

Servizi Onion

Dalla teoria all'implementazione

Leonardo Migliorelli¹
leonardo.migliorelli@studenti.unicam.it



COMPUTER SCIENCE @ UNICAM

Tesi Unicam - Servizi Onion, dalla teoria dall'implementazione

- ▶ Relatore: Marcantoni Fausto
<https://computerscience.unicam.it/marcantoni/>
- ▶ Studente: Leonardo Migliorelli Mat.113920
leonardo.migliorelli@studenti.unicam.it

INDICE

Introduzione

Onion

Creazione del circuito

Chaum Mix

Tor § Onion v2

Network Design

Implementazione

Pubblicizzare il servizio

INTRODUZIONE

In questa tesi spiegheremo il funzionamento delle reti **Onion** e parleremo in particolare della rete **Tor**, in fine mostreremo com'è possibile implementare un **servizio Onion/Tor**.

Le reti Onion sono state create per risolvere le due più grandi vulnerabilità di Internet che gravano sulla **privacy** e **sull'anonimato**, ovvero l'**analisi del traffico** e le **intercettazioni**. Una rete di questo tipo nasconde infatti gli indirizzi e il contenuto di ogni richiesta.

ONION



La rete Onion è una rete distribuita composta da nodi chiamati **Onion Router**, collegati tra loro tramite i circuiti creati dagli **Onion Proxy**.

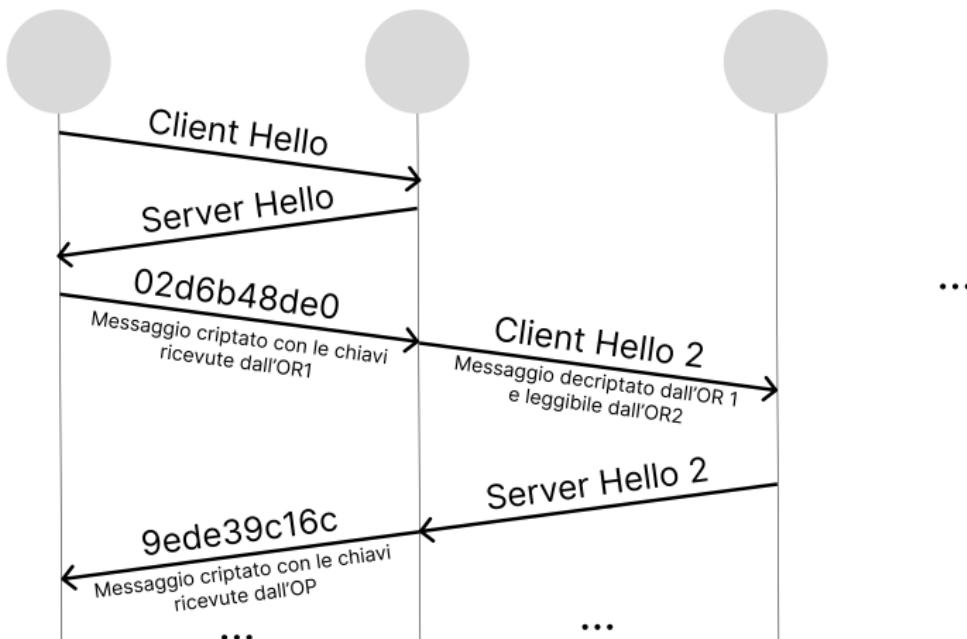
Ogni pacchetto che passa nel circuito viene decriptato in maniera sequenziale dai relativi nodi fino ad arrivare all'exit node che si occupa di instradare il pacchetto nella rete Internet. Grazie a questo meccanismo nessun nodo conosce contemporaneamente l'indirizzo del mittente e del destinatario.

CREAZIONE DEL CIRCUITO

La generazione del circuito è un processo **iterativo** e **progressivo** in cui il proxy sceglie i nodi del circuito. A partire dal primo nodo viene instaurata una connessione **TLS** e vengono scambiate le **chiavi simmetriche** tramite un processo **asimmetrico** (in maniera simile allo scambio di chiavi di HTTPS).

Questo processo continua fino all'exit node, a questo punto il circuito è completo e può iniziare a trasmettere stream dati.

Onion Proxy Onion Router 1 Onion Router 2



Il proxy onion è composto da tre componenti:

- ▶ Un proxy che genera e gestisce le connessioni, ha necessità di conoscere la topologia della rete Onion.
- ▶ **Application Specific Proxy**, si occupa di convertire le richieste delle applicazioni in pacchetti Onion.
- ▶ **Application Specific Privacy Filter**, un proxy opzionale che sanifica lo stream di dati rimuovendo informazioni che potrebbero identificare il client.

CHAUM MIX

La rete Onion è basata sullo studio di David Chaum che propose un sistema di comunicazione anonima basato sulla crittografia. Nella sua conclusione una rete di questo tipo doveva avere le seguenti caratteristiche:

- ▶ **Sealing**, una tecnica con cui il messaggio viene annesso ad una stringa casuale prima di essere criptato per aumentarne la sicurezza.
- ▶ **Indirizzo non tracciabile**, un indirizzo generato dal client criptando quello reale, rendendone possibile la decriptazione solo dal primo nodo. Viene usato come indirizzo di risposta dal destinatario.

TOR

La rete Tor è la più famosa implementazione di Onion, doveva essere semplice da usare, così da incrementare il numero di nodi e semplificare l'anonimizzazione.

Le principali migliorie rispetto alla rete Onion sono:



- ▶ Circuiti telescopici, il pacchetto viene criptato anche con la chiave di sessione oltre che con quella pubblica del nodo
- ▶ Utilizzo del protocollo SOCKS
- ▶ Controllo di congestione, consiste in due finestre che tengono traccia del numero di celle che possono transitare per un circuito
- ▶ Directory server

DIRECTORY SERVERS

I Directory Server sono un sottogruppo di onion router che tracciano i cambiamenti nella **topologia di rete** e agiscono da **DNS server** per i servizi onion. Mantengono infatti i **descriptor**, pacchetti generati dai servizi onion criptati con la propria chiave privata, che contengono gli introduction points e la chiave pubblica.

Il sistema di generazione di indirizzi onion fornisce un **meccanismo di sicurezza** per evitare che un malintenzionato possa alterare i descriptor e reindirizzare gli utenti ai propri introduction points, infatti se così fosse la chiave pubblica nascosta nell'indirizzo onion non sarebbe in grado di decifrare il descriptor.

NETWORK DESIGN

La

rete Tor è definita una **overlay network** ovvero
una rete che esiste al di sopra delle reti esistenti.

I pacchetti che viaggiano
nella rete Tor sono chiamati **celle**, hanno una
dimensione fissa 512 bytes, e sono di due tipi:

- ▶ **Control**, gestisce il circuito
- ▶ **Relay**, trasporta stream dati

Applicazioni

Rete TOR

Trasporto (TCP)

Internet

Network Access

TOR RELAY

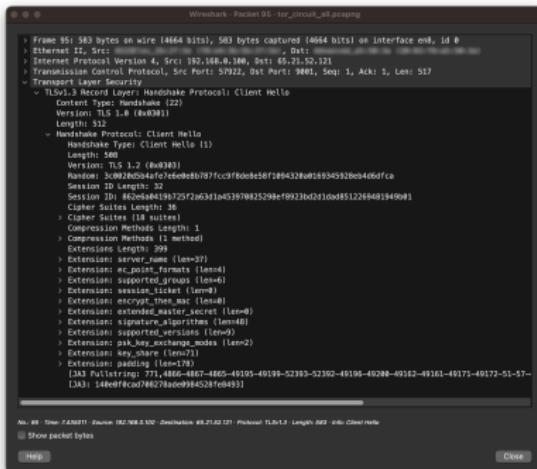
La rete TOR si basa su un insieme di nodi gestiti da volontari chiamati TOR Relay, possono essere di tre tipi:

- ▶ **Non-exit Relay**, i nodi interni della rete che a loro volta si dividono in:
 - ▶ **Guard Relay**, il primo nodo del circuito
 - ▶ **Middle Relay**, i nodi intermedi del circuito
- ▶ **Exit Relay**, i nodi di uscita della rete Tor, instradano il traffico nella rete comune. Essendo gli unici IP visibili all'esterno sono i più esposti a rischi legali.
- ▶ **Bridge Relay**, nodi non pubblici che non possono quindi essere bloccati

Tutte le informazioni sui Relay esistenti sono visitabili al seguente indirizzo:

<https://metrics.torproject.org/rs.html>

DIMOSTRAZIONE WIRESHARK



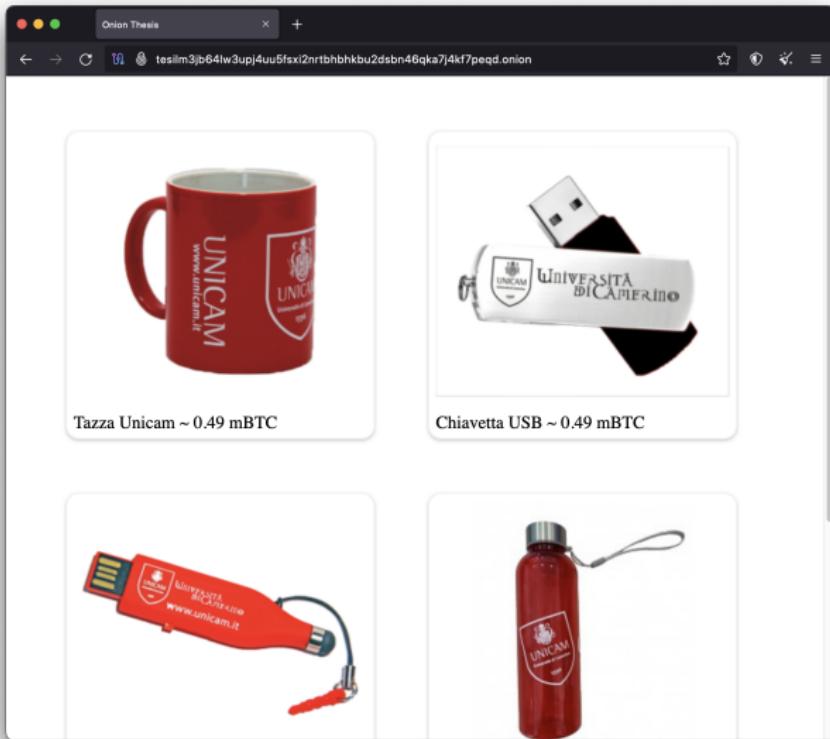
IMPLEMENTAZIONE

Per implementare un servizio onion è necessario utilizzare un **Proxy Onion** per inoltrare le richieste dalla rete Tor al server web. Il proxy gestisce la generazione delle chiavi, dell'indirizzo e la definizione e connessione con gli **introduction points**, oltre alla generazione del descriptor e la sua pubblicazione nei Directory Servers.

In particolare la nostra implementazione **Onion V3** (dato che Onion V2 è stato deprecato) userà un server **Linux** su AWS per ospitare il Proxy Onion e un **server nginx**.

Nel sistema è necessario installare il web server NGINX e dopo una serie di configurazioni dei repository possiamo installare il Proxy Tor. Nel file di configurazione torrc è possibile indicare la directory in cui salvare le chiavi e il tipo di connessione con il web server.

In questa implementazione abbiamo usato un tool (mkp224o) per generare la coppia chiave privata e pubblica che risulti in un indirizzo onion personalizzato.



PUBBLICIZZARE IL SERVIZIO

Nel lavoro di tesi abbiamo mostrato come pubblicizzare il servizio onion tramite un'Header Tag, è infatti possibile inserire un tag all'interno della pagina html che quando aperto con Tor consente di reindirizzare l'utente al relativo sito onion.

```
<meta http-equiv="onion-location" content="http://tesilm3jb64lw3upj4uu5fsxi2nrtbhbhkbu2dsbn46qka7j4kf7peqd.onion" />
```

The screenshot shows a web browser window with the title "Onion Thesis". The address bar displays the URL "https://tesi.miglio.dev". A purple status bar at the top right indicates ".onion available". The main content area contains four items listed in a grid:

- Tazza Unicam ~ 0.49 mBTC**: A red ceramic mug with the "UNICAM" logo and website "www.unicam.it".
- Chiavetta USB ~ 0.49 mBTC**: A white USB flash drive with the "UNICAM" logo and website "www.unicam.it".
- Chiavetta USB rossa ~ 0.49 mBTC**: A red USB flash drive with the "UNICAM" logo and website "www.unicam.it".
- Borraccia Unicam rossa ~ 0.283 mBTC**: A red water bottle with the "UNICAM" logo and website "www.unicam.it".

ACKNOWLEDGEMENTS

