Projet d'Interconnexion

Rapport

Nouhailla ACHEBOUNE – Othmane CHAOUCHAOU – Amine JAAFARI

Enzo PETIT – Rokia SBAI – Nam VU – Julien WUSZKO

Département Sciences du Numérique - 2e année ASR

16 janvier 2022

Table des matières

1 Introduction

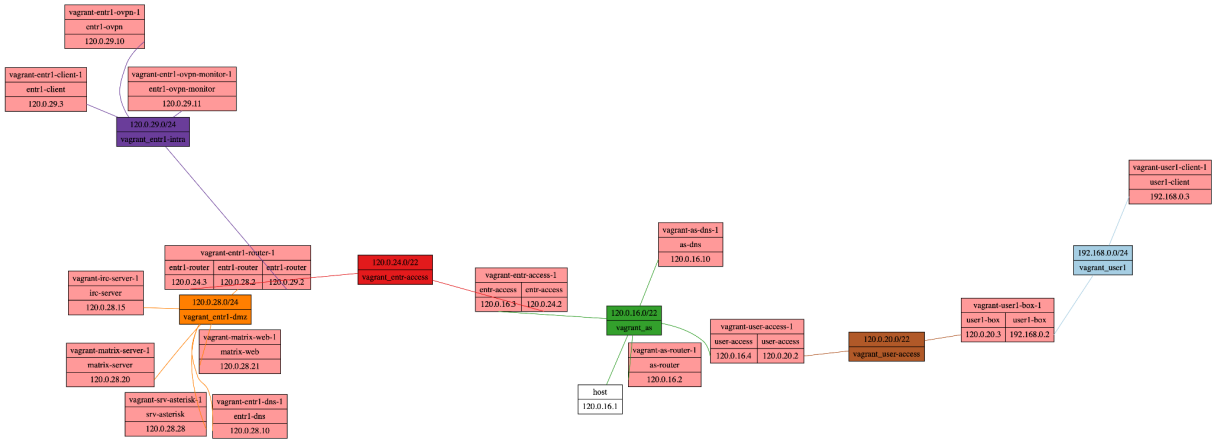
Le projet de réseau “interconnexion” de 2ème année nous propose de mettre en place un petit réseau d’entreprise, ainsi que des services et le réseau opérateur associé. Nous avons choisi de le déployer sous Docker afin que tous les membres de l’équipe puissent avoir la main sur toutes les configurations et toutes les machines d’un coup en local (à condition d’avoir une connexion internet pour se mettre à jour).

2 Architecture générale

Nous avons fait le choix d’une virtualisation complète du réseau d’AS sous Docker, le tout tournant dans une VM Linux. Ce choix nous permet une mise en place légère et facilement sauvegardable/reproductible de tout le réseau.

Chaque machine du réseau correspond à un container Docker, tous orchestrés par Docker Compose.

Les réseaux de l’AS sont des réseaux Docker, aussi définis en bas du fichier de configuration `docker-compose.yml`



3 Réalisation

3.1 Routage (Nam)

Un routage OSPF a été déployé à l’intérieur de l’AS grâce à la configuration de FRRouting sur les différents routeurs. FRR a été préféré à quagga car ce dernier ne voulait pas s'exécuter dans nos containers Docker…

BGP est déployé au niveau du routeur d’interconnexion pour le routage inter-AS.

3.2 Services applicatifs

3.2.1 Services DNS (Othmane et Julien)

Après avoir étudié les différents serveurs disponibles (Bind9, Unbound, dnsmasq, PowerDNS), nous avons choisi d'utiliser Unbound comme serveur DNS, et ce pour les deux DNS que nous devions déployer. Les deux DNS utilisent une configuration assez similaire, en transférant simplement les requêtes quand ils ne sont pas capables de résoudre les noms de domaine demandés (DMZ vers AS si il ne sait pas résoudre, et AS vers DMZ quand il reçoit une demande des autres DNS). Peu de difficultés ont été rencontrées lors des configurations (le plus dur étant de récupérer les adresses des DNS AS des autres groupes…), mais celles-ci restent sommaires et plus d'options pourraient être ajoutées dans le cadre d'un déploiement dans un vrai environnement d'entreprise.
Des informations complémentaires disponibles à l'adresse suivante (en anglais) :
https://nlnetlabs.nl/projects/unbound/about/

3.2.2 Service VoIP (Julien)

J'ai choisi d'utiliser Asterisk en tant que serveur de téléphonie IP, car j'avais déjà entendu parler de ce service. Il est à noter que certaines entreprises utilisent ce service pour leur VoIP interne, ce qui leur permet de gérer les coûts et le configuration plus facilement, et d'avoir la main directement dessus au lieu de le louer en 'SaaS'. Cela renforce aussi la crédibilité que l'on peut apporter à un tel service, d'où mon choix. Des difficultés ont été rencontrées lors de la configuration, dues au fait que Docker ne se comporte pas comme une VM (pas de systemd ou autre donc il faut charger les scripts selon la "méthode Docker" ou au lancement).
Des informations complémentaires disponibles à l'adresse suivante (en anglais) :
https://www.asterisk.org/

Cependant, ne parvenant pas à le faire fonctionner correctement, j'ai changé de service VoIP pour TeamSpeak. Le service TeamSpeak est lui opérationnel.
Des informations complémentaires sont disponibles à l'adresse suivante (en anglais) :
https://www.teamspeak.com/en/

3.2.3 Autres services (Nam)

Un serveur matrix et son client web ainsi qu'un serveur irc ont été déployés.

3.3 Accès OpenVPN (Rokia & Nouhaila)

Nous avons choisi de sécuriser la communication et donc établir un tunnel sécurisé entre le client VPN (réseau privé) et le serveur VPN (réseau entreprise) à l'aide d'OpenVPN,

car c'est l'un des protocoles les plus sûrs et fiables actuellement malgré qu'il soit difficile à configurer.
En ce qui concerne la configuration, nous avons initialisé OpenVPN à l'aide d'une image : kylemanna/openvpn image qui contient déjà les scripts pour la génération automatique de beaucoup de paramètres (certificats, clés PKI, authentification…). Cette image permet également de récupérer le certificat pour le compte du client après l'avoir explicitement généré, dans notre cas pour le user_1.
Une fois la connexion établie, tout le trafic entre le réseau privé et le réseau entreprise passera par le vpn.
Des informations complémentaires disponibles à l'adresse suivante (en anglais) :
https://hub.docker.com/r/kylemanna/openvpn

3.4 Accès domestique (Enzo)

Afin de protéger notre réseau privé, la configuration d'iptables nous a permis de cacher aux autres réseaux qui sont considérés comme "extérieurs" donc potentiellement dangereux, le réseau 192.168.0.0/24.
Sur la box, nous avons simplement ajouté la ligne de commande suivante :
iptables -t nat -A POSTROUTING -o eth0 -j MASQUERADE
en sachant que l'interface eth0 est notre interface extérieur "WAN" en 192.168.0.0/24.

3.5 Sécurité (Rokia & Nouhaila)

Le réseau entreprise doit aussi être protégé, c'est pour cela que nous avons décidé de mettre en place un pare-feu iptables pour interdire les paquets entrants par le routeur de l'entreprise, à l'exception de :

- communication entre les serveurs DNS (port 53)
- paquets OpenVPN (port 1194)
- paquets des communications déjà établies
- paquets des connexions HTTP et HTTPS (port 80 et 443)
- paquets ICMP pour interdire les pings de l'extérieur

Cependant, les règles bloquent les communications OpenVPN c'est pour ça elles sont mises dans le fichier firewall.sh.
Des informations complémentaires disponibles à l'adresse suivante : https://doc.ubuntu-fr.org/iptables

3.6 Sites secondaires (Amine)

Le choix original de mettre en place mailcow comme service du second site s'est avéré difficile en vue de sa lourdeur et de ses besoins en ram. Je me suis donc tourné vers Mail-in-a-box. La configuration se fait à l'aide d'une image mtrnord/mailinabox, cette image

permet de faciliter la configuration en ignorant la configuration de chaque utilisateur, elle est sensée tourner d'elle-même. Néanmoins, la mise en place docker est très lourde. Pour le service VPN censé connecter les deux sites, je me suis basé sur la partie OpenVpn de Rokia & Nouhaila, le service qui semblait bien s'exécuter avec docker compose se terminait toutefois brusquement : Il est possible que ce soit à cause de l'absence de clients, qui se trouvent sur le premier site, mais la mise en place d'un "client tampon" sur le second site ne m'a pas permis de résoudre le problème.