Titre du document

Nom de l'auteur

12 mai 2024

1 Abstract

Tor :
— Serice de communication anonyme
— À faible latence
— Basé sur un circuit
— protocole pour des routeurs en oignon asynchrones
— vaguement fédéré
Plus par rapport à la première version :
— secret de forward parfait
— contrôle de congestion
— serveur d'annuaire
— contrôle d'intégrité
— politiques de sortie confugurables
— points de rendez-vous : conception pratique pour les services cachés
— fonctionne sur l'internet reel
— pas de privilèges spéciaux
— pas de modifications du noyau
— peu de synchro ou de coordination entre les noeuds
— compromis entre anonymat, facilité d'utilisation et efficacité
ici => réseau de + de 30 noeuds

2 Overview

Routage onion : réseau distribué superposé conçu pour rendre anonymes les applications basées sur le protocole TCP telles que :
— Navigation sur le web : Tor browser
— Secure shell : ssh
— Messagerie instantannée

Comment ? Les clients choisissent un chemin à travers le réseau et construisent un circuit dans lequel chaque noeud ("routeur onion" ou "OR") du chemin ne connaît que son prédécesseur ainsi que son successeur.

Le traffic (paquets de données) étant propagé en cellules de tailles fixes (pour conserver l'anonymat : aucune infos sur l'expéditeur). Chaque cellule est cryptée en couches (une couche par noeud où transitent les données), chaque routeur déchiffre une couche via clé symétrique avant de transférer les données au routeur suivant.

2.1 OR 1 vs OR 2

2.1.1 Secret parfait

Avant : structure de données unique à chiffrement multiple Maintenant : Conception de construction de chemin téléscopique

Avant : un noeud hostile pouvant capturer le traffic et ainsi rompre l'anonymat Maintennt : impossible de connaître l'historique du traffic

Avant : Maintenant : Diffie-Hellman pour la connexion avec le premier noeud, celle entre le premiere et le deuxième,..

2.1.2 Séparation du nettoyage du protocole de l'anonymisation

Usage de privoxy plutot de que services spécifiques à Tor avant : proxy d'application Maintenant : SOCKS et privoxy

2.1.3 Partage d'un circuit par plusieurs flux TCP

Avant : un circuit par request d'application TCP Maintenant : un circuit multiplexant toutes les requetes d'application TCP

2.1.4 Topologie de circuit à fuite

Possibilité de rediriger le trafic vers d'autres noeuds pour contrer des attaques

2.1.5 Contrôle de la congestion

contrôle de congestion décentralisée par les noeuds périphériques

2.1.6 Serveur d'annuaire

Avant : inoder le réseau d'informations Maintenant : serveurs d'annuaire sur les noeuds les plus sûrs (routeurs + états)

2.1.7 Politiques de sortie variables

1 noeud = 1 politique d'hôtes et d'inferfaces auxquelles se connecter

2.1.8 Contrôle d'intégrité de bout en bout

Maintenant : Tor vérifie l'intégrité des données avant qu'elles ne quittent le réseau pour empêcher leur altération par un noeud.

2.1.9 Points de rendez-vous et services cachés

Avant : onions de réponses à longue durée de vies Maintenant : points de rendez-vous pour se connecter à des serveurs cachés

3

4 Desing

Connexions TLS maintenue entre chaque OR et chaque autre OR

Chaque OR execute un OP pour récupérer les annuaires/ répertoires, et établir des circuits de réseaux

Chaque OR a une clé d'identité long terme et une clé onion court terme :

— La clé identité est utile pour :
  — Signer les certificats TLS
  — Signer la description du OR : its keys, address, bandwidth, exit policy, and so on
  — Signer les annuaires (via serveurs d'annuaires)
— La clé onion est utilisée pour décrypter les requêtes d'utilisateurs poue mettre en place un circuit et négacier les clés éphémères Le protocole TLS établit aussi un lien de clé court terme lors des communications entre OR Les clés court terme sont rotatées périodiquement et indépendamment pour limiter l'impact sur la compromission des clés

> TLS, abréviation de Transport Layer Security, est un protocole de sécurité conçu pour fournir des communications sécurisées sur un réseau informatique. Il est largement utilisé sur Internet pour sécuriser les échanges de données entre un site web et un navigateur, garantissant ainsi que les données transmises, telles que les détails de carte de crédit et les informations personnelles, restent confidentielles et à l'abri des interceptions malveillantes.
>
> Tor offre un anonymat en faisant transiter les communications à travers un réseau de serveurs, appelés nœuds ou routeurs oignon, empêchant ainsi quiconque d'observer qui communique avec qui ou de surveiller les sites que les utilisateurs visitent. Tor utilise le chiffrement dans sa structure en couches (d'où l'analogie de l'oignon) pour assurer la confidentialité et l'intégrité des données à chaque étape de leur transit à travers le réseau.
>
> En revanche, TLS fonctionne en établissant un canal sécurisé entre deux parties (par exemple, un site web et un navigateur) pour la communication sécurisée, en utilisant des certificats numériques pour authentifier l'identité du serveur. Une fois la connexion sécurisée établie, toutes les données transmises sont cryptées, ce qui rend difficile pour les tiers d'intercepter ou de modifier les informations.
>
> Bien que Tor et TLS servent tous deux à améliorer la sécurité et la confidentialité des communications sur Internet, ils opèrent à différents niveaux et pour des objectifs légèrement différents. Tor se concentre sur l'anonymat et la protection contre la surveillance réseau, tandis que TLS se concentre sur la sécurisation des communications point à point entre le client et le serveur.

— *Citation fournie par ChatGPT*

4.1 cellules

5 Clés

6 Conclusion