionamiento de éste:

**sudo apt-get install tor-arm:** comando con el que instalamos la aplicación “arm” en nuestra raspberry.

**sudo -u debian-tor arm:** ponemos en funcionamiento la aplicación.

## 5.2. Primer contacto con Tor-arm

Cuando cargamos por primera vez la aplicación “Tor-arm” nos podemos llegar a ver superados por la enorme cantidad de información que transmite al usuario. Gracias al buen diseño de la aplicación, separando los datos en diferentes páginas y apartados, al poco tiempo de comenzar a usarla podremos encontrar el dato que buscamos sin ningún tipo de problema y conseguiremos una gran fluidez en su uso.

En un primer momento la zona estática que encontramos en la parte superior de cada página de arm es la que nos va a dar la información más relevante frente al funcionamiento y estado inicial de nuestro relay.

En nuestro relay vemos la dirección IP pública que dará a conocer la red local donde nos situamos: 62.42.28.71

Podemos ver además datos relevantes como las políticas de salida que hemos explicado anteriormente y las banderas que se nos otorgarán con el paso del tiempo.

En nuestro caso, siendo un nodo de salida, hemos obtenido los siguientes flags:

- **Exit:** bandera que indica que nuestro relay trabaja íntegramente como nodo de salida en la red.
- **Fast:** se otorga cuando nuestro relay ha sido capaz de trabajar en circuitos con un ancho de banda elevado. Al trabajar en una red con una conexión de fibra óptica de gran calidad esta bandera ha sido sencillo de conseguir.
- **Stable:** al igual que en la bandera anterior, la conexión que posee el relay nos otorga una estabilidad y una calidad de servicio que nos permite trabajar en circuitos de larga duración.
- **Running:** desde que nuestro relay demuestra a la red que está funcionando y conectado, poseemos esta bandera.
- **Valid:** la red nos ha validado como nodo Tor.

Vamos a explicar, desde el punto de vista de un voluntario de la red Tor la, información encontrada en cada página a la que hay que deberemos dar un mayor valor y trascendencia.

### Página 1

Al conectar nuestro relay a la red, “Tor ARM” comenzará a darnos un registro temporal con todos los sucesos que ocurran en nuestro dispositivo.

En este registro se nos informa de la realización de un test de ancho de banda, el porcentaje de arranque que lleva nuestro relay mientras crea sus primeros circuitos de prueba y todo tipo de “consejos” con los que mejorar el servicio.

La otra gran parte de la página inicial de “Tor-ARM” enseña el tráfico de carga y descarga mediante dos gráficas bien diferenciadas.

## **Página 2**

Esta página nos enseña todas las conexiones entrantes y salientes, así como los circuitos de tres saltos en los que participa nuestro relay.

Lógicamente la lista cambia con el tiempo ya que los circuitos se abren y cierran constantemente según las necesidades de los clientes de Tor.

Cuando inicializamos Tor, se crean cuatro circuitos donde se nos incluye.

Es un sistema que crea la red para comprobar si vamos a actuar como clientes dentro de la red, en caso contrario, estos circuitos desaparecen al poco tiempo.

En cada conexión podemos ver, mediante la tecla “enter”, información sobre cada participante del enlace o circuito que visualizamos. De esta manera la red está más unificada mediante el posible contacto entre todos los voluntarios y clientes de Tor.

## **Página 3**

Como ya sabemos esta tercera página nos resume de la forma más óptima posible toda la configuración que hemos implantado en el relay.

Dentro de “Tor-ARM” tenemos la opción de visualizar la configuración al completo como si nos encontrásemos dentro del mismo archivo torrc y, por otro lado, podemos editar cualquiera de nuestros parámetros según nuestra necesidad.

## **Página 4**

Esta página es exactamente igual que el archivo torrc, incluido en la propia aplicación para darnos la facilidad de edición sin necesidad de salir y navegar por los directorios de nuestra raspberry.

## **Página 5**

La última página nos otorga una ventana de comandos muy útil con la que realizar todas nuestras actividades.

Por último, cuando necesitemos ayuda dentro de la aplicación tenemos en esta quinta página un asistente mediante el comando “help” que nos da un listado de los comandos usables dentro del prompt.

### **5.3. Ataques y vulnerabilidades de nuestro relay**

La red Tor, y el resto de las redes que cumplen funciones similares, dan a cada usuario de internet confianza para navegar sintiéndose seguro mediante los sistemas de seguridad que tienen implantados.

Esto no asegura la integridad permanente de cada nodo de la red Tor ya que no es un sistema infalible frente a los ciberataques existentes en la actualidad y menos aún frente a los que están por aparecer en el futuro.

Vamos a demostrar que, aunque Tor es casi infranqueable como una única red, los relays que habitan en ella están muy expuestos a todo tipo de ataques día tras día y a sufrir pruebas de penetración.

### 5.3.1. Tortazo

La protagonista para llevar a cabo esta explicación es la herramienta llamada “Tortazo”, escrita en lenguaje Python.

Posee tres modos de operación: recolección de información, modo “Botnet” y modo “Onion Repository”.

Se trata de una herramienta que, aunque se encuentra aún en estado de desarrollo, posee ya un número considerable de características con las que auditar un “Exit relay” y extraer así casi cualquier tipo de información. Las principales funcionalidades para comprometer cualquier nodo de salida de Tor son:

- Permite extraer información de los “Server descriptors” del último consenso emitido por los “directory authorities” y así filtrar por sistema operativo y número de repetidores.
- Realiza un escaneo inicial a los nodos que encontramos empleando Nmap y genera con ello un fichero de texto con los puertos abiertos para cada uno de los dispositivos.
- En el caso de que tengamos el fingerprint del nodo de salida que queremos auditar, el cuál conseguimos en la lista de nodos de “Tor status”, podemos realizar la extracción de información y posterior análisis.
- Después de realizar el escaneo con Nmap, opcionalmente Tortazo puede buscar en Shodan(buscador de nodos) cualquier información adicional sobre el nodo de salida.
- Permite realizar ataques por diccionario contra servidores SSH y FTP. En tal caso, admite como argumento un diccionario conformado por pares de usuarios y claves. Si no se especifica un fichero de diccionario, Tortazo utilizará algunos de los ficheros de usuarios y passwords existentes en el proyecto FuzzDB. Este proceso, como todos los ataques de “bruteforcing”, es muy ruidoso y puede llevar mucho tiempo en ejecutarse.
- En el caso de que se consiga acceso a un servidor SSH, se actualiza el fichero `tortazo_botnet.bot` el cual contiene en cada línea, el host, usuario y contraseña. Esta información luego es utilizada para ejecutar comandos de forma paralela sobre todos los servidores SSH definidos en dicho fichero o solamente sobre un conjunto determinado.

Lo primero que debemos conocer son las opciones existentes que nos da “Tortazo”. Esto lo conseguimos mediante el simple comando:

```
python Tortazo.py -h
```

Para buscar nuestro nodo de salida filtrando por el sistema operativo empleado deberemos introducir:

```
python Tortazo.py -d -v -m linux
```

Con el comando `-m` podemos filtrar el repetidor por SO.

El comando `-d` nos permite conectarnos a las autoridades Tor mediante las autoridades de directorio espejo.

Gracias a `-v` activamos el modo "Verbose" el cual nos da la mayor cantidad de información que ninguna otra opción.

A la hora de emplear uno de los tres modos de funcionamiento existentes en tortazo, debemos saber qué información es la que queremos obtener, y actuar a continuación en consecuencia.

Para recolectar información podemos conectarnos directamente a las autoridades de directorio de Tor, conectarnos a los clones(espejos) de estos "directory authorities" que poseen la misma información que los anteriores más el añadido de un historial con toda la información pasada y por último tenemos la opción del empleo del protocolo de control de Tor que permite conectarse y controlar las instancias de esta red.

```
Python Tortazo.py -n 100 -v -m linux -a "-sSV -A -n"
```

La respuesta a este commando reflejado en nuestro nodo de salida es:

**Nickname:** raultfg

**OS Version:** Linux

**Fingerprint:** FED918F9FB0E769D2C3748EB965DEBD2699F5458

**Scan address:** 88.8.47.214

**Scan Arguments:** -sSV -A -n

**Scan ports:**20, 21, 22, 43, 53, 79, 80, 81, 88, 110, 143, 220, 389, 443, 464, 531, 543, 554, 636, 706, 749, 873, 902, 903, 904, 981, 989, 990, 991, 992, 993, 995, 1194, 1220, 1293, 1500, 1533, 1677, 1723, 1755, 1863, 2083, 2086, 2087, 2095, 2096, 2102, 2103, 2104, 3690, 4321, 4643, 5050, 5190, 5222, 5223, 5228, 8008, 8074, 8082, 8087, 8088, 8232, 8233, 8332, 8333, 8443, 8888, 9418, 10000, 11371, 19294, 19638, 50002, 64738.

El primer modo de recolección lo podemos realizar mediante el comando:

```
Python Tortazo.py -m linux -a "-sSV -A -n" 100 -v
```

Filtramos la búsqueda por el Sistema operativo Linux en los relays a buscar.

Con la opción `-a` indicamos opciones propias de Nmap.y añadiendo el valor 100(puede ser cualquier otro) filtramos la primera centena de nodos que nos aparecen.

Mediante la adición de `-v` decimos que queremos realizar la búsqueda

Este comando nos da los nicknames de los repetidores encontrados, el fingerprint único de cada uno, el sistema operativo empleado en cada caso y la versión de Tor incluida. Además de esta manera podemos conocer la dirección IP y los puertos por defecto que escanea Nmap.

El segundo modo en el que nos conectamos a los espejos de cada autoridad de directorio solo precisa de un pequeño cambio en la ejecución del comando respecto a su predecesor:

```
Python Tortazo.py -m linux -a "-sSV -A -n" 100 -v -d
```

De esta manera nos conectamos a los espejos y recorremos todos los registros de la base de datos(parseamos).

Mediante la opción "-sSV -A -n" realizamos el escaneo de Nmap.

El último modo de recolección, como hemos explicado anteriormente, consiste en el empleo del protocolo de control de una instancia que se mantiene en ejecución.

```
Python Tortazo.py -m linux -a "-sSV -A -n" 100 -v -c
```

#### **Start-tor-browser**

Con este modo vemos la información que nos otorgan los descriptors de Tor y con ello la información de los repetidores de salida (los que filtramos). Entre esta información encontramos: nickname, dirección pública, claves públicas RSA de entrada y salida, clave única onion, firma de entrada y de salida

La opción -c permite que nos conectemos a una instancia de Tor y que usemos los nodos de salida almacenados en los descriptors locales.

Si nuestro objetivo es la lectura del fichero "tortazo:botnet.bot", el cual contiene los servidores ssh y su información esencial, deberemos trabajar en modo "Botnet".

```
Python Tortazo.py -z all -r "uname -a;id" -v
```

Con -z all -r decimos que vamos a ejecutar "uname -a;id" como un ataque de fuerza bruta sobre los dispositivos que aparecen en el archive "tortazo:botnet.bot".

Por otra parte el añadido de -v no permite realizar la acción de modo "Verbose".

```
Python Tortazo.py -z all -o -v
```

De esta forma vemos mediante un listado todos los dispositivos existentes dentro del documento. A continuación tendremos la opción de elegir el que deseemos, y así mediante una consola que nos aparecerá en la pantalla podremos controlarlo usando los comandos necesarios.

Mediante el plugin "bruter" realizamos ataques por diccionario, actualizando continuamente el fichero anteriormente nombrado.

Por último, gracias al modo "Onion Repository" podemos descubrir servicios ocultos dentro de la Deep Web y almacenarlos a continuación en una base de datos.

Por otra parte, mediante este último modo generamos direcciones onion de forma incremental o aleatoria.



Podemos ver los servicios ocultos en los archivos de configuración “Torrc” de cada instancia.

Si poseemos una dirección parcial .onion podemos mediante este modo hacer una reconstrucción y obtener la dirección completa.

**Python Tortazo.py -R (tipo de servicio) -O (dirección parcial) -V 4rsu -v**

Con la opción -O indicamos seguidamente la dirección .onion parcial que tenemos.

Mediante -V decimos el rango de caracteres que queremos probar.

**Python Tortazo.py -R (tipo de servicio) -O (dirección parcial) -V 4rsu -v -U -T (dirección archive torrc)**

Necesitamos un proxy Socks para poder conectarnos a Tor y esto lo realizamos mediante la creación de una instancia gracias a Tortazo. Se consigue con el añadido de las opciones -U y -T y la localización del archivo torrc.

Nos devuelve una lista con todas las direcciones que poseen la parcialidad que hemos otorgado al igual que un ataque de fuerza bruta.

**Python Tortazo.py -R (tipo de servicio) -O RANDOM -V 4rsu -v -U -T (dirección archive torrc)**

De esta manera realizamos lo mismo pero genera direcciones .onion aleatorias y las va probando.

Mediante -R indicamos que “Tortazo” trabaje en modo “Onion Repository”.

Uno de los puntos fuertes de la herramienta Tortazo es la posibilidad de ejecutar plugins creados en Python con la propia aplicación. De esta manera cualquier auditor de Tor puede emplear sus propios plugins de confianza para realizar auditorías específicas.

### 5.3.2. Tor’s Hammer

Una vez vista la relativa facilidad para auditar nodos de salida instalados en la red Tor y con ello extraer información importante para vulnerar la seguridad de estos componentes vamos a ver uno de los ataques más efectivos para realizar una denegación de servicio de nodos Tor.

“Tor’s Hammer” es una herramienta escrita en lenguaje python la cual nos permite realizar ataques DDoS sobre relays Tor y servidores con sitios web.

Para usarla lo primero que deberemos realizar en un entorno Linux es su instalación de la siguiente manera:

**Apt install Tor**

Instalamos la aplicación de Tor en nuestro dispositivo.

**Apt install git**

Instalamos git, permitiéndonos más adelante obtener la aplicación “Tor’s hammer”.

**Git clone <https://github.com/dotfighter/torshammer>**

Obtenemos el “Martillo de Tor” del repositorio.

**Tor**

Activamos Tor y cargamos a continuación los descriptores de los relays.

### **Python2 torshammer.py**

Ejecutamos la aplicación la cual despliega el formato completo del comando a ejecutar y todas las opciones disponibles con las que actuar.

Estamos en disposición de comenzar un ataque de denegación de servicio a uno de los nodos instalados en la red mediante el comando:

### **Torshammer.py -t <target> [-r <threads> -p <port> -T -h]**

Con la opción -t indicamos seguidamente la víctima de nuestro ataque mediante su dirección IP o dominio web.

Gracias a -r podemos indicar el número de subprocesos existentes, con un valor predeterminado de 256.

La opción -p indica que el valor siguiente se refiere al puerto por el que actuamos, por defecto el 80 que nos da conexión http.

-T es una opción muy útil ya que nos otorga anonimato a través de Tor en la dirección 127.0.0.1:9050.

Por ultimo -h nos publica por pantalla estas opciones a modo de ayuda.

Si lo que buscamos es detener el servicio que da un sitio web solo tendríamos que conocer su dirección IP o su dominio y el puerto web por el que atacar.

### **Torshammer.py -t <IP o nombre de la web> -r 500 -p 80 -T**

Por otro lado, si lo que pretendemos es realizar un ataque sobre un nodo de Tor, deberemos estar conectados a esta red y escribir:

### **Torshammer.py -t <IP pública del nodo> -r 500 -p <Puerto abierto en la configuración del nodo> -T**

Para realizar un ataque sobre nuestro nodo de salida y así inhabilitarlo:

### **Torshammer.py -t 88.8.47.214 -r 500 -p 9001 -T**

## **5.4. Problemas técnicos en el desarrollo**

Los imprevistos en la instalación y mantenimiento del exit relay protagonista en este proyecto se han ido sucediendo por numerosos factores que explicaremos a continuación.

El mayor contratiempo nos ha surgido de las políticas de censura que tienen instaurados los proveedores de servicios en nuestro país.

Al buscar realizar la completa instalación del relay en mi propio domicilio, con buena conexión de fibra estable, dependemos del servicio que nos da nuestro distribuidor Ono-Vodafone.

En este punto llega el mayor problema, nuestro ISP bloquea conexiones del relay con la red Tor. No recibimos ningún aviso ni pista con lo que esclarecer este suceso, mientras que nuestro relay no conectará nunca con Tor.

Podemos deducir este grave contratiempo gracias al sitio web “Tor status” y su apartado que relaciona todos los nodos de la red Tor con sus respectivos ISPs. En esta tabla vemos que en nuestro país solo podemos encontrar relays que trabajan con el sistema de “Telefónica España”, “SoloGigabit” y “Jazznet” (Jazztel).

Al darnos cuenta de que nuestro proveedor de servicios no se encuentra en el listado de “Tor status”, debemos buscar una red que trabaje con esos proveedores.

La primera opción que se nos viene a la cabeza es la Universidad de Cantabria, rápidamente descartada al poseer puertos bloqueados. No permitiría que la red Tor nos alcanzase en ningún momento.

La elección óptima es el uso de una red que emplee tecnología de fibra óptica de gran ancho de banda junto con la compañía “Telefónica España”.

Por otra parte, el otro gran problema al que nos hemos enfrentado ha sido la sobrecarga de la red en la que conectamos nuestro nodo de salida.

El “exit relay” es el más costoso de todos los existentes de mantener en la red. Entonces en un primer momento, al darle un gran ancho de banda de la red, éste se “adueñaba” de toda la red con su continua transmisión y recepción de información y provocaba que nuestra red lo detectase como un robot. En este punto no se podía usar ningún otro servicio en la red.

La solución para este desagradable problema fue reducir los valores relacionados con el ancho de banda en el archivo de configuración del relay. De esta manera la situación en la red se suavizó y se pudo continuar con el funcionamiento habitual de la LAN.

Un detalle que teníamos previsto a sabiendas que íbamos a configurar un nodo de salida Tor era que necesitaríamos unos niveles de seguridad dentro de nuestro sistema para prevenir posibles futuros ataques.

Aunque esto no es una complicación en sí, se puede considerar un contratiempo por el hecho de que durante el tiempo de vida de nuestro relay vamos a tener que estar pendientes de que nuestro sistema no sea vulnerado.

Para esto, desde la primera configuración deberemos otorgar unas políticas de salida a nuestro nodo bloqueando las conexiones con direcciones IP y puertos que suelen ser objetivos para los ciberdelincuentes.

Continuamente deberemos mejorar los sistemas de seguridad de nuestra red local y nuestro nodo añadiendo políticas más restrictivas en el router que nos da salida a la red y en el propio relay.

# Capítulo 6

## 6. Conclusiones y líneas futuras

### 6.1. Conclusiones

Durante el transcurso de este proyecto hemos podido averiguar que la Deep Web es mucho más que un simple mercado ilegal repleto de sitios web bizarros.

La cultura popular nos ha llevado, mediante exageraciones y leyendas urbanas, a exagerar al máximo nivel las actividades y contenidos de esta red. Ciertamente es que esta red alberga un abanico de actividades ilegales que la dan su fama mundial, pero dentro de ella hay mucho más.

Por suerte hemos podido investigar los entresijos de la red profunda gracias a la red Tor y con ello somos capaces de afirmar que el mundo que existe en su interior es una de las invenciones más útiles para toda la sociedad.

El procesamiento de un nodo de salida Tor es arduo y precisa de unos requerimientos mínimos bastante elevados como es un gran ancho de banda que se logra mediante una línea de fibra óptica de gran capacidad, elevada potencia de computación y una integración de sistemas de seguridad óptimos.

La mayoría de ISPs existentes en España bloquean conexiones directas de sus clientes con la red Tor lo que complica un poco más la instalación de un exit relay.

Aunque sabemos que Tor es la red de redes en cuanto a seguridad otorgada al usuario se refiere, esta tecnología posee muchas vulnerabilidades ante los ataques más comunes existentes actualmente en la red.

En la red encontramos rápidamente aplicaciones, frameworks y scripts que brindan multitud de ataques directos sobre nodos Tor y, por otro lado, software que nos permite desarrollarlos nosotros mismos desde nuestro dispositivo.

Aunque la red Tor como unidad es muy robusta, cada relay de manera independiente se encuentra muy expuesto ante los ciberataques, pudiendo éstos apropiarse de información relevante o dejándoles sin servicio durante periodos prolongados.

Diariamente grupos de investigadores asociados a diferentes universidades alrededor del mundo e independientes, buscan mejorar la seguridad Tor en su constante lucha contra el cibercrimen y los gobiernos que buscan extraer información de cada usuario.

### 6.2. Líneas futuras

Las posibles continuaciones de este proyecto van en dos claros sentidos.

El primero pasa por probar las funcionalidades de todas las redes semejantes a Tor.

La primera gran opción es “PrivaTegrity”. Red que permite que los mensajes entre origen y destino se dividan de forma aleatoria en una serie de servidores con nodos mixtos, llegando de forma segura al destinatario, donde se descifra la información. [31]

Otra gran alternativa para realizar una profunda investigación es “Freenet”, red de distribución de información descentralizada y resistente a la censura, considerablemente más segura que la conocida red Tor pero con una velocidad mucho menor y sin posibilidad de conexiones en tiempo real. [32]

Dentro del primer grupo podemos centrarnos también en la opción de la red anónima mediante servidores proxy llamada “UltraSurf”. **[34]**

Por último, la opción más interesante ya que puede ser la red definitiva y por otro lado la menos investigada en estos momentos es “Riffle”. Es una especie de mezcla de red en la que se permutan los mensajes enviados desde diferentes fuentes. **[21]**

Es como un juego de cartas en el que se barajan éstas antes de distribuirlas a otros jugadores. Lo mismo sucede en Riffle, si hay miles de usuarios enviando mensajes, estos se barajan cada vez que el mensaje llega a un servidor, provocando que nadie sea capaz de decir de donde vino cada mensaje en la red asegurando elevando al máximo nivel el anonimato de cada cliente. **[33]**

El segundo camino por seguir en las investigaciones futuras pasa por el estudio de los ataques más comunes realizados en la actualidad sobre redes anónimas, y buscar como respuesta, escudos y protecciones frente a ellos que garanticen la integridad de los usuarios.

# Bibliografía

[1] **Tor-Pi exit relay (without getting raided).**

<http://www.instructables.com/id/Tor-Pi-Exit-Relay-without-getting-raided/>

12 de diciembre de 2017

[2] **Instalando un relay Tor.**

<http://write.flossmanuals.net/bypassing-es/instalando-un-relay-tor/>

12 de diciembre de 2017

[3] **Logrando el anonimato con tor parte-4.**

<https://elbauldelprogramador.com/logrando-el-anonimato-con-tor-parte-4/>

12 de diciembre de 2017

[4] **Crear repetidor de TOR (relay, exit).**

<http://www.taringa.net/posts/linux/19188505/Crear-repetidor-de-TOR-relay-exit.html>

12 de diciembre de 2017

[5] **Preservando el Anonimato y Extendiendo su Uso – Relay y Bridges en TOR – Parte VII.**

<https://thehackerway.com/2011/10/24/preservando-el-anonimato-y-extendiendo-su-uso---relay-y-bridges-en-tor---parte-vii/>

16 de diciembre de 2017

[6] **Raspberry Pi Tor relay.**

<http://www.instructables.com/id/Raspberry-Pi-Tor-relay/>

10 de enero de 2018

[7] **The lifecycle of a new relay.**

<https://blog.torproject.org/lifecycle-new-relay/>

25 de diciembre de 2017

[8] **Terminal y acceso por nombre de red a la Raspberry.**

<https://raspberryparatorpes.net/tag/putty/>

7 de enero de 2018

[9] **Poner la dirección IP fija en Raspbian.**

<https://raspberryparatorpes.net/instalacion/poner-la-direccin-ip-fija-en-raspbian/>

20 de diciembre de 2017

[10] **Conectarse a la Raspberry por Terminal.**

<https://raspberryparatorpes.net/empezando/conectarse-a-la-raspberry-por-terminal/>

12 de diciembre de 2017

[11] **24 horas en la vida de un nodo de salida de Tor.**

<https://www.fwhibbit.es/24-horas-en-la-vida-de-un-nodo-de-salida-de-tor>

15 de enero de 2018

[12] **Red TOR y Deep Web (Internet Profunda).**

<http://catedraseguridad.usal.es/blog/2014/11/red-tor-y-deep-web-internet-profunda/>

7 de enero de 2018

[13] **Deep Web, ¿qué es?,**

<https://sites.google.com/site/trabajodeepweeb/home/-que-es>

25 de noviembre de 2017

**[14] La Deep Web – Introduccion.**

<http://deepwebintroduccion.blogspot.com.es/>

20 de diciembre de 2017

**[15] Buceando por la 'Internet Invisible' Deep Web.**

<https://www.muycomputer.com/2014/01/24/deep-web-introduccion/>

25 de noviembre de 2017

**[16] Kit de supervivencia en la "deep web".**

<https://www.genbeta.com/a-fondo/kit-de-supervivencia-en-la-deep-web>

26 de diciembre de 2017

**[17] En 2021 tendremos 7 dispositivos conectados por persona.**

<https://bitlifemedia.com/2017/09/2021-tendremos-7-dispositivos-conectados-persona/>

27 de noviembre de 2017

**[18] Los hogares españoles tienen una media de 19 dispositivos conectados.**

<http://www.techweek.es/empresas/noticias/1015399002701/hogares-espanoles-disponen-media-19-dispositivos-conectados.1.html/>

27 de diciembre de 2017

**[19] Internet live stats.**

<http://www.internetlvestats.com/>

13 de diciembre de 2017

**[20] 10 características de internet.**

<https://www.caracteristicas.co/internet/>

27 de noviembre de 2017

**[21] Riffle, la red anónima del MIT que soluciona los problemas de seguridad de Tor.**

<https://es.gizmodo.com/riffle-la-red-anonima-del-mit-que-soluciona-los-proble-1783517629/>

19 de enero de 2018

**[22] Anonimato en Internet, ¿qué es mejor: VPN o TOR?.**

<http://computerhoy.com/noticias/internet/anonimato-internet-que-es-mejor-vpn-tor-37197/>

7 de enero de 2018

**[23] ¿Qué es la indexación o indexar un contenido?.**

<https://www.inboundcycle.com/blog-de-inbound-marketing/que-es-la-indexacion-o-indexar-un-contenido/> .

20 de diciembre de 2017

**[24] La Deep Web: ¿Qué es y cuáles son sus niveles?.**

<https://www.elvinculodigital.com/que-es-la-deep-web/> .

2 de diciembre de 2017

**[25] Deep Web (Niveles, y que contiene cada uno).**

<https://www.taringa.net/posts/offtopic/16047905/Deep-Web-Niveles-y-que-contiene-cada-uno.html/> .

2 de diciembre de 2017

**[26] ¿Qué es un crawler o arañas de la web y qué hacen?.**

<http://seocoaching.co/que-es-un-crawler-o-aranas-de-la-web-y-que-hacen/> .

15 de diciembre de 2017

**[27] Web Crawler ¿Qué son? y ¿Cómo Funcionan?.**

<http://vuxmi.com/web-crawler-que-son-y-como-funcionan/amp/> .

15 de diciembre de 2017

**[28] Raspbian.**

<https://es.wikipedia.org/wiki/Raspbian/> .

21 de diciembre de 2017

**[29] ¿Qué es y cómo funciona Tor?.**

<http://www.ticbeat.com/seguridad/que-es-y-como-funciona-tor/> .

16 de diciembre de 2017

**[30] Evita las restricciones de tu ISP a la red TOR utilizando bridges.**

<https://lamiradadelreplicante.com/2013/12/30/evita-las-restricciones-de-tu-isp-a-la-red-tor-utilizando-bridges/> .

15 de febrero de 2018

**[31] Nace PrivaTegrity, el sustituto de la red TOR y el VPN.**

<https://computerhoy.com/noticias/software/nace-privategrity-sustituto-red-tor-vpn-38963/> .

2 de febrero de 2018

**[32] Así es Freenet, deep web alternativa a Tor e I2P.**

<https://www.genbeta.com/a-fondo/asi-es-freenet-deep-web-alternativa-a-tor-e-i2p/>

2 de febrero de 2018

**[33] Todo lo que debes saber Riffle – La nueva red anónima más segura que TOR.**

<http://www.1000tipsinformaticos.com/2016/07/todo-lo-que-debes-saber-riffle-una-nueva-red-anonima-mejor-que-tor.html/>

2 de febrero de 2018

**[34] UltraSurf, alternativa al popular TOR para navegar anónimamente.**

<https://www.adslzone.net/article12682-ultrasurf-alternativa-al-popular-tor-para-navegar-anonimamente.html/>

2 de febrero de 2018

**[35] Dark web browser Tor is overwhelmingly used for crime, says study.**

<http://www.telegraph.co.uk/technology/2016/02/02/dark-web-browser-tor-is-overwhelmingly-used-for-crime-says-study/>

26 de diciembre de 2017

**[36] ¿Quieres saber lo que esconde la Red profunda?.**

<https://www.owldetect.com/es/blog-el-lado-oscuro/posts/directorio-onion/>

18 de enero de 2018

**[37] ¿Qué son los dominios .ONION y cómo ofrecer servicios ocultos?.**

<https://blog.segu-info.com.ar/2012/07/que-son-los-dominios-onion-y-como.html/>

12 de diciembre de 2017

**[38] Estructura de la url de la deep web.**

<http://ladeepweb1011.blogspot.com.es/2015/10/estructura-de-la-url-de-la-deep-web.html/>

17 de enero de 2018

**[39] Visualización de ataques web.**

<https://www.akamai.com/es/es/about/our-thinking/state-of-the-internet-report/web-attack-visualization.jsp/>

3 de diciembre de 2017

**[40] Estadísticas ciberataques empresas.**

<http://www.onasystems.net/estadisticas-ciberataques-empresas/>

5 de diciembre de 2017



**[41] Las 15 principales estadísticas de 2017 para IT.**

<https://revistaitnow.com/las-15-principales-estadisticas-2017/>

1 de diciembre de 2017

**[42] 10 tips de seguridad para usar Internet.**

<http://expansion.mx/tecnologia/2009/09/03/10-tips-de-seguridad-para-usar-internet/>

9 de diciembre de 2017

**[43] La Diferencia Entre Hacker Y Cibercriminal.**

<http://blog.capacityacademy.com/2013/06/24/la-diferencia-entre-hacker-cibercriminal/>

10 de diciembre de 2017

**[44] El cibercrimen hoy: Las razones tras la creciente sofisticación de los ciberdelinquentes.**

<http://www.think-progression.com/es/uncategorised/el-cibercrimen-hoy-las-razones-tras-la-creciente-sofisticacion-de-los-cibercriminales/>

16 de diciembre de 2017

**[45] Los mayores ciberataques de 2017 hasta la fecha.**

<https://www.pandasecurity.com/spain/mediacenter/noticias/ciberataques-hasta-la-fecha/>

7 de diciembre de 2017

**[46] Tipos de ataques cibernéticos.**

<http://armandof.cubava.cu/2015/06/26/tipos-de-ataques-ciberneticos-2/>

9 de diciembre de 2017

**[47] Weaknesses Tor.**

[https://en.wikipedia.org/wiki/Tor\\_\(anonymity\\_network\)#Weaknesses/](https://en.wikipedia.org/wiki/Tor_(anonymity_network)#Weaknesses/)

12 de diciembre de 2017

**[48] List of Tor hidden services.**

[https://en.wikipedia.org/wiki/List\\_of\\_Tor\\_hidden\\_services/](https://en.wikipedia.org/wiki/List_of_Tor_hidden_services/)

12 de diciembre de 2017

**[49] People use Tor.**

<https://www.torproject.org/about/torusers.html/>

13 de diciembre de 2017

**[50] Tor vs VPN.**

<https://www.hotspotshield.com/resources/tor-vs-vpn/>

20 de enero de 2018

**[51] Anonymity Online.**

<https://people.torproject.org/>

1 de diciembre de 2017

**[52] Algoritmos de cifrado.**

<https://elbinario.net/2016/04/05/algoritmos-de-cifrado-i/>

21 de enero de 2018

**[53] Secure Hash Algorithm.**

[https://es.wikipedia.org/wiki/Secure\\_Hash\\_Algorithm/](https://es.wikipedia.org/wiki/Secure_Hash_Algorithm/)

21 de enero de 2018

**[54] El cifrado SHA-1 ya no es seguro.**

<https://www.adslzone.net/2017/02/23/cifrado-sha-1-ya-no-seguro-google-lo-ha-roto-despues-22-anos/>

22 de enero de 2018

**[55] TORTAZO para auditar nodos de salida de TOR.**

<https://thehackerway.com/2014/03/05/tortazo-para-auditar-nodos-de-salida-de-tor/>

3 de febrero de 2018