

Rust Remote.Access.Trojan

Antoine MARTIN, Wesley EDE
Amad MOHAMMAD, Denis REMACLE

April 26, 2022

Sommaire

Qu'est-ce qu'un Remote.Access.Trojan ?

Pourquoi un R.A.T ?

Mais pourquoi en RUST absolument ?

Comment fonctionne-t-il en somme ?

Etat des avancements

PoC de notre solution

PoC de notre solution (Démonstration)

Batterie de fonctionnalités restant à implémenter

Qu'est-ce qu'un Remote.Access.Trojan ?

- ▶ Un R.A.T est un logiciel qui n'est pas forcément malveillant et qui permet la prise de contrôle à distance d'un PC
- ▶ Dans notre cas c'est un malware qui permet de prendre contrôle à distance et exécuter des commandes sur un poste ou un ensemble de postes infecté(s).
- ▶ Exemples notables : DarkComet, NanoCore, NJRat...

Pourquoi un R.A.T ?

- ▶ Un challenge stimulant et enrichissant
- ▶ Choix cohérent avec les compétences diverses du groupe
- ▶ Une occasion d'apprendre un langage dont l'importance ne fait que croître

Mais pourquoi en RUST absolument ?

- ▶ Un langage permettant un code "sur" orienté bas niveau
- ▶ Un langage qui prends sans cesse de l'importance de part son utilisation : noyau linux, moteur HTML de firefox, etc.
- ▶ Une communauté grandissante et active

Comment fonctionne-t-il en somme ?

- ▶ Kptain-Ratz est capable pour l'instant :
- ▶ D'utiliser le port 53 en UDP pour se camoufler parmi les flux DNS
- ▶ D'envoyer un heartbeat a intervalle aléatoire allant de 30 min à 1 heure, le serveur est capable de l'interpréter et d'envoyer les instructions dans la réponse au heartbeat
- ▶ Il est codé en RUST

Etat des avancements

- ▶ Mise en revue du GAANT associé au Projet
- ▶ Différentes difficultés rencontrées

Nom de la tâche ▼	Durée ▼
Projet_annuel	56 jours
Création de la base	21 jours
Configuration du client	14 jours
Configuration du serveur	14 jours
Interface utilisateur	7 jours
Création des différents Plug-In	42 jours
Chiffrement des connexion	7 jours
Keylogger	7 jours
Gestion de fichier	14 jours
Gestionnaire de tache	7 jours
Envoi de message	14 jours
exécution de programme	7 jours
screenshot	7 jours
écoute du micro	7 jours
récupération des mots de passe navigateur	7 jours
Shell distant	7 jours
mode persistant	14 jours
Correction de bug	7 jours

PoC de notre solution

- ▶ Voici les étapes qui constituent l'initiation d'une connexion entre le client et le serveur :
- ▶ Chiffrement de la connexion
- ▶ Envoi des instructions
- ▶ Réception et interprétation du Heartbeat envoyé par le client



Batterie de fonctionnalités restant à implémenter

- ▶ Une interface graphique, des fonctionnalités diverses :
- ▶ Keylogger, Remote Desktop, SCP etc.
- ▶ L'interface cible devra être semblable à celle-ci :

