Diagramme de séquence de l'établissement d'une session SSH

<uml> @startuml group Etablissement d'une session 3-Way Handshake ClientSSH → Serveur : demande session avec paquet SYN (Synchronize) Serveur → ClientSSH : approuve connexion avec paquet SYN-ACK (acknowledgement) ClientSSH → Serveur : confirme avec paquet ACK end group Negociation des algorithmes de chghffrement ClientSSH → ClientSSH : annonce la version de SSH utilisée Serveur → ClientSSH : annonce la version de SSH utilisée ClientSSH → Serveur : Key Exchange Init (liste des algorithmes supportés) Serveur → ClientSSH : Key Exchange Init (liste des algorithmes supportés)

Serveur → ClientSSH : certificat ClientSSH → ClientSSH : vérification certificat & empreinte end group Partage clé secrète ClientSSH → Serveur : liste d'algorithmes de chiffrement Serveur → ClientSSH : liste d'algorithmes de chiffrement ClientSSH → ClientSSH : choix algorithme Serveur → Serveur : choix algorithme ClientSSH → ClientSSH : génèrent des paires de clés \n publiques-privées temporaires ClientSSH → ClientSSH : génére cle de session end @enduml </uml>

Compléments

Les connexions TCP

L'agorithme de chiffrement symétrique **ChaCha20** est utilisé à la palec de AES 256 pêyt ChaCha20 : algorithme de chiffrement symétrique plus rapide qu'AES sur un matériel générique (mise en œuvre purement en logiciel) Poly1305 : c'est un MAC (message authentication code) permet d'assurer l'intégrité des données en vérifiant qu'elles n'ont subi aucune modification après une transmission = semblable aux fonctions de hachage. La combinaison des deux permet de faire du chiffrement intègre

Ressources

- https://www.bortzmeyer.org/7539.html
- https://fr.wikipedia.org/wiki/Code_d%27authentification_de_message
- https://serverfault.com/questions/586638/understand-wireshark-capture-for-ssh-key-exchange