

Tor(The onion router) y Riffle

Protocolos de comunicación anónima

Ignacio Aguilera Martos

Luis Balderas Ruiz

14 de noviembre de 2017

1. Tor(The onion router)
2. Riffle como solución
3. Arquitectura Riffle
4. Demo de ARM
5. Demo Tor Browser
6. Conclusiones

Tor(The onion router)

- Comunicaciones anónimas.

- Comunicaciones anónimas.
- **NO** se pretende respaldar a delincuentes.

Propósito de Tor

- Comunicaciones anónimas.
- **NO** se pretende respaldar a delincuentes.
- Es una red compleja de analizar.

Las entidades básicas de Tor son:

Las entidades básicas de Tor son:

- Nodos.

Las entidades básicas de Tor son:

- Nodos.
- Usuarios.

Las entidades básicas de Tor son:

- Nodos.
- Usuarios.
- Autoridades de Directorio.

Las entidades básicas de Tor son:

- Nodos.
- Usuarios.
- Autoridades de Directorio.

Proxys y Tor

La relación entre nodos es similar a los proxys pero no es la misma.

Nodos y sus tipos

Los nodos son las piezas fundamentales de Tor.

Nodos y sus tipos

Los nodos son las piezas fundamentales de Tor.

Tipos:

Nodos y sus tipos

Los nodos son las piezas fundamentales de Tor.

Tipos:

Nodos Guard o Nodos de entrada

Son los nodos que ocupan el primer lugar en los circuitos. Son críticos ya que conocen la identidad del usuario.

Nodos y sus tipos

Los nodos son las piezas fundamentales de Tor.

Tipos:

Nodos Guard o Nodos de entrada

Son los nodos que ocupan el primer lugar en los circuitos. Son críticos ya que conocen la identidad del usuario.

Middle Nodes o Relay

Son los nodos intermedios dentro de los circuitos. Son los más básicos.

Nodos y sus tipos

Los nodos son las piezas fundamentales de Tor.

Tipos:

Nodos Guard o Nodos de entrada

Son los nodos que ocupan el primer lugar en los circuitos. Son críticos ya que conocen la identidad del usuario.

Middle Nodes o Relay

Son los nodos intermedios dentro de los circuitos. Son los más básicos.

Exit Nodes o Nodos de salida

Son los nodos que ocupan el último lugar de los circuitos. Estos nodos tienen la información sin encriptar para enviarla al servidor de destino.

Circuito

Es un camino de nodos dentro del grafo de la red Tor. Incluye un nodo de entrada, varios nodos intermedios y un nodo de salida.

Circuito

Es un camino de nodos dentro del grafo de la red Tor. Incluye un nodo de entrada, varios nodos intermedios y un nodo de salida.

Los circuitos tienen una caducidad (modificable por el usuario) por motivos de seguridad.

Flags de calificación de nodos

- BadExit.

Flags de calificación de nodos

- BadExit.
- Fast : 100KB/s.

Flags de calificación de nodos

- BadExit.
- Fast : 100KB/s.
- Guard: 250KB/s.

Flags de calificación de nodos

- BadExit.
- Fast : 100KB/s.
- Guard: 250KB/s.
- Authority.

Flags de calificación de nodos

- BadExit.
- Fast : 100KB/s.
- Guard: 250KB/s.
- Authority.
- Exit.

Flags de calificación de nodos

- BadExit.
- Fast : 100KB/s.
- Guard: 250KB/s.
- Authority.
- Exit.
- HSDir: Hidden Service Directory.

Flags de calificación de nodos

- BadExit.
- Fast : 100KB/s.
- Guard: 250KB/s.
- Authority.
- Exit.
- HSDir: Hidden Service Directory.
- Named o Unnamed.

Flags de calificación de nodos

- BadExit.
- Fast : 100KB/s.
- Guard: 250KB/s.
- Authority.
- Exit.
- HSDir: Hidden Service Directory.
- Named o Unnamed.
- Running: 45 minutos en ejecución.

Flags de calificación de nodos

- BadExit.
- Fast : 100KB/s.
- Guard: 250KB/s.
- Authority.
- Exit.
- HSDir: Hidden Service Directory.
- Named o Unnamed.
- Running: 45 minutos en ejecución.
- Stable: 7 días en ejecución.

Flags de calificación de nodos

- BadExit.
- Fast : 100KB/s.
- Guard: 250KB/s.
- Authority.
- Exit.
- HSDir: Hidden Service Directory.
- Named o Unnamed.
- Running: 45 minutos en ejecución.
- Stable: 7 días en ejecución.
- Valid: lista negra y versión de Tor sin alterar.

Flags de calificación de nodos

- BadExit.
- Fast : 100KB/s.
- Guard: 250KB/s.
- Authority.
- Exit.
- HSDir: Hidden Service Directory.
- Named o Unnamed.
- Running: 45 minutos en ejecución.
- Stable: 7 días en ejecución.
- Valid: lista negra y versión de Tor sin alterar.
- V2Dir

Flags de calificación de nodos

- BadExit.
- Fast : 100KB/s.
- Guard: 250KB/s.
- Authority.
- Exit.
- HSDir: Hidden Service Directory.
- Named o Unnamed.
- Running: 45 minutos en ejecución.
- Stable: 7 días en ejecución.
- Valid: lista negra y versión de Tor sin alterar.
- V2Dir

Flags de calificación de nodos

- BadExit.
- Fast : 100KB/s.
- Guard: 250KB/s.
- Authority.
- Exit.
- HSDir: Hidden Service Directory.
- Named o Unnamed.
- Running: 45 minutos en ejecución.
- Stable: 7 días en ejecución.
- Valid: lista negra y versión de Tor sin alterar.
- V2Dir

Gracias a estos flags los nodos quedan valorados para saber su posición en los circuitos y su validez como nodo en general.

Ciclo de vida de un nodo

Ocurre en 4 fases:

Ocurre en 4 fases:

- Fase 1(0-3): Comprobaciones básicas. Tests de seguridad y velocidad.

Ocurre en 4 fases:

- Fase 1(0-3): Comprobaciones básicas. Tests de seguridad y velocidad.
- Fase 2(3-8): Nodo intermedio. Comprobaciones sobre Guard y Exit.

Ocurre en 4 fases:

- Fase 1(0-3): Comprobaciones básicas. Tests de seguridad y velocidad.
- Fase 2(3-8): Nodo intermedio. Comprobaciones sobre Guard y Exit.
- Fase 3(8-68): Nodo Guard. Comprobaciones de estabilidad y Exit.

Ocurre en 4 fases:

- Fase 1(0-3): Comprobaciones básicas. Tests de seguridad y velocidad.
- Fase 2(3-8): Nodo intermedio. Comprobaciones sobre Guard y Exit.
- Fase 3(8-68): Nodo Guard. Comprobaciones de estabilidad y Exit.
- Fase 4(68-...): Nodo Exit. Comprobaciones esporádicas generales.

Ocurre en 4 fases:

- Fase 1(0-3): Comprobaciones básicas. Tests de seguridad y velocidad.
- Fase 2(3-8): Nodo intermedio. Comprobaciones sobre Guard y Exit.
- Fase 3(8-68): Nodo Guard. Comprobaciones de estabilidad y Exit.
- Fase 4(68-...): Nodo Exit. Comprobaciones esporádicas generales.

Los integrantes de esta comunicación son:

Los integrantes de esta comunicación son:

- Usuario.

Los integrantes de esta comunicación son:

- Usuario.
- Servicio Oculto.

Los integrantes de esta comunicación son:

- Usuario.
- Servicio Oculto.
- Puntos Introdutorios.

Los integrantes de esta comunicación son:

- Usuario.
- Servicio Oculto.
- Puntos Introdutorios.
- Nodo Rendezvous.

Los integrantes de esta comunicación son:

- Usuario.
- Servicio Oculto.
- Puntos Introdutorios.
- Nodo Rendezvous.

Los integrantes de esta comunicación son:

- Usuario.
- Servicio Oculto.
- Puntos Introdutorios.
- Nodo Rendezvous.

Diagrama de Comunicación

Usuario ⇔ Guard ⇔ Relay ⇔ Rendezvous ⇔ Relay ⇔ Guard ⇔ Servicio oculto

Servicios Ocultos

Los integrantes de esta comunicación son:

- Usuario.
- Servicio Oculto.
- Puntos Introdutorios.
- Nodo Rendezvous.

Diagrama de Comunicación

Usuario ⇔ Guard ⇔ Relay ⇔ Rendezvous ⇔ Relay ⇔ Guard ⇔ Servicio oculto

Direcciones Onion

Los servicios ocultos tienen URLs con el dominio onion y contienen 16 caracteres.

Por ejemplo: ab2dafgh1jklmi3t.onion ó facebookcorewwi.onion.

- Server Descriptor: IP, ORPort, ...

- Server Descriptor: IP, ORPort, ...
- ExtraInfo Descriptor: Información completa del nodo.

- Server Descriptor: IP, ORPort, ...
- ExtraInfo Descriptor: Información completa del nodo.
- Micro Descriptor: Información reducida del nodo.

- Server Descriptor: IP, ORPort, ...
- ExtraInfo Descriptor: Información completa del nodo.
- Micro Descriptor: Información reducida del nodo.
- Network Status Document: fichero de consenso.

- Server Descriptor: IP, ORPort, ...
- ExtraInfo Descriptor: Información completa del nodo.
- Micro Descriptor: Información reducida del nodo.
- Network Status Document: fichero de consenso.
- Router Status Entry: Información completa de nodos incluyendo flags y cálculos heurísticos.

- Server Descriptor: IP, ORPort, ...
- ExtraInfo Descriptor: Información completa del nodo.
- Micro Descriptor: Información reducida del nodo.
- Network Status Document: fichero de consenso.
- Router Status Entry: Información completa de nodos incluyendo flags y cálculos heurísticos.
- Hidden Service Descriptor: Información del servicio oculto.

- Server Descriptor: IP, ORPort, ...
- ExtraInfo Descriptor: Información completa del nodo.
- Micro Descriptor: Información reducida del nodo.
- Network Status Document: fichero de consenso.
- Router Status Entry: Información completa de nodos incluyendo flags y cálculos heurísticos.
- Hidden Service Descriptor: Información del servicio oculto.

Puente

Nodos ocultos usados para impedir las prohibiciones de uso de Tor por gobiernos o cualquier otra entidad.

Autoridad de Directorio

Nodo con permisos totales en la red. Son los únicos nodos de confianza. Tienen como misión controlar los nodos, valorarlos y administrar la red en general.

- Correlación punto a punto.

- Correlación punto a punto.
- Pérdida de información del nodo de salida.

- Correlación punto a punto.
- Pérdida de información del nodo de salida.
- Bloqueo de los nodos de salida.

- Correlación punto a punto.
- Pérdida de información del nodo de salida.
- Bloqueo de los nodos de salida.
- Ataque DDOS a Tor.

- Correlación punto a punto.
- Pérdida de información del nodo de salida.
- Bloqueo de los nodos de salida.
- Ataque DDOS a Tor.
- HeartBleed.

- Correlación punto a punto.
- Pérdida de información del nodo de salida.
- Bloqueo de los nodos de salida.
- Ataque DDOS a Tor.
- HeartBleed.

Riffle como solución

La aparición de un nuevo paradigma de comunicación segura y anónima se debe a:

La aparición de un nuevo paradigma de comunicación segura y anónima se debe a:

- El anonimato es un derecho fundamental en las sociedades democráticas.

La aparición de un nuevo paradigma de comunicación segura y anónima se debe a:

- El anonimato es un derecho fundamental en las sociedades democráticas.
- Las Redes Tor son susceptibles ante ataques de análisis de tráfico.

La aparición de un nuevo paradigma de comunicación segura y anónima se debe a:

- El anonimato es un derecho fundamental en las sociedades democráticas.
- Las Redes Tor son susceptibles ante ataques de análisis de tráfico.
- Alternativa: Redes que ofrecen resistencia al análisis del tráfico: DC-Nets y MixNets.

La aparición de un nuevo paradigma de comunicación segura y anónima se debe a:

- El anonimato es un derecho fundamental en las sociedades democráticas.
- Las Redes Tor son susceptibles ante ataques de análisis de tráfico.
- Alternativa: Redes que ofrecen resistencia al análisis del tráfico: DC-Nets y MixNets.
- Dichas redes tienen problemas de eficiencia (Overhead de banda ancha y computacional).

Riffle: Sistema de comunicación anónima que garantiza el anonimato y minimiza los costes computacionales y de banda ancha. Prototipado por MIT en agosto de 2016. Propiedades:

Riffle: Sistema de comunicación anónima que garantiza el anonimato y minimiza los costes computacionales y de banda ancha. Prototipado por MIT en agosto de 2016. Propiedades:

- Sistema híbrido de mezcla garantizada con encriptación simétrica.

Riffle: Sistema de comunicación anónima que garantiza el anonimato y minimiza los costes computacionales y de banda ancha. Prototipado por MIT en agosto de 2016. Propiedades:

- Sistema híbrido de mezcla garantizada con encriptación simétrica.
- Recuperación de información privada.

Riffle: Sistema de comunicación anónima que garantiza el anonimato y minimiza los costes computacionales y de banda ancha. Prototipado por MIT en agosto de 2016. Propiedades:

- Sistema híbrido de mezcla garantizada con encriptación simétrica.
- Recuperación de información privada.
- Eficientes comunicaciones anónimas resistentes al análisis del tráfico de datos como a clientes maliciosos.

Riffle: Sistema de comunicación anónima que garantiza el anonimato y minimiza los costes computacionales y de banda ancha. Prototipado por MIT en agosto de 2016. Propiedades:

- Sistema híbrido de mezcla garantizada con encriptación simétrica.
- Recuperación de información privada.
- Eficientes comunicaciones anónimas resistentes al análisis del tráfico de datos como a clientes maliciosos.

Definición 1

El protocolo es correcto si, después de una ejecución satisfactoria del protocolo, todo mensaje de un cliente honesto está disponible para todos los clientes honestos.

Definición 1

El protocolo es correcto si, después de una ejecución satisfactoria del protocolo, todo mensaje de un cliente honesto está disponible para todos los clientes honestos.

Definición 2

El protocolo provee anonimato al emisor si, para toda ronda de comunicación, la probabilidad de que un adversario descubra el cliente honesto que mandó un mensaje es suficientemente cercana a $\frac{1}{k}$, siendo k el número de clientes honestos.

Propiedades de seguridad

Definición 1

El protocolo es correcto si, después de una ejecución satisfactoria del protocolo, todo mensaje de un cliente honesto está disponible para todos los clientes honestos.

Definición 2

El protocolo provee anonimato al emisor si, para toda ronda de comunicación, la probabilidad de que un adversario descubra el cliente honesto que mandó un mensaje es suficientemente cercana a $\frac{1}{k}$, siendo k el número de clientes honestos.

Definición 3

El protocolo provee anonimato al receptor si, para toda ronda de comunicación, la probabilidad de que un adversario descubra el cliente honesto que mandó un mensaje es suficientemente cercana a $\frac{1}{n}$, siendo n el número de mensajes disponibles.

Arquitectura Riffle

Mezcla Híbrida Comprobable. Algoritmo

1. **Compartir claves:** Un probador P y un verificador V generan parejas de claves públicas-privadas (s_P, p_P) y (s_V, p_V) y hacen públicas las claves p_P y p_V . Un cliente C comparte sus claves $\{k'_i : i \in [n]\}$ con P .
2. **Permutación de las claves:** C genera $\{k_i : i \in [n]\}$ para V y manda $\{Enc_{p_P}(Enc_{p_V}(k_j))\}$ a P y V . P descripta y mezcla de forma verificada usando una permutación aleatoria π , y manda $\{Enc_{p_V}(k_{\pi_j})\}$ a V . V verifica la mezcla y la descriptación, y descripta para ver $\{k_{\pi_j}\}$.
3. **Envío de mensajes:** Para $r = 1, \dots, R$,
 - 3.1 Mezcla: Para mandar mensajes $\{M_j^r\}_{j \in [n]}$ C encripta los mensajes en capas (onion-encrypt) y envía $\{AEnc_{k'_j, r}(AEnc_{k_j, r}(M_j^r))\}_{j \in [n]}$ a P , donde $AEnc$ es un esquema autenticado encriptado que usa r como nonce. P descripta una capa de la encriptación usando $\{k'_j\}_{j \in [n]}$, los permuta usando la misma π y manda $\{AEnc_{k_{\pi(j)}, r}(M_{\pi(j)}^r)\}_{j \in [n]}$ a V .
 - 3.2 Verificación: V verifica el texto cifrado comprobando la autenticación a través de las claves $\{k_{\pi(j)}\}_{j \in [n]}$ y r , descripta una capa y adquiere los mensajes $\{M_{\pi(j)}^r\}_{j \in [n]}$.

Recuperación de información privada. Algoritmo

1. Configuración. Cada cliente C_j comparte dos secretos $m_{i,j}$ y $s_{i,j}$ con cada servidor S_i excepto con su servidor principal S_{p_j} . Esto pasa una vez por época.
2. Descarga:
 - 2.1 Generación de máscaras. Sea l_j el índice del mensaje que C_j quiere descargar. C_j genera $m_{p_j,j}$ de forma que $\oplus_i m_{i,j} = e_{l_j}$, donde e_{l_j} es una máscara de bits con un 1 en el índice l_j . C_j manda $m_{p_j,j}$ a S_{p_j} .
 - 2.2 Generación de la respuesta. Cada servidor S_i computa la respuesta $r_{i,j}$ para C_j calculando la XOR de los mensajes en las posiciones donde hay 1's en $m_{i,j}$, y haciendo la XOR a los $s_{i,j}$. Específicamente, $r_{i,j} = (\oplus_l m_{i,j}(l) \wedge M_l) \oplus s_{i,j}$, donde M_l es el mensaje en este plano número l . Entonces, los servidores mandan $r_{i,j}$ a S_{p_j} y éste calcula r_j :
$$r_j = \oplus_i r_{i,j} = (\oplus_i \oplus_l m_{i,j}[l] M_l) \oplus (\oplus_i s_{i,j}) = M_{l_j} \oplus (\oplus_i s_{i,j}).$$
 - 2.3 Descarga del mensaje. C_j descarga r_j de S_{p_j} y hace la XOR de todos los $\{s_{i,j}\}_{i \in [m]}$ para obtener el mensaje deseado, $M_{l_j} = r_j \oplus (\oplus_i s_{i,j})$.
 - 2.4 Actualización de secretos. C y S aplican PRNG a sus máscaras y secretos para refrescarlos.

- Configuración.
 1. Mezcla de las claves:
 - 1.1 Cada servidor S_i genera pares de claves públicas-privada y facilita la pública p_i a los clientes. Además, genera la permutación π_i .
 - 1.2 Cada cliente C_j genera las claves $k_{i,j}$ para S_i y las encripta con las claves p_1, \dots, p_i para $i = 1, \dots, m$. C_j entrega $m \{k_{i,j}\}$ encriptadas por capas a S_1 a través del servidor principal.
 - 1.3 Desde S_1 a S_m , S_i desencripta las claves. S_i las guarda, realiza una mezcla verificada de las otras claves usando π_i y las envía mezcladas a S_{i+1} . Los servidores verifican la desencriptación y mezclan.
 2. Compartir secretos: Cada pareja S_i, C_j genera una pareja de secretos $m_{i,j}, s_{i,j}$ utilizados en PIR (Algoritmo 2), en la etapa de descarga.

- Comunicación. En la ronda r ,
 1. Subida de datos. C_j encripta por capas el mensaje M_j^r usando encriptación autenticada con $\{k_{i,j}\}$ y r como un nonce:
 $AEnc_{1,\dots,m}(M_j^r) = AEnc_{k_{i,j},r}(\dots(AEnc_{k_{mj},r}(M_j^r))\dots)$. C_j entonces envía $AEnc_{1,\dots,m}(M_j^r)$ a S_1 a través de su servidor principal.
 2. Mezcla. Desde S_1 hasta $S - m$, S_i autentica, desencripta y mezcla los textos cifrados usando π_i , y envía mezclado $\{AEnc_{i+1,\dots,m}(M_j^r)\}_{j \in [n]}$ a S_{i+1} . S_m comparte los mensajes finales en texto plano con todos los servidores.
 3. Descarga. Los clientes descargan los mensajes en texto plano utilizando PIR.

- Corrección

- Corrección
- Anonimato del emisor

- Corrección
- Anonimato del emisor
- Anonimato del receptor

- Petición de bloques

- Petición de bloques
- Subida de bloques

- Petición de bloques
- Subida de bloques
- Descarga de bloques

Demo de ARM

¿Qué es ARM?

¿Qué es ARM?

ARM

ARM es un monitor del estado de un nodo de la red.

¿Qué es ARM?

ARM

ARM es un monitor del estado de un nodo de la red.

Nosotros lo hemos utilizado para ver la evolución de nuestro nodo.

Datos que nos proporciona ARM

Datos que nos proporciona ARM

ARM nos proporciona datos como:

Datos que nos proporciona ARM

ARM nos proporciona datos como:

- Flags del nodo.

ARM nos proporciona datos como:

- Flags del nodo.
- Consumo de recursos locales.

ARM nos proporciona datos como:

- Flags del nodo.
- Consumo de recursos locales.
- Gráficas del tráfico de red producido por Tor.

ARM nos proporciona datos como:

- Flags del nodo.
- Consumo de recursos locales.
- Gráficas del tráfico de red producido por Tor.

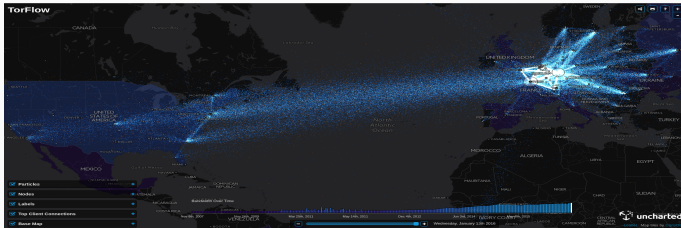
Demo con ARM



Otras webs de monitorización de la red

[illegible]

Tor Status



Tor Flow

Demo Tor Browser

¿Qué es el Tor Browser?

¿Qué es el Tor Browser?

Tor Browser

Es una modificación del navegador Mozilla Firefox de código libre que configura automáticamente el acceso del usuario a la red Tor.

¿Qué es el Tor Browser?

Tor Browser

Es una modificación del navegador Mozilla Firefox de código libre que configura automáticamente el acceso del usuario a la red Tor.

Ventajas del Tor Browser

Gracias a este navegador los usuarios sin conocimiento acerca de la red pueden utilizarla. Distribuciones como Tails traen este navegador por defecto.

Conclusiones

El anonimato en la red es algo importante y debemos ser conscientes de ello.

El anonimato en la red es algo importante y debemos ser conscientes de ello.

'Argumentar que no te importa el derecho a la privacidad porque no tienes nada que esconder es como decir que no te importa la libertad de expresión porque no tienes nada que decir.'

Edward Snowden

¿Preguntas?