# Quel est log ouvert une session d'accès pour le trafic HTTPS ?

## Contenu



Contribué par Kei Ozaki et Siddharth Rajpathak, ingénieurs TAC Cisco.

## Question :

Quel est log ouvert une session d'accès pour le trafic HTTPS ?

**Environnement** : Appliance de sécurité Web de Cisco (WSA) exécutant des versions 7.1.x et ultérieures d'AsyncOS, proxy HTTPS activé

La manière que l'appliance de sécurité Web de Cisco (WSA) se connecte le trafic HTTPS est différente comparée au trafic http normal.  Les entrées HTTPS enregistrées dans les accesslogs sembleront différentes selon la façon dont la demande a été traitée. En général il a différentes caractéristiques comparées au trafic http normal.

Ce qui est enregistré dépendra de quel mode de déploiement vous utilisez (mode en avant explicite ou mode transparent).

Permettez-d'abord nous regardent quelques mots clé qui vous aideraient des logs d'accès en lecture facilement.

**TCP_CONNECT** - ceci affiche que le trafic a été reçu d'une manière transparente (par l'intermédiaire du WCCP ou du L4 réorientez… etc.)
**CONNECTEZ** - ceci affiche que le trafic a été reçu explicitement
**DECRYPT_WBRS** - ceci affiche que WSA a décidé de déchiffrer le trafic dû au score WBRS
**PASSTHRU_WBRS** - ceci affiche que WSA a décidé de traverser le trafic dû au score WBRS
**DROP_WBRS** - ceci affiche que WSA a décidé de relâcher le trafic dû au score WBRS

- Quand le trafic **HTTPS** est déchiffré, WSA se connectera deux entrées.
- **TCP_CONNECT** ou **SE CONNECTENT** selon le type de demande étant reçue et **« OBTENEZ https:// »** affichant l'URL déchiffré.
- Le plein **URL** sera seulement visible si WSA déchiffre le trafic.

Veuillez noter également cela :

- En mode transparent, WSA verra seulement l'adresse IP de destination au commencement
- En mode explicite, WSA verra l'adresse Internet de destination

Sont ci-dessous quelques exemples de ce que vous verriez dans les accesslogs :

| Transparent - Déchiffrage |
| --- |
| 1252543170.769 386 192.168.30.103 TCP_MISS_SSL/200 0 TCP_CONNECT tunnel://192.168.34.32:443/ - DIRECT/192.168.34.32 - DECRYPT_WBRS-DefaultGroup-test.id-NONE-NONE-DefaultRouting <Sear,5.0,-,-,-,-,-,-,-,-,-,-,-,-> -<br><br>1252543171.166 395 192.168.30.103 TCP_MISS_SSL/200 2061 OBTIENNENT https://www.example.com:443/sample.gif - DIRECT/192.168.34.32 l'image/GIF DEFAULT_CASE-test.policy-test.id-NONE-NONE-NONE <Sear,5.0,0,-,-,-,-,0,-,-,-,-,-,-,-> - |
| Connexion transparente |
| 1252543337.373 690 192.168.30.103 TCP_MISS/200 2044 TCP_CONNECT tunnel://192.168.34.32:443/ - DIRECT/192.168.34.32 - PASSTHRU_WBRS-DefaultGroup-test.id-NONE-NONE-DefaultRouting <Sear,9.0,-,-,-,-,-,-,-,-,-,-,-,-> - |
| Transparent - Baisse |
| 1252543418.175 430 192.168.30.103 TCP_DENIED/403 0 TCP_CONNECT tunnel://192.168.34.32:443/ - DIRECT/192.168.34.32 - DROP_WBRS-DefaultGroup-test.id-NONE-NONE-DefaultRouting <Sear,-9.1.0,-,-,-,-,-,-,-,-,-,-,-,-> - |
| Explicite - Déchiffrage |
| 252543558.405 385 10.66.71.105 TCP_CLIENT_REFRESH_MISS_SSL/200 40 CONNECTENT tunnel://www.example.com:443/ - www.example.com DIRECT - DECRYPT_WBRS-DefaultGroup-test.id-NONE-NONE-DefaultRouting <Sear,5.0,-,-,-,-,-,-,-,-,-,-,-,-> -<br>1252543559.535 1127 10.66.71.105 TCP_MISS_SSL/200 2061 OBTIENNENT https://www.example.com:443/sample.gif - l'image DIRECTE de www.example.com/GIF DEFAULT_CASE-test.policy-test.id-NONE-NONE-NONE <Sear,5.0,0,-,-,-,-,0,-,-,-,-,-,-,-> - |
| Explicite - Traversez |
| 1252543491.302 568 10.66.71.105 TCP_CLIENT_REFRESH_MISS/200 2256 CONNECTENT tunnel://www.example.com:443/ - www.example.com DIRECT - PASSTHRU_WBRS-DefaultGroup-test.id-NONE-NONE-DefaultRouting <Sear,9.0,-,-,-,-,-,-,-,-,-,-,-,-> - |
| Explicite - Baisse |
| 1252543668.375 1 10.66.71.105 TCP_DENIED/403 1578 CONNECTENT tunnel://www.example.com:443/ - AUCUNS - DROP_WBRS-DefaultGroup-test.id-NONE-NONE-NONE <Sear,-9.1,-,-,-,-,-,-,-,-,-,-,-,-> - |