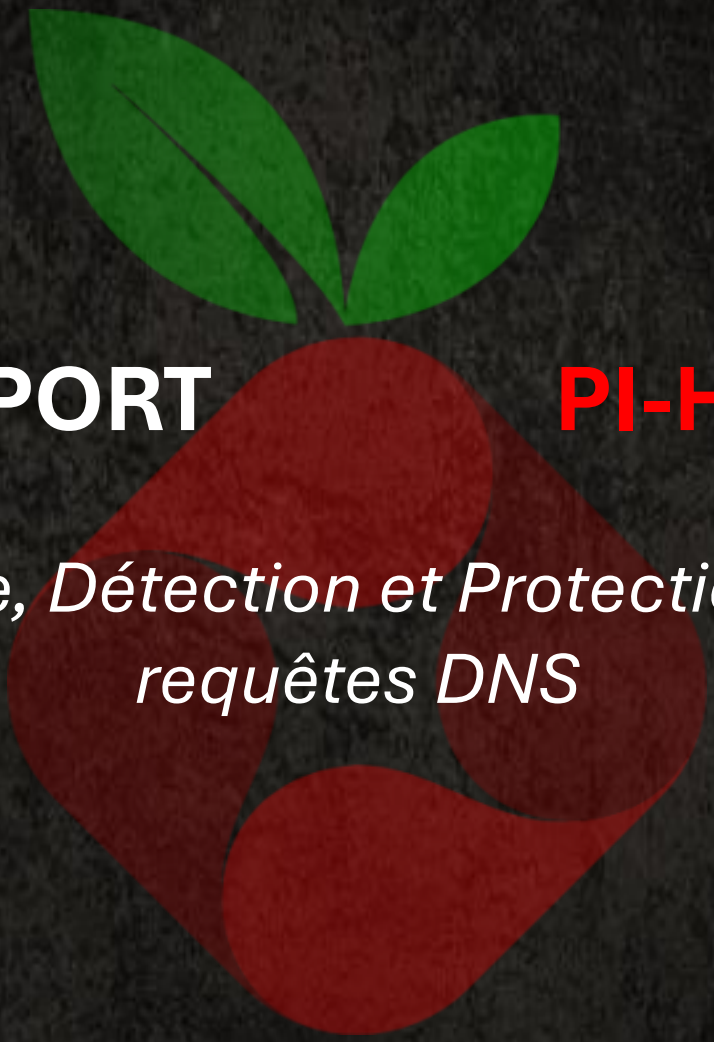
Réalisé par :

Zoran Tauvry
Quentin Gavilan
Khaled Mahdi
Brian Lock

UVSQ
université PARIS-SACLAY



RAPPORT PI-HOLE

Risque, Détection et Protection des requêtes DNS

18 Novembre 2024
2024-2025

Encadré par :
MR Olivier Callef

Sommaire :

Introduction

Dans cette étude, nous avons exploré l'installation et l'utilisation de Pi-hole dans deux contextes distincts : domestique, et en entreprise. Pi-hole, une solution de blocage de publicités et d'analyse DNS, s'est révélé être un outil puissant pour comprendre et maîtriser le trafic réseau, en particulier en ce qui concerne les requêtes DNS.

Dans un contexte domestique, l'analyse a montré que de nombreux appareils IoT, tels que les assistants vocaux, les téléviseurs connectés ou encore les caméras de sécurité, envoient régulièrement des requêtes DNS à leurs fabricants. Ces communications, souvent inutiles pour le fonctionnement de ces appareils, soulèvent des questions sur la confidentialité des données et l'utilisation abusive des informations des utilisateurs.

En entreprise, et plus précisément dans le cadre d'une société de vidéosurveillance, nous avons déployé Pi-hole pour analyser et bloquer les requêtes DNS jugées inutiles ou potentiellement dangereuses. Cette analyse a permis d'identifier des domaines de suivi publicitaire et, plus rarement, des domaines douteux ou suspects, cette étude a été conduite tout en respectant l'architecture réseau existante et en préservant les besoins de l'entreprise.

Cette mise en œuvre a également démontré son efficacité à bloquer les domaines de suivi et à limiter les risques liés aux TLD (top-level domains) peu courants ou suspects.

Recherche sur le Service DNS

Service DNS

Le Domain Name System est un service informatique, déployé vers 1985 suite à la demande de la DARPA (*Defense Advanced Research Projects)* permettant d'associer des noms de domaine avec des adresses IPV4 ou IPV6.
La hiérarchie de ce système est composée du sommet qui se nomme la *racine.* Les domaines immédiatement sous la racine sont nommés les domaines de premier niveau :

- Les domaines de premier niveau national, avec le code du pays : fr pour France, us pour les Etats-Unis ou ru pour la Russie par exemple.
- Les domaines génériques : .com, .org ou .dog par exemple, rassemblant tous ceux autre qu'une extension de pays.

Un nom de domaine entièrement nommé, *FQDN pour Fully Qualified Domain Name*, contient tous les éléments jusqu'au domaine de premier niveau, exemple avec « www.aforp.fr »
On compte 1597 actuellement en 2024, sachant que l'ICANN, l'organisme principal gérant les DNS, ajoute continuellement des nouveaux domaines de premier niveau.
Un nom de domaine est composé de plusieurs variables afin de l'identifier et de diriger le trafic Internet en fonction des besoins, on peut en citer quelques-uns :

- L'enregistrement A : permet d'associer le nom de domaine à une adresse IPv4, on aura le AAAA pour une IPv6.
- L'enregistrement MX (*Mail Exchange*) : détermine le serveur de messagerie pour ce domaine, un serveur public ou un serveur de l'intranet de l'entreprise par exemple.
- L'enregistrement NS (*Name Server*) : détermine le serveur DNS autoritaire pour ce domaine (ce même serveur DNS peut gérer une zone et donc de multiples noms de domaine).

Le nombre de noms de domaine étant abondant et Internet étant sur la globalité de la planète. On retrouve de multiples serveurs DNS gérant la racine DNS ainsi que les requêtes de premier niveau, on décompte actuellement treize identités de serveur gérés et répartis sur l'ensemble des continents par de multiples acteurs DNS. Les requêtes sont ensuite dirigées par *unicast* permettant de la router vers le serveur faisant autorité (serveurs racines) le plus proche selon le protocole de routage et le continent/région. La requête est résolue de façon itérative par le serveur récursif où la demande a eu lieu jusqu'à trouver le bon serveur de nom s'occupant de l'enregistrement demandé.

Il est également possible de faire une résolution inverse pour trouver le nom de domaine à partir de l'adresse. La particularité est que dans l'adresse IPv4, la partie la plus générale est à gauche, par exemple dans 184.219.176.29, c'est 184 qui représente la partie la plus générale (comme pour le .com dans google.com en nom de domaine). Pour la recherche, on inverse alors les

quatre termes et on y ajoute le domaine in-addr.arpa, on obtient alors 29.176.219.184.in-addr.arpa pour trouver le nom de domaine de l'IP de l'exemple.
Il est important d'avoir une résolution inverse car sa non-présence est détecté comme une erreur opérationnelle et des services comme la transmission de mail peut être refusé par l'hôte distant s'il n'y a pas de résolution inverse. D'un point de vue utilisation, cela permet également une utilisation plus simple de la commande traceroute car elle simplifie ses résultats.

Serveur DNS dans une infrastructure

La présence d'un serveur DNS dans un système d'information s'avère crucial principalement pour certains points :

- Il permet premièrement une indépendance non négligeable face à des DNS publics et empêche les utilisateurs de faire des aller-retours avec ces derniers pour optimiser la vitesse du réseau et la surcharge inutile de celui-ci. Les données DNS ne sont pas exposées protégeant ainsi les données sensibles. De plus, si une coupure internet a lieu, avoir un DNS interne permet de continuer à bénéficier des ressources interne du réseau en continuant son service.
- Il permet de rassembler les enregistrements DNS du réseau, particulièrement critique dans un intranet quand il y a un serveur de mails ou d'autres services hébergés permettant la bonne communication entre les utilisateurs et ces services. La gestion y a également centralisé ce qui simplifie la configuration des noms de domaines et IP, surtout dans un environnement AD où les noms d'ordinateurs peuvent être résolus automatiquement.

Sur un serveur Windows avec le rôle du gestionnaire DNS installé, on peut configurer les enregistrements DNS de la zone du nom de domaine ainsi que les pointeurs pour les résolutions inversées (recherche du nom domaine à partir de l'IP). Il contient également les noms d'hôtes des machines du réseau et des IP associés pour la bonne communication entre celles-ci.

- Il garde également en cache les recherches effectuées auparavant pour accélérer l'accès aux ressources pour les utilisateurs futurs, le cache dépend du TTL des enregistrements DNS et cela est unique pour chaque enregistrement (en général 3600 secondes).

Solutions aux risques

DNSSEC

Le service DNS peut présenter certaines faiblesses, en effet les requêtes circulent en clair et sont vulnérables aux attaques de *DNS poisonning,* en redirigeant les victimes sur une IP frauduleuse via l'injection de fausses réponses DNS dans le cache du serveur DNS, ou

Spoofing DNS, l'attaquant se fait passer pour le serveur DNS auprès de la victime et fournit les réponses pouvant mener à une faille.

L'extension DNSSEC permet de rajouter une couche de cryptographie asymétrique où un système de clé privée/publique est utilisé pour signer les zones ainsi que les enregistrements afin de s'assurer de la légitimité de la source ainsi que de toute la chaîne de confiance entre la zone source et la racine DNS.

Les réponses restent malgré tout en clair mais cela empêche le DNS poisoning car le résolveur rejettera la réponse si la signature de celle-ci est incorrecte à la suite d'une vérification avec la clé publique de la zone où elle est issue.

Sur un serveur Windows, il est possible de signer la zone DNS en créant un couple de clé, l'algorithme de chiffrement ainsi que la taille de la clé est personnalisable ainsi qu'un changement de clé tous les X jours si cela est nécessaire. On peut également choisir le TTL des enregistrements DNS.

Filtrage DNS

On peut également effectuer du filtrage DNS sur le réseau en bloquant l'accès à des sites malveillants aux utilisateurs essayant de s'y connecter. La requête est filtrée par le DNS qui répertorie celle-ci comme dangereuse et empêche l'utilisateur d'y accéder. Cette base peut être améliorer grâce à des listes communautaires ou même via de l'intelligence artificielle permettant d'analyser le trafic DNS pour détecter des comportements suspects et d'identifier de nouveaux sites en temps réel.

DNS-over-HTTPS

Un autre moyen pour protéger du DNS spoofing ou poisoning est ce protocole suivant, il permet d'encapsuler les requêtes DNS dans des requêtes HTTPS, via le port 443 chiffré en TLS. La requête est envoyée au serveur DoH qui peut aussi valider avec DNSSEC en addition et la renvoyer au client.
Cela concerne plus une utilisation externe car cela passe par un navigateur et des problèmes de confiance peuvent apparaître aux vues de la dépendance de serveurs DoH tiers mais il permet néanmoins une protection supplémentaire aux attaques DNS.

Cas pratique : Pi-Hole dans une infrastructure

La sécurité et l'efficacité du service DNS, comme nous l'avons vu, sont essentielles au bon fonctionnement et à la protection d'un réseau. Bien qu'un serveur DNS soit utilisé comme outil de résolution de domaine, il peut également exposer le réseau local et l'utilisateur à des menaces, comme des pop-ups malveillants ou des publicités mensongères comportant un risque de sécurité. Un filtrage des contenus indésirables ou malveillants s'impose donc pour limiter ces risques. Un outil très connu dans le domaine des filtres DNS est Pi-hole.

Ce dernier agit comme un adblocker pour le réseau entier, bloquant les requêtes vers des domaines non désirés, ce qui contribue à réduire les risques d'accès à des contenus malveillants. En installant Pi-hole sur un Raspberry Pi, vous pouvez ainsi offrir une première ligne de défense pour les appareils de votre réseau tout en optimisant les performances grâce au cache des requêtes DNS.

Dans le cadre de ce rapport, nous mettrons en place plusieurs instances de Pi-hole afin d'analyser les requêtes DNS de notre réseau et de renforcer cette sécurité.

Mise en situation

Dans cette partie, nous vous expliquerons pas à pas la configuration et l'installation de nos trois différentes instances Pi-hole afin de rappeler le contexte :

Étape 1 : Préparer le système d'exploitation

- Pi-hole est un outil principalement utilisé sous Linux. Grâce à l'OS Pi OS, il peut être installé sur un Raspberry Pi, un ordinateur miniature ne consommant pratiquement pas d'énergie et pouvant héberger des services de sécurité avancés. Nous aurons donc une instance Pi-hole installée sur un Raspberry Pi.
- Une seconde instance sera installée sur un serveur à l'aide d'une machine virtuelle émulant l'OS Debian 12, qui pourra accueillir notre Pi-hole.
- La troisième instance sera installée sur un autre système.
- Le système d'exploitation devra être à jour en utilisant les commandes de mise à jour de paquets adéquates selon son système.

Étape 2 : Installer Pi-hole

- Sur chaque instance, la commande suivante sera exécutée pour lancer le script d'installation de Pi-hole :
  - curl -sSL https://install.pi-hole.net | bash
- Une fois le script lancé, nous choisirons nos configurations spécifiques pour être en adéquation avec notre réseau :
  - **Sélectionner l'interface réseau** : Choisissez l'interface Ethernet pour une connexion stable.
  - **Choisir le fournisseur de DNS** : Pi-hole propose plusieurs options (Google, OpenDNS, etc.). Sélectionnez-en un ou configurez un serveur DNS personnalisé si vous en avez un.
  - **Configurer les listes de blocage** : Laissez les options par défaut pour les listes de blocage, qui incluent les sites de publicité courants.
  - **Activer le serveur Web** : Pi-hole dispose d'une interface Web pour la gestion, que vous pouvez activer pendant l'installation.

- **Définir l’adresse IP statique** : Pi-hole propose de fixer l’adresse IP pour éviter qu’elle ne change.
- **Configurer les options de blocage** : Laissez les paramètres par défaut pour bloquer les publicités et les trackers.
- **Notez les informations d'accès** : À la fin de l'installation, Pi-hole affichera l'adresse IP et le mot de passe pour accéder à l'interface Web de gestion (par défaut, http://<adresse-ip-pi-hole>/admin).

Une fois les deux étapes effectuées, nous nous connectons à la page d’administration de Pi-hole et constatons qu’il n’y a aucune entrée de requête. Cela est tout à fait normal, car dans la plupart des réseaux domestiques, le fournisseur d’accès internet configure le routeur avec des services DNS et DHCP pour fournir les informations de connexion, dont une IP spécifique pour le serveur DNS, qui n’est pas l’IP de notre serveur Pi-hole.
Pour résoudre ce problème, nous devons activer l’option DHCP sur notre serveur Pi-hole.

Étape 3 : Configurer Pi-hole en tant que serveur DHCP

- Allez dans *Settings* > *DHCP*.
- Cochez l'option *DHCP server enabled*.
- Définissez la plage d’adresses IP DHCP : Indiquez la plage d’adresses que vous souhaitez distribuer (par exemple, de 192.168.1.100 à 192.168.1.200).
- Définissez l’adresse IP du routeur : Indiquez l'adresse IP de votre routeur (par exemple, 192.168.1.1).
- Cliquez sur *Save* pour activer le serveur DHCP de Pi-hole.
- Désactivez le serveur DHCP de votre routeur :
  - Accédez aux paramètres de votre routeur (habituellement via une adresse comme 192.168.1.1 dans le navigateur).
  - Trouvez les paramètres DHCP et désactivez le serveur DHCP pour éviter les conflits avec Pi-hole.

À présent, après avoir attendu la fin des baux DHCP de tous les clients du réseau, nous pouvons observer la collecte des données et les requêtes effectuées par nos différents périphériques.

Analyse

Les résultats de notre Premier Pi-Hole :

Mise en place sur un petit réseau de 3 appareils et ne comportant pas d’IOT mais d’un ordinateur et d’un téléphone portable IPhone, montre un taux de blocage de 46% de requête DNS bloqué au cours des dernières 24H.

Voici une brève explication des résultats affiché sur 24h *(ref : figure1)* :

- **Total Queries (9362)** : Ce nombre indique le total de requêtes DNS faites par les appareils sur mon réseau. Ce chiffre montre l'ampleur de l'utilisation DNS.
- **Queries Blocked (4446)** : Près de la moitié des requêtes (47,5%) sont bloquées. Cela signifie que beaucoup de ces requêtes proviennent de domaines indésirables ou de publicités.
- **Domains on Adlists (121555)** : J'ai ajouté des listes d'interdiction (adlists) contenant plus de 121 000 domaines connus pour des publicités et des traqueurs. Pi-hole bloque automatiquement les requêtes vers ces domaines.

Le tableau de bord montre un aperçu de l'activité de Pi-hole, y compris le nombre total de requêtes DNS traitées, le nombre de requêtes bloquées, le pourcentage de blocage, et les domaines dans les listes d'interdiction (adlists).

Le nombre élevé de requêtes bloquées (46%) indique que Pi-hole est efficace pour bloquer les annonces et les domaines indésirables. Le pourcentage élevé de blocage est un indicateur de la quantité de publicité et de traqueurs bloqués sur mon réseau.

D'autre donnés récupère sur la page d'administration peu nous indiquer plusieurs informations importantes comme cette image capturée ***(ref : figure 2)*** montrant l'évidence des domaines les plus bloqué et les plus admis :

- **Query Types :** La majorité des requêtes sont de type A (IPv4) et AAAA (IPv6), ce qui est standard pour la navigation sur Internet. Le reste comprend des requêtes pour HTTPS et d'autres services spécifiques.
- **Upstream Servers** : J'ai configuré plusieurs serveurs DNS amont pour gérer la résolution DNS, dont Cloudflare, Google, OpenDNS et Quad9. L'utilisation de plusieurs serveurs assure une meilleure performance et une redondance en cas de défaillance de l'un d'entre eux.
- **Top Permitted Domains :** Je remarque que certains domaines, comme www.google.com et client.2.google.com, sont fréquemment autorisés, ce qui correspond à l'usage quotidien de services Google.
- **Top Blocked Domains :** De nombreux domaines bloqués, comme beacons.gvt2.com et shepherd.iff.avast.com, sont associés à des traqueurs publicitaires. Pi-hole empêche donc efficacement les publicités et le suivi sur mon réseau.

Ce graphique en camembert présente les différents types de requêtes (A, AAAA, etc.), et les serveurs DNS amont utilisés pour la résolution (caché, bloqué, ou envoyé à des serveurs DNS spécifiques comme Google ou OpenDNS).

La majorité des requêtes sont des requêtes A (IPv4) et AAAA (IPv6), l'utilisation de plusieurs serveurs DNS augmente la résilience et la rapidité de Pi-hole dans la résolution de noms de domaine.
Le tableau ***(ref : figure 3)*** montre les domaines les plus souvent autorisés et bloqués sur votre réseau. Le nombre de hits montre l'activité de certains sites comme Google, qui est fréquent dans les requêtes autorisées. En revanche, les domaines bloqués, comme ceux liés à "gv2.com" pour la publicité, indiquent que Pi-hole bloque efficacement les annonces et traqueurs indésirables.

On retrouve donc ceci dans le tableau *(ref : figure 3)* :

- **Domaines Bloqués** : Plusieurs domaines associés à la publicité, comme beacons.gvt2.com, sont bloqués automatiquement. Ces blocages empêchent les appareils de se connecter à des serveurs publicitaires.
- **Clients** : Pi-hole enregistre quel appareil a généré chaque requête, ce qui me permet d'analyser le trafic de chaque appareil. Par exemple, ici, on voit que DESKTOP-S4E2TP1 est un appareil qui génère plusieurs requêtes bloquées.

Ces tables détaillent chaque requête DNS, montrant l'heure, le type, le domaine, le client, le statut (bloqué ou autorisé), et la réponse du DNS.
Cette vue offre un aperçu détaillé de la manière dont Pi-hole filtre les requêtes en temps réel. Les requêtes bloquées pour des domaines de publicité montrent que Pi-hole identifie et bloque activement des domaines nuisibles en fonction des listes d'adresses.

Les résultats de notre Second Pi-Hole :

Mise en place sur un réseau domestique nomade comprenant plus de cinq appareils informatiques et plusieurs appareils IoT, comme un four et une TV connectée, Pi-hole montre un taux de blocage de 26,3 % des requêtes DNS au cours des sept derniers jours.
(ref : figure 4)

Avec une liste de blocage similaire à celle de notre premier Pi-hole, on observe ***(ref : figure 5)*** un affinement des résultats, passant à un blocage de 8 000 requêtes par jour, considérées comme des requêtes malveillantes, de collecte de données et de publicités. Dans ces 26,3 % de requêtes refusées, on retrouve un top 10 des noms de domaines bloqués ***(ref : figure 5)***, dont :

- Logs.netflix.com avec pas moins de 30 616 requêtes,
- Adserver.reklamstore.com avec 2 784 requêtes,
- S10.histats.com avec 2 722 requêtes.

On remarque ici une nette différence entre les requêtes du domaine *logs.netflix.com* et les deux autres domaines du top 3 des plus bloqués. À l'inverse, nous avons également un top 10 des requêtes les plus demandées et non bloquées, dont les trois premières sont également adressées à Netflix.
Après plusieurs analyses, j'ai remarqué que le périphérique qui envoie ces requêtes est ma TV Samsung, qui est d'ailleurs le périphérique envoyant le plus de requêtes DNS parmi mes appareils ***(ref : figure 6)***.

D'après cette liste appareils ***(ref : figure 6)***, nous observons que les périphériques IoT envoient en permanence des requêtes DNS à leurs fabricants alors que cela n'est pas nécessaire pour le fonctionnement de leurs technologies. Par exemple, le four *Kitchen-Service* a envoyé 5 042 requêtes en sept jours sans que la fonction de gestion à distance ait été utilisée. De même, la TV Samsung n'envoie pas moins de 133 000 requêtes DNS par semaine à Netflix, alors que nous ne sommes pas de grands utilisateurs de ce service dans notre réseau domestique. Il est donc très important de vérifier constamment les requêtes DNS envoyées et d'analyser l'utilisation des utilisateurs de ses périphérique afin de distinguer ce qui est normal de ce qui ne l'est pas.

Les analyses montrent que des dispositifs comme les télévisions et autres appareils électroménagers envoient des milliers de requêtes DNS, souvent vers des serveurs de leurs fabricant, même en l'absence d'une utilisation active de ces services. Cela soulève des préoccupations quant à la confidentialité et à la sécurité des données, car ces requêtes peuvent indiquer une collecte de données personnelles ou des comportements inattendus des appareils.

De plus, la capacité de Pi-hole à bloquer les publicités et les domaines potentiellement malveillants améliore non seulement l'expérience utilisateur en limitant les intrusions indésirables, mais aussi la sécurité globale du réseau en empêchant l'accès à des sites malveillants. Le taux de blocage de 26,3 % des requêtes DNS souligne l'ampleur des tentatives de connexion non sollicitées et démontre la vulnérabilité des réseaux face à des flux de données peu maîtrisés.

Ainsi, pour un réseau domestique nomade, l'utilisation d'un système de filtrage DNS tel que Pi-hole devient un outil indispensable pour surveiller et contrôler le trafic DNS.

Les résultats de notre Troisième Pi-Hole :

Dans cette partie du projet, nous avons déployé un Pi-hole dans le contexte d'une entreprise déjà équipée d'un serveur DNS. Il s'agit d'une entreprise spécialisée dans la vidéosurveillance, dont le nom restera confidentiel. L'infrastructure existante comprend un serveur Active Directory ainsi qu'un serveur DNS configuré pour gérer les requêtes DNS internes.

Dans l'architecture réseau actuelle, seul le serveur DNS de l'entreprise est autorisé à résoudre les requêtes DNS. Si une requête ne peut être résolue localement, elle est redirigée vers un serveur DNS public, en l'occurrence Google (8.8.8.8) ***(ref : figure 7)***.

Pour intégrer le Pi-Hole dans cette architecture, nous l'avons configuré comme redirecteur. Il reçoit les requêtes DNS transférées par le serveur DNS interne, les analyse, puis les transmet vers un serveur DNS (Upstream server) public pour leur résolution. Cette approche nous permet d'étudier les requêtes DNS tout en garantissant que le réseau de l'entreprise reste fonctionnel et non perturbé.

Lors de la fin de mon étude, les paramètres précédents ont été rétablis. Il est important de noter que nous avons déjà effectué une configuration DNS pour ce client, où nous avons non seulement désactivé les requêtes de types LLMNR, NBT et NS, mais aussi configuré une entrée wildcard dans le serveur DNS. Cette entrée redirige toutes les requêtes non résolues par le serveur DNS ou ses redirecteurs (Google, en l'occurrence) vers une adresse fictive.
Cela signifie que si une requête, comme ***nslookup example.com***, ne peut être résolue ni par le DNS local ni par Google, elle sera interceptée par l'entrée wildcard et redirigée vers l'adresse définie (par exemple, 192.168.0.1). Cette configuration a pour but de bloquer les requêtes vers des domaines inconnus ou non autorisés.

Cependant, dans le cadre de cet exercice, le client nous a accordé une fenêtre de 24 heures pour désactiver cette entrée wildcard afin d'analyser le trafic DNS, après quoi les configurations précédentes ont été rétablies. La capture d'écran suivante illustre la configuration mise en place pour rediriger le trafic DNS via le Pi-Hole.

L’analyse des données recueillies par le troisième Pi-hole a révélé un total de 23 833 requêtes, dont 2001 ont été bloquées, ce qui représente 8,4 % des requêtes totales ***(ref : figure 8)***.

L'objectif était d'étudier les requêtes vers des TLD (Top-Level Domains) peu courants, tels que ***.ru, .top, .xyz***, et d'identifier les répétitions fréquentes de requêtes vers ces domaines.
Pour ce faire, nous avons consulté des bases publiques telles que ***Talos Intelligence*** et ***VirusTotal***, afin de nous renseigner sur ces domaines et de les classer selon leur niveau de dangerosité (publicité, phishing, malware, etc.).
Dans notre tableau de bord ***(ref : figure 8)***, les domaines les plus bloqués sont les suivants :

- Beacons.gvt2.com à Beacons5.gvt2.com
- Default.exp-tas.com
- Ogads-pa.googleapis.com

Ces domaines sont principalement associés à Google, comme ***beacons.gvt2.com***, qui fait partie des services de suivi d’Analytics collectant des données sur les utilisateurs et leurs interactions avec Google. ***Default.exp-tas.com*** est quant à lui lié à des fins publicitaires.

Ces domaines figurent parmi les plus sollicités et bloqués, car Pi-hole est configuré pour protéger la vie privée des utilisateurs. Bien que ces informations soient intéressantes, nous souhaitons approfondir notre analyse en recherchant d'autres domaines potentiellement plus dangereux, mais moins fréquemment sollicités. Nous allons donc utiliser des filtres pour identifier ces requêtes moins courantes et vérifier si Pi-hole les a tout de même bloquées

Dans l'onglet Long term data Query log ***(ref : figure 10)***, il est possible d'appliquer des filtres pour visualiser et analyser les requêtes DNS émises. Il est également possible de sélectionner les types de requêtes à filtrer (permisses ou bloquées). Dans notre cas, nous avons opté pour l'affichage de toutes les requêtes, tout en cherchant les noms de domaines fréquemment associés à des attaques de type phishing, tels que les domaines en .ru, .xyz, et .cn.

Dans notre analyse ***(ref : figure 9)***, nous avons relevé plusieurs domaines bloqués, à savoir :

- mc.yandex.ru
- dpm.demdex.net
- wave.outbrain.com
- fastlane.rubiconproject.com

Après une brève recherche sur des sites comme **VirusTotal** ***(ref : figure 11)***, un service en ligne gratuit permettant d'analyser des fichiers, des URL, des adresses IP et des domaines pour détecter des menaces potentielles, il s'est avéré que ces sites sont principalement associés à des activités de tracking publicitaire.
Bien qu'ils ne soient pas toujours considérés comme malveillants, leur réputation est souvent douteuse. Ces sites sont fréquemment bloqués par Pi-hole en raison de leur activité publicitaire intrusive, qui peut être perçue comme une menace pour la confidentialité des utilisateurs.

Par exemple le domaine mc.yandex.ru **Yandex**, un moteur de recherche majeur en Russie, souvent comparé à Google dans le contexte russe.
Yandex est également impliqué dans divers services, notamment la publicité en ligne, la recherche, l'analyse de données, ainsi que des services de cloud computing et de stockage.

Bien que cela ne fasse pas de ce domaine un site malveillant au sens strict, il peut être perçu comme suspect, notamment lorsqu'on remet les choses dans leur contexte : pourquoi une entreprise française de vidéosurveillance solliciterait-elle un domaine russe, et pourquoi ce domaine aurait-il besoin d'accéder à leurs informations de suivi ?

Dans la figure 9, les domaines autorisés concernent principalement les applications de vidéosurveillance. Ces informations ont été cachés à la demande du client et relèvent d'un usage légitime.

Les autres domaines, tels que **google-ohttp-relay-safebrowsing.fastly-edge.com**, sont utilisés à des fins d'équilibrage de charge. Ils permettent de connecter les utilisateurs aux centres de données les plus proches pour optimiser les performances.

Enfin, une précision importante : dans ce cas précis, nous observons qu'une seule machine effectue les requêtes DNS. Cela est logique et intentionnel, car notre DNS principal est configuré pour être le seul à solliciter ou à transférer les requêtes non résolues vers le Pi-hole. Par conséquent, toutes les requêtes issues du parc de machines passent d'abord par ce DNS principal avant d'être transmises au Pi-hole pour analyse.

L'adresse IP de la machine DNS a été masqué à la demande du client.

Conclusion

L’utilisation de Pi-hole dans ces deux contextes met en évidence la pertinence et la flexibilité de cet outil pour améliorer la sécurité des échanges réseau

Dans un cadre domestique, il révèle l’omniprésence des requêtes DNS inutiles provenant des appareils IoT, pointant vers un besoin croissant de sensibilisation à la confidentialité des données.

Dans un environnement professionnel et d’entreprise, Pi-hole joue un rôle central dans l’identification des risques, en bloquant les domaines publicitaires, et potentiellement malveillants, tout en respectant les besoins légitimes.

Ces résultats montrent que l’analyse DNS via Pi-hole constitue un levier important pour une meilleure gestion de la sécurité réseau.

Dans un monde ou la collecte de données est croissante il est nécessaire de mettre en place de tels outils pour protéger la confidentialité et vie privée d’une entreprise mais aussi celles des individus dans leur domicile.

Annexe :

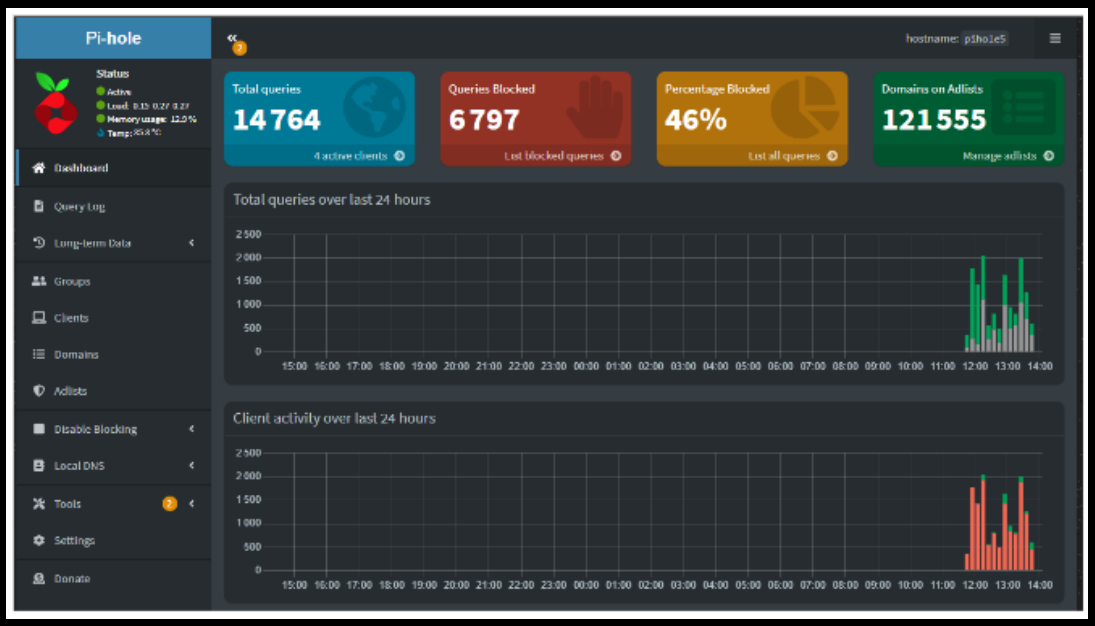


Figure 1 : Tableau de Bord du Pi-Hole montrant les requêtes DNS bloqué et envoyé du 1er Pi-Hole

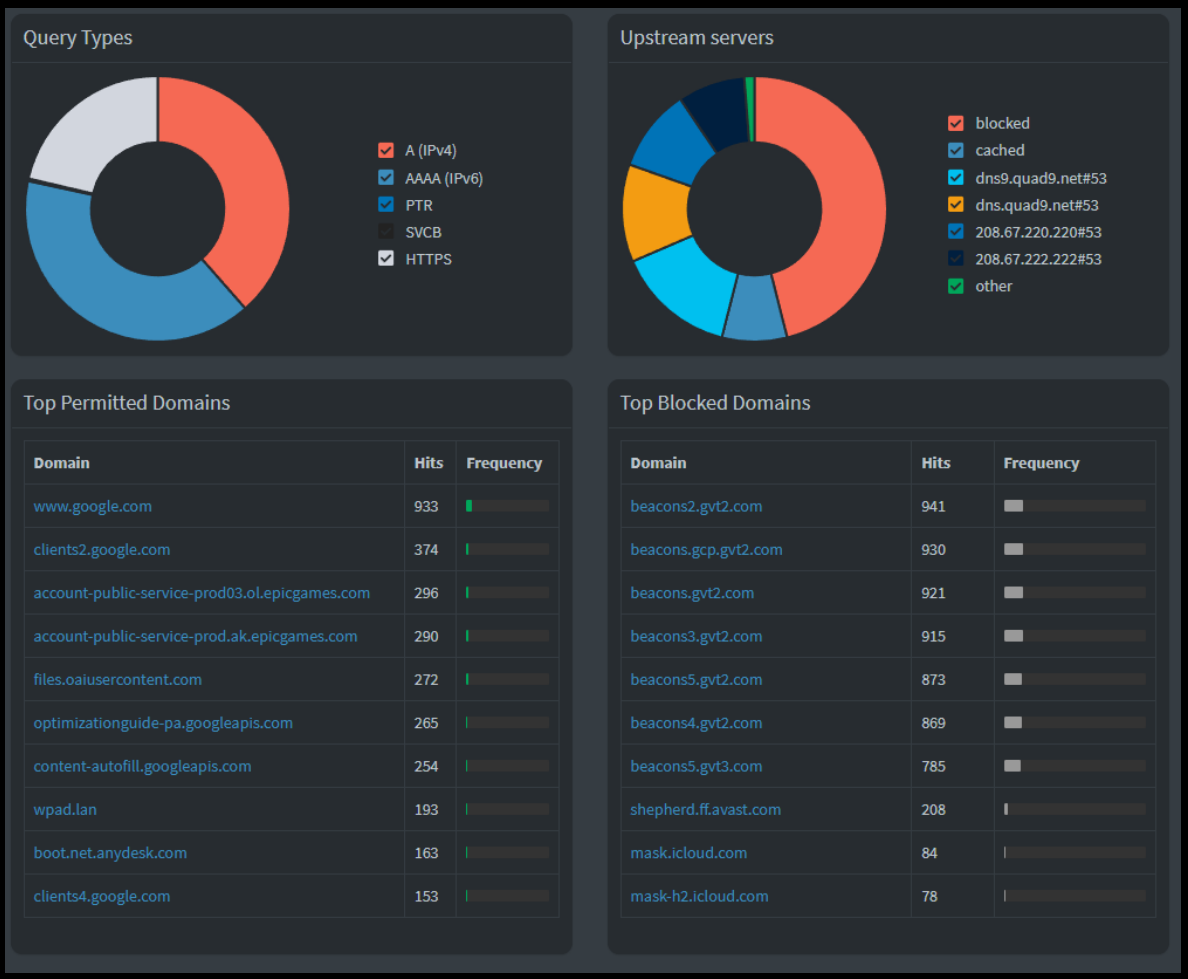


Figure 2 : Liste Top Domaine permit et Bloqué du 1er Pi-Hole

Time	Type	Domain	Client	Status	Reply	Action
2024-11-15 13:47:54	AAAA	www.m5jquj.top	localhost	Blocked (gravity)	IP (3.3ms)	Whitelist
2024-11-15 13:47:54	A	download.realtimegaming.com	localhost	Blocked (gravity)	IP (2.5ms)	Whitelist
2024-11-15 13:43:24	HTTPS	beacons4.gvt2.com	DESKTOP-S4E26TP.Ian	Blocked (gravity)	NODATA (0.0ms)	Whitelist
2024-11-15 13:43:24	A	beacons4.gvt2.com	DESKTOP-S4E26TP.Ian	Blocked (gravity)	IP (0.0ms)	Whitelist
2024-11-15 13:43:24	AAAA	beacons4.gvt2.com	DESKTOP-S4E26TP.Ian	Blocked (gravity)	IP (0.0ms)	Whitelist
2024-11-15 13:43:24	AAAA	beacons4.gvt2.com	DESKTOP-S4E26TP.Ian	Blocked (gravity)	IP (0.0ms)	Whitelist
2024-11-15 13:43:24	A	beacons4.gvt2.com	DESKTOP-S4E26TP.Ian	Blocked (gravity)	IP (0.0ms)	Whitelist
2024-11-15 13:43:24	HTTPS	beacons4.gvt2.com	DESKTOP-S4E26TP.Ian	Blocked (gravity)	NODATA (0.0ms)	Whitelist
2024-11-15 13:43:24	A	beacons4.gvt2.com	DESKTOP-S4E26TP.Ian	Blocked (gravity)	IP (0.0ms)	Whitelist
2024-11-15 13:43:24	AAAA	beacons4.gvt2.com	DESKTOP-S4E26TP.Ian	Blocked (gravity)	IP (0.0ms)	Whitelist
Time	Type	Domain	Client	Status	Reply	Action

Figure 3 : Tableau listant les domaines bloqués à un temps donné du 1er Pi-Hole

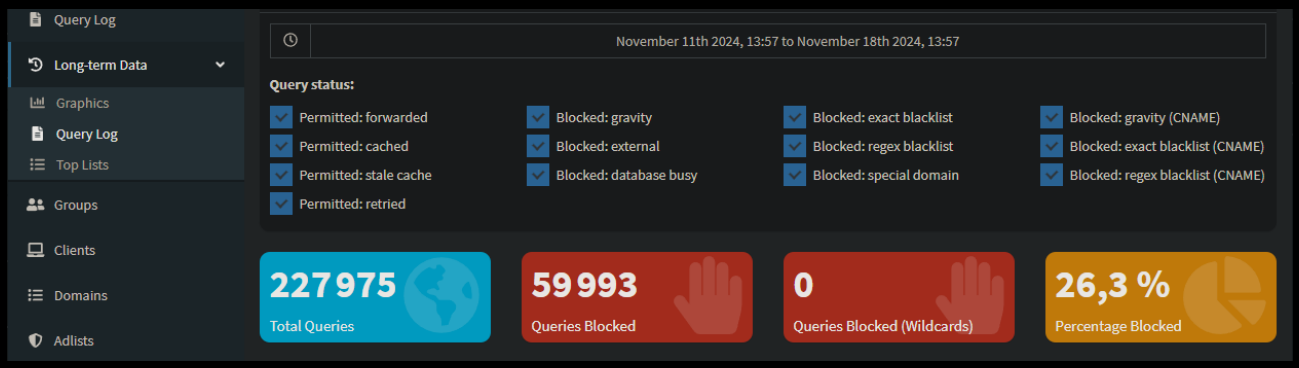


Figure 4 : Résultat total des requêtes DNS émis et bloqué sur 7 jours du 2e Pi-Hole

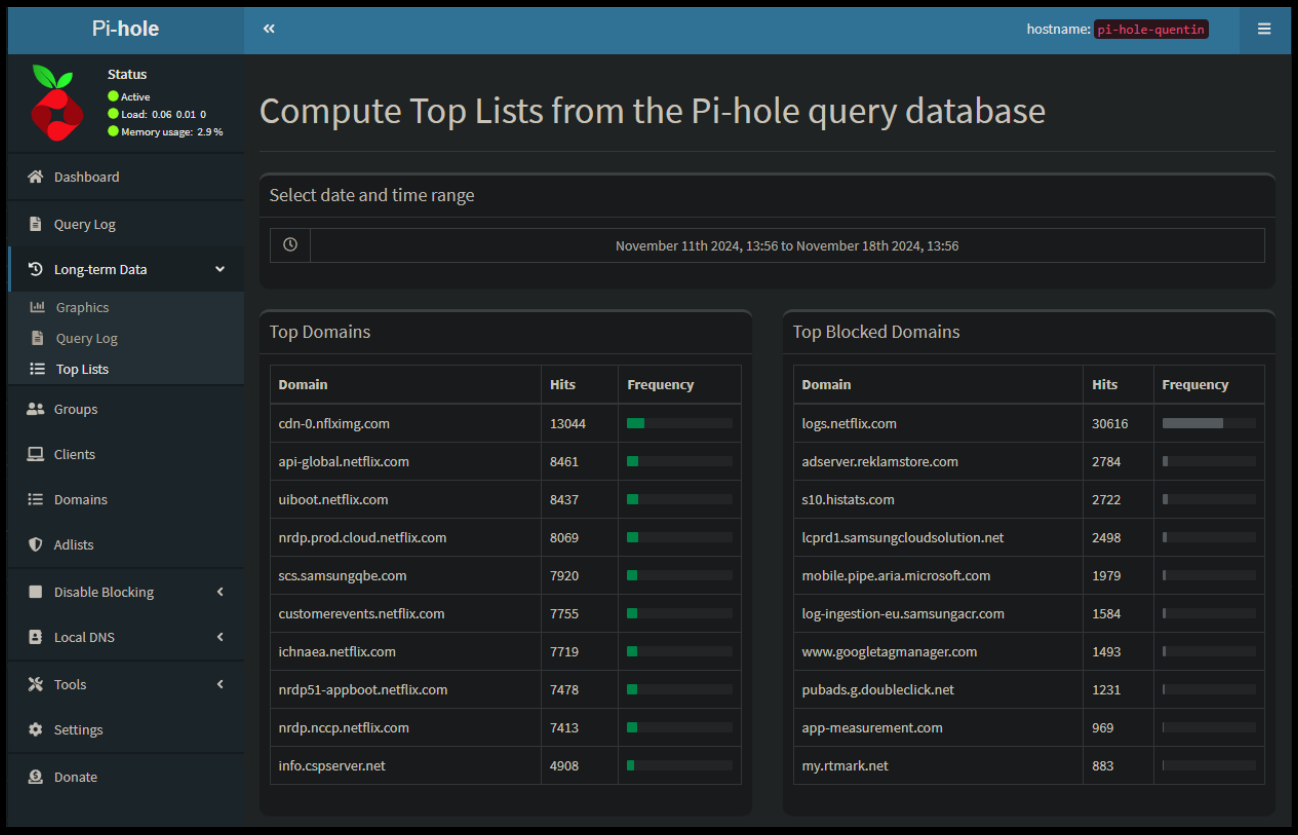


Figure 5 : Liste Top Domaine permit et Bloqué du 2e Pi-Hole

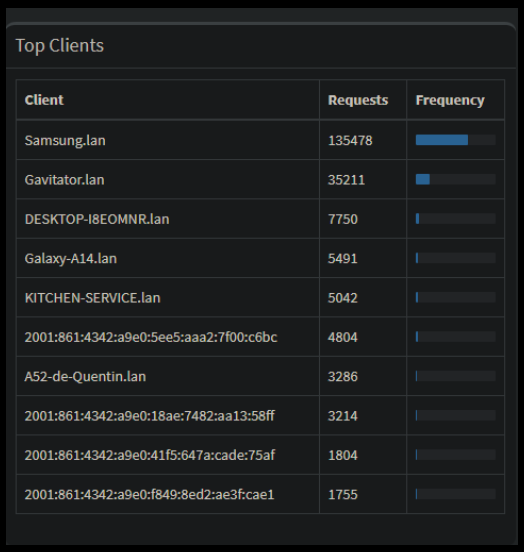


Figure 6 : Liste Top client du 2e PI-Hole

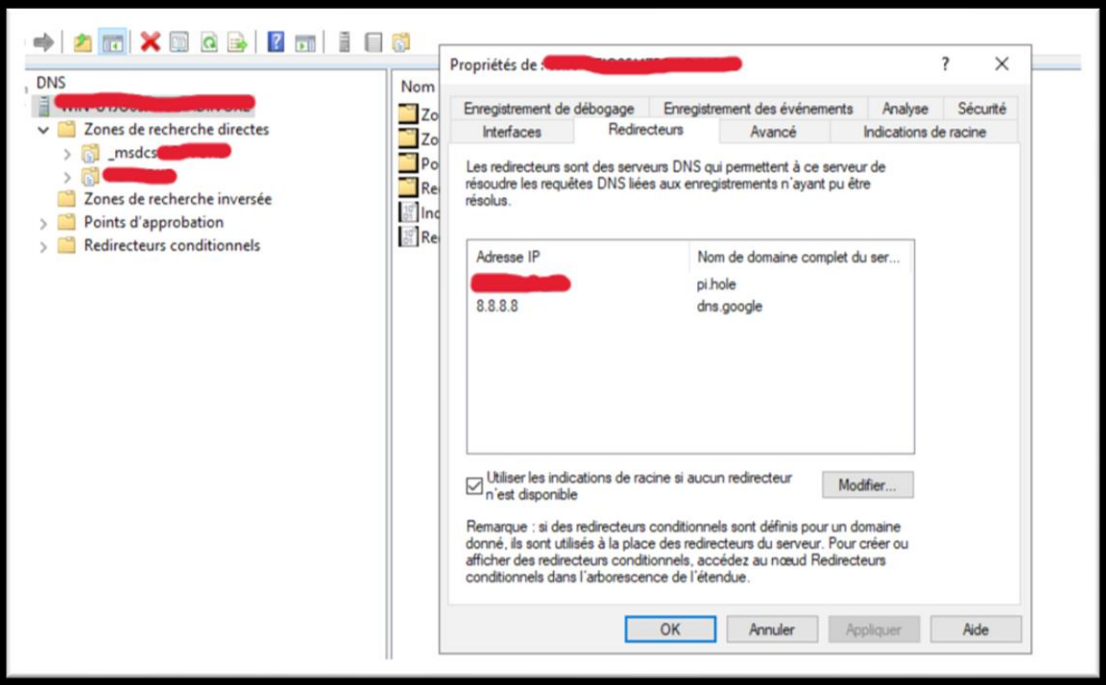


Figure 7 : Configuration Pi-Hole comme redirecteur principal du 3e Pi-Hole

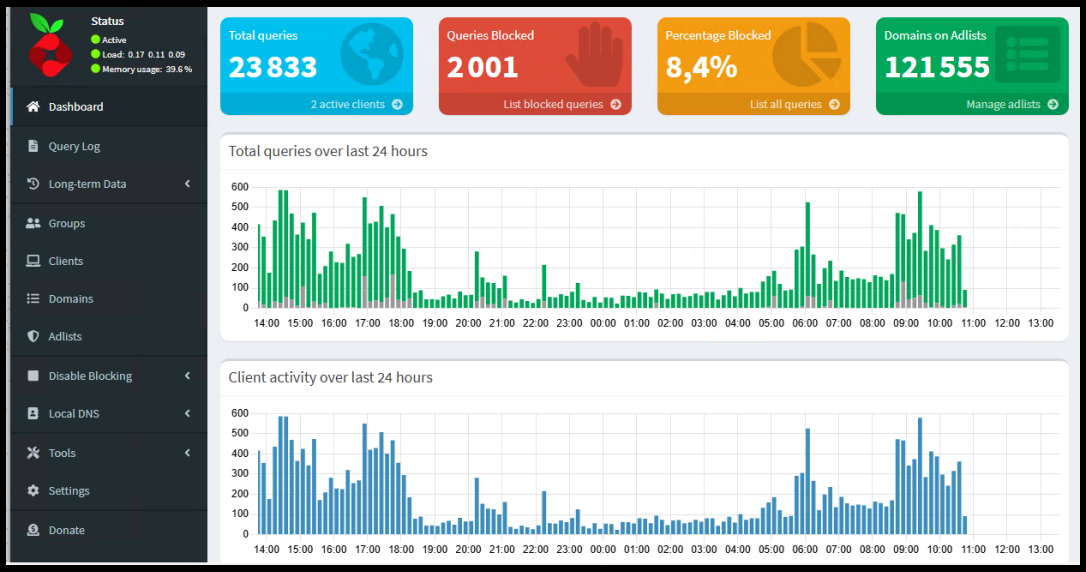
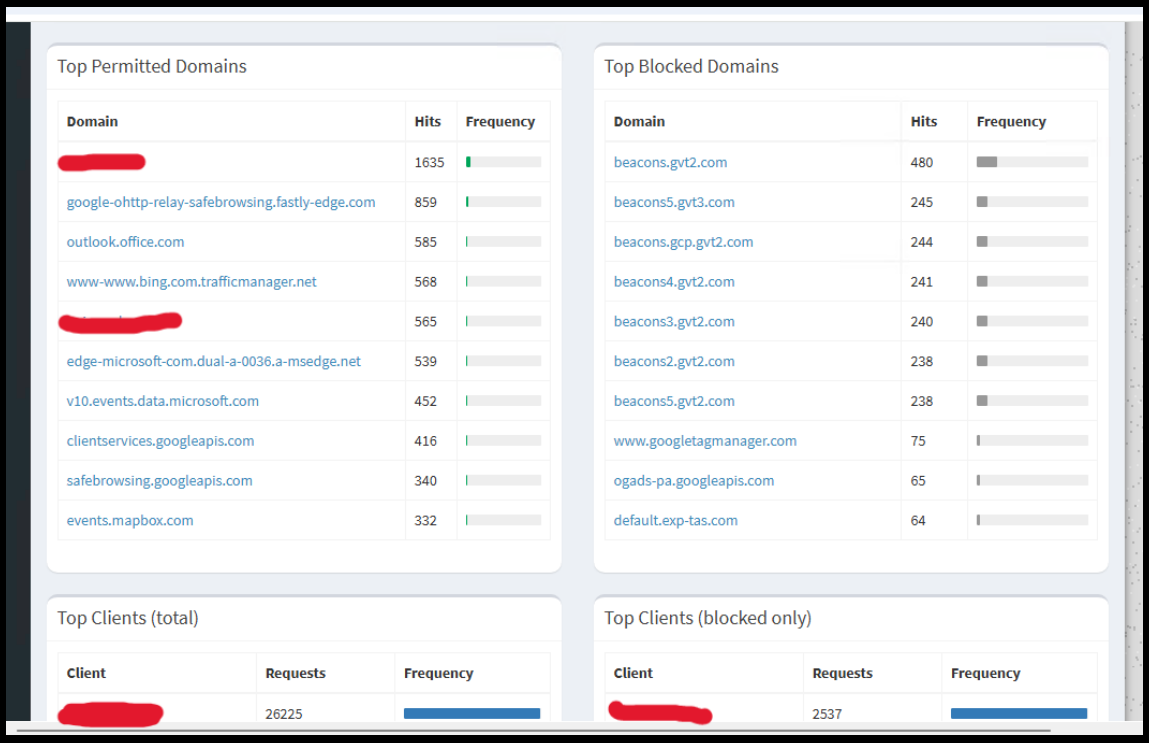


Figure 8 : Tableau de bord de la page d'admin du 3e Pi-Hole



Top Permitted Domains

Domain	Hits	Frequency
[redacted]	1635	
google-ohttp-relay-safebrowsing.fastly-edge.com	859	
outlook.office.com	585	
www-www.bing.com.trafficmanager.net	568	
[redacted]	565	
edge-microsoft-com.dual-a-0036.a-msedge.net	539	
v10.events.data.microsoft.com	452	
clientservices.googleapis.com	416	
safebrowsing.googleapis.com	340	
events.mapbox.com	332	

Top Blocked Domains

Domain	Hits	Frequency
beacons.gvt2.com	480	
beacons5.gvt3.com	245	
beacons.gcp.gvt2.com	244	
beacons4.gvt2.com	241	
beacons3.gvt2.com	240	
beacons2.gvt2.com	238	
beacons5.gvt2.com	238	
www.googletagmanager.com	75	
ogads-pa.googleapis.com	65	
default.exp-tas.com	64	

Top Clients (total)

Client	Requests	Frequency
[redacted]	26225	

Top Clients (blocked only)

Client	Requests	Frequency
[redacted]	2537	

Figure 9 : listes des domaines bloqués et permis du 3e Pi-Hole

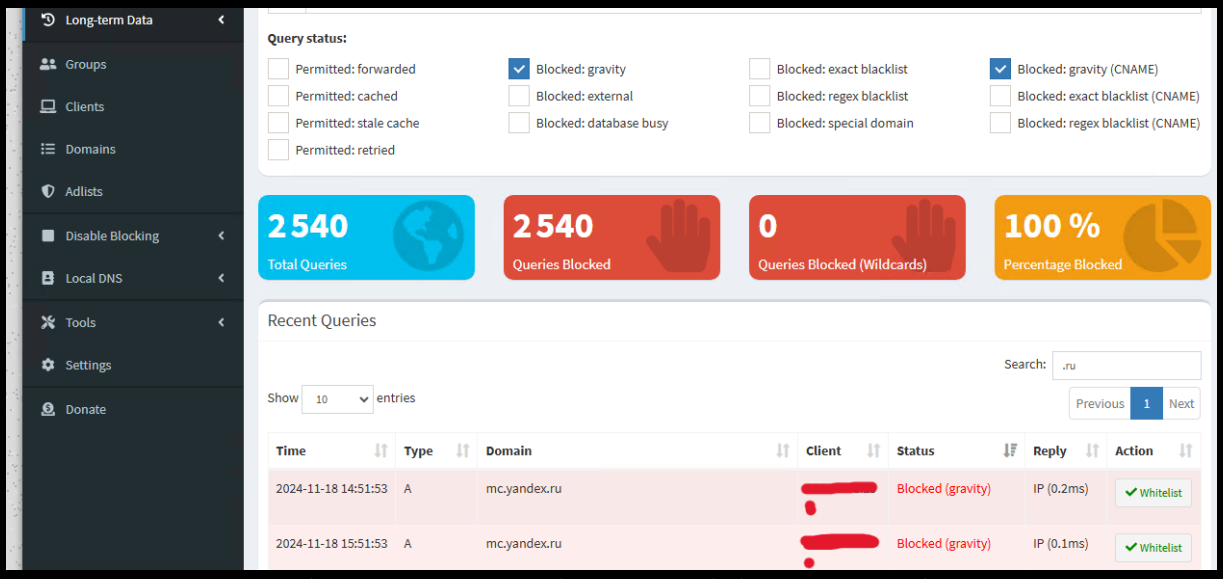


Figure 10 : Domaines suspicieux du 3e Pi-Hole

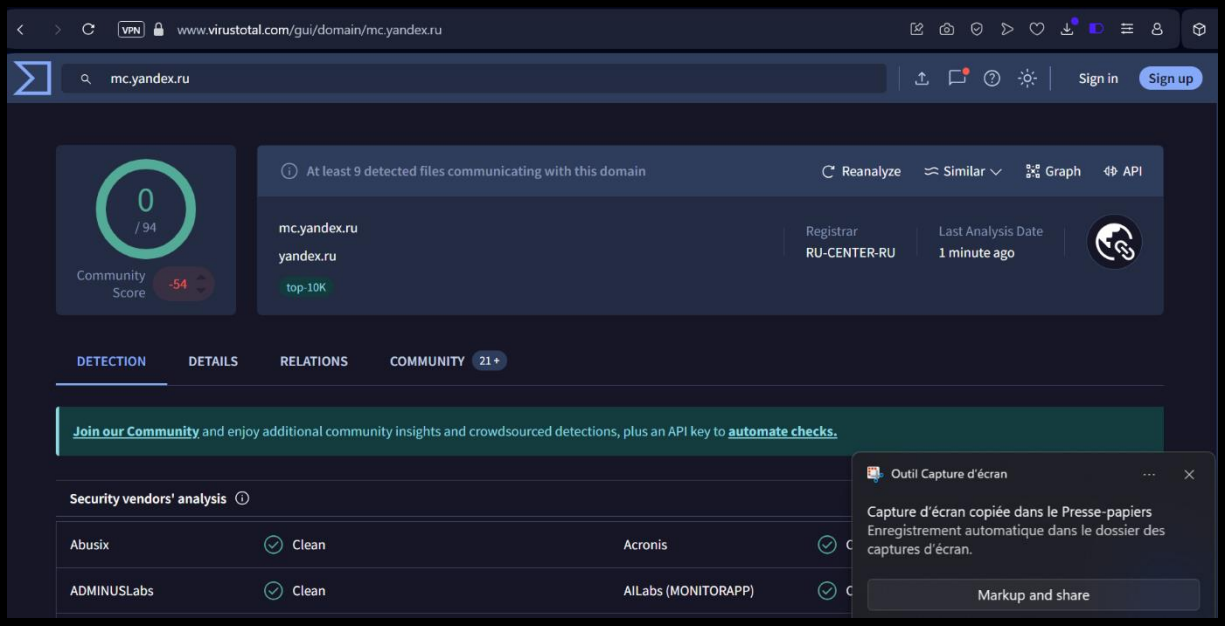


Figure 11 : Réputation Yandex sur VirusTotal

Source :

Site Officiel de Pi-Hole : https://pi-hole.net/

Forum Communautaire pour résolution de problème : https://discourse.pi-hole.net/

Page reddit pour les recommandations : https://www.reddit.com/r/pihole/

Wikipédia de Pi-Hole : https://fr.wikipedia.org/wiki/Pi-hole