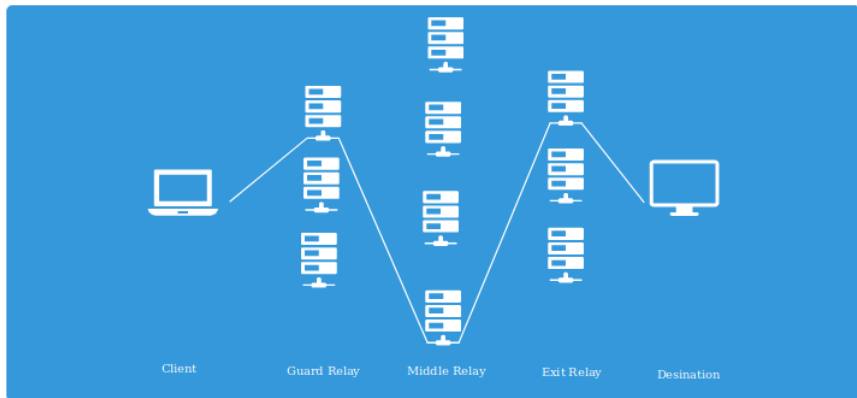


The Onion Router (Tor)

Cornel Bucurescu, Adrian Manea, Cristian Nicolae

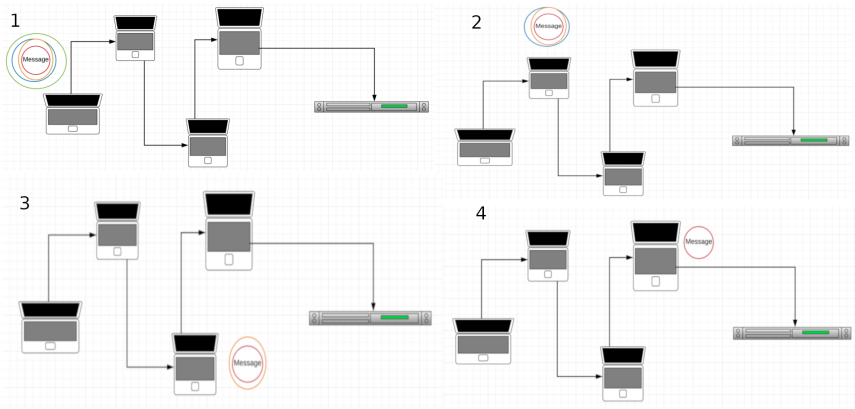
SLA, 406 & 410

Ideea principală (TL;DR)



Ilustrație: Cele 3 relee dintr-o conexiune standard [Wright, 2015]

Ideea principală (TL;DR)



Ilustrație: Criptarea telescopică [Skerritt, 2018]

Obiective de design (2004)

- perfect forward secrecy;
- separarea filtrării de anonimitate;
- TCP streams multiplexing;
- topologie “leaky pipe”;
- autorități directoare;
- servicii ascunse.

Non-objective (limitări admise)

- nu servește drept conexiune peer-to-peer (vulnerabilă!);
- nu protejează împotriva atacurilor end-to-end (timing, intersection);
- nu normalizează protocoalele;
- nu ascunde traficul (\neq *steganografic*).

Modelul este de “overlay network”.

OR au chei de identitate (IDK) și cheie onion (OK).

IDK semnează certificate TLS și directoarele.

OK decriptează cererile, stabilește circuite și negociază cheile efemere.

Comunicarea se face prin TLS cu chei efemere.

Traficul circulă în *celule de câte 512 bytes*, cu *header* și *payload*.

Construcția unui circuit Alice \longrightarrow Bob

Alice $\xrightarrow{\text{create}}$ Bob și alege circID C_{AB} .

Alice $\xrightarrow{OK=g^x}$ Bob.

Bob $\xrightarrow{\text{created}=g^y, \text{ hash}(K=g^{xy})}$ Alice.

Acum, Alice și Bob pot comunica folosind cheia K .

Extinderea circuitului Alice \rightarrow Bob \rightarrow Carol

A $\xrightarrow{\text{relay extend, } g^{x_2}, \text{ addr}(C)}$ Bob.

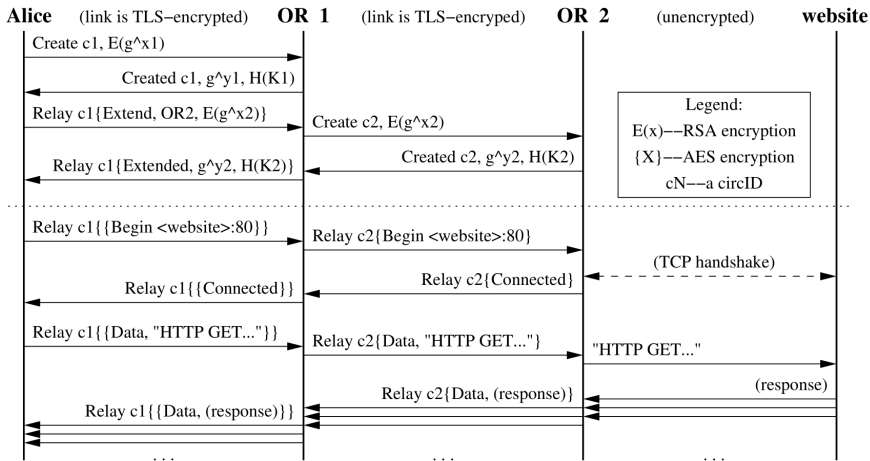
Bob $\xrightarrow{\text{create}}$ Carol && copy(g^{x_2}) && alege circID C_{BC} .

Carol $\xrightarrow{\text{created}=g^{y_2}}$ Bob $\xrightarrow{\text{payload}}$ Carol.

Bob $\xrightarrow{\text{relay extended}}$ Alice.

Acum, se poate comunica A—C cu cheia $K_2 = g^{x_2 y_2}$.

Circuit cu 3 rele



Ilustrație: Celule schimbate la inițierea unui circuit ([Dingledine et al., 2004], §4.1

Servicii ascunse ([hash].onion)

Serverul ascuns creează o pereche de chei pentru criptare asimetrică și cere unui OR să devină punct de intrare.

După acceptare, serverul pune într-un *tabel de hash-uri distribuit* IP-urile punctelor de introducere acceptate.

Clientul primește adresa de tip [hash].onion (cheia publică), deci știe IP-urile cu care poate intra în Tor.

Clientul creează o celulă de introducere, cu adresa nodului ales pentru acces.

Nodul decriptează celula de introducere cu cheia privată și acceptă clientul.

Comunicarea se face cu un nod intermediar, deci conexiunea cu serverul ascuns are cel puțin 6 noduri.

Autorități directoare (AD)

Există cîteva autorități directoare = noduri de încredere.

AD votează și actualizează statusul rețelei în *consens*:

- fac o listă de relee cunoscute;
- calculează alți parametri necesari (țara, lățimea de bandă);
- transmit cele de mai sus către celelalte AD, ca status;
- rezultă un „status mediat“;
- din status \Rightarrow vot, transmis între AD cu semnătură.

Votul și actualizarea sînt publice.

Atacuri × apărări

Passive:

- Observarea tiparelor de trafic × TCP streams multiplexing;
- End-to-end correlation × leaky pipe topology;
- Confirmation attacks, website fingerprinting × celule mici, padding.

Active:

- OR impersonation × chei efemere;
- Legal backdoors, probleme politice sau sociale × ???;
- Cenzură la ieşire × ???;
- Cenzură la intrare × bridges.



Dingledine, R., Mathewson, N., and Sylverson, P. (2004).

Tor: The second generation onion router.

SSYM Proceedings of the 13th conference on USENIX Security Symposium, 13.



Skerritt, B. (2018).

How Tor *really* works.

HackerNoon.



Tor (2019).

The Tor Project.

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



Wright, J. (2015).

How Tor Works: Part One.

blog post.