

Smartphone et forensique : comment attraper Pegasus for fun and non-profit.

Étienne Maynier

`etienne.maynier@amnesty.org`

Amnesty International

Résumé. En juillet 2021, le projet Pegasus [15,32] a mis sur le devant de la scène les abus commis par 11 pays utilisant le logiciel-espion Pegasus vendu par la société israélienne NSO Group. Cette investigation a été menée par un consortium de 17 médias internationaux, coordonné par l'organisation Forbidden Stories et en collaboration avec le Security Lab d'Amnesty International. Sans rentrer dans le détail de chaque révélation, cet article propose de revenir sur l'histoire de NSO Group, le fonctionnement de Pegasus et de détailler la méthodologie forensique utilisée pour démontrer techniquement l'infection ou tentative d'infection d'un grand nombre de téléphones de défenseur·ses des droits humains.

1 Des attaques informatiques contre la société civile

Ces dernières années, les attaques informatiques contre les défenseur·ses des droits humains (appelés DDH par la suite) ainsi que les journalistes se sont multipliées [4], bien souvent avec les mêmes outils et techniques utilisées contre des gouvernements ou des entreprises. Par exemple, les attaques contre les ONG tibétaines en exil sont le cas le plus documenté depuis 2009, avec plus d'une dizaine de rapports décrivant l'évolution de ces attaques, certaines attribuées à des groupes bien connus dans l'industrie comme APT1 [20].

Dès 2012, il est apparu qu'un certain nombre d'états utilisaient les malwares développés par des entreprises commerciales, notamment les entreprises européennes Hacking Team et Gamma Group/Finfisher, pour cibler des DDH et journalistes (par exemple au Bahreïn [19] ou au Maroc [30]). Cette première génération d'entreprises a été à la tête de ce marché de la surveillance entre 2010 et 2015 vendant principalement des malwares pour Windows et Android. Les piratages de ces entreprises en 2014 et 2015 [33] ont permis de mettre en lumière l'ampleur des abus rendus possibles par cette industrie, avec des clients dans plusieurs dizaines de pays. Même s'il a fallu attendre mars 2022 pour voir la fin de FinFisher [3], ces deux entreprises se sont effondrées suite à ces révélations, laissant une place à prendre.

En 2016, un premier rapport [21] a révélé l'existence de NSO Group, nouveau leader de ce marché avec un logiciel-espion appelé Pégasus permettant de pirater des téléphones portables. NSO Group s'est construit dans l'espace libre laissé par Hacking Team et FinFisher mais également sur la promesse de pouvoir pirater des iPhone à grand renfort de vulnérabilités 0-day, ce qui a su attirer des investisseurs, notamment européens et états-uniens. Cette seconde génération d'entreprises (basées principalement en Israël) a pu se développer de manière démesurée : avant les révélations du projet Pégasus, NSO Group comptait par exemple plus de 800 salarié·es et sa valeur était estimée à plus d'un milliard de dollars (à titre de comparaison, l'entreprise Hacking Team n'a jamais compté plus de 50 salarié·es).

2 NSO Group & Pégasus

Au sein d'Amnesty International, nous avons commencé à nous intéresser de près à NSO Group lorsqu'un membre du staff s'est fait attaquer par les outils de cette entreprise en juin 2018 [10]. Nous avons ensuite publié plusieurs rapports sur des attaques utilisant Pégasus contre des DDH au Maroc [11,12], puis au Mexique [31], avant de participer au projet Pégasus en 2021. Ces années de recherche nous ont donné une bonne vue d'ensemble du fonctionnement de Pégasus et notamment des moyens d'infection.

2.1 Fonctionnement

Un document commercial de NSO Group datant de 2012 [5] fournit encore à ce jour la meilleure vue d'ensemble des fonctionnalités de Pégasus. Pégasus est un malware commercial développé exclusivement pour smartphone et vendu officiellement à des fins de lutte contre le terrorisme. Il se base sur l'utilisation de vulnérabilités 0-day pour l'infection et l'élévation de privilège afin de s'installer au plus haut niveau de privilège du système, ce qui lui donne accès à toutes les données présentes sur le téléphone : SMS, appels, images mais également données d'applications chiffrées de bout en bout ainsi qu'à la caméra, au micro et à la puce GPS.

Il est malheureusement difficile de savoir à quoi ressemble Pégasus aujourd'hui car les derniers samples de Pégasus identifiés et analysés datent de 2016 pour la version iOS [25] et de 2017 pour la version Android [26, 29]. Ceci est largement dû aux précautions prises par NSO Group pour éviter d'être découvert. Ainsi, ces anciennes versions surveillent l'état

du téléphone et suppriment le malware si une tentative de jailbreak est découverte, ou si le malware ne peut pas communiquer avec les serveurs de Commande & Contrôle pendant un certain temps. Nous pensons également que les versions récentes de Pegasus n'ont pas de persistance sur le système et sont donc supprimées par un simple redémarrage du téléphone (les infections sans clics permettant de réinfecter le téléphone si besoin), rendant très difficile la récupération du logiciel lors d'une analyse.

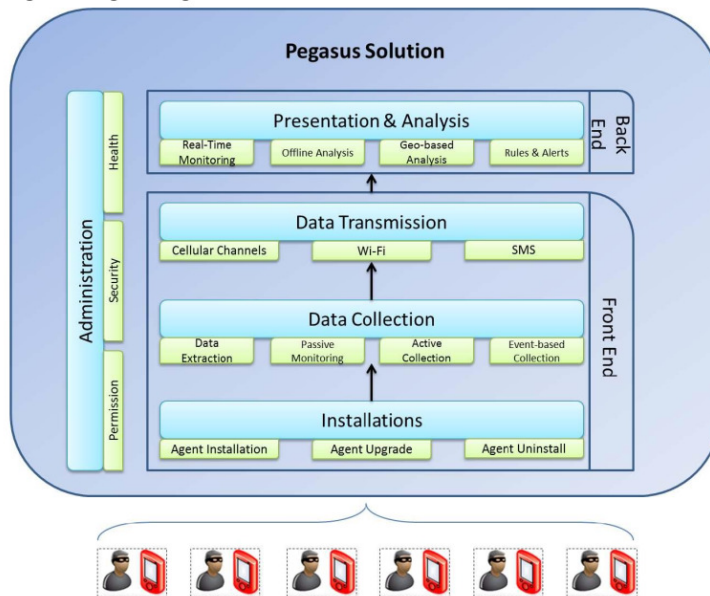


Fig. 1. Infrastructure logicielle de Pegasus en 2012 (Source : NSO documentation [5])

Pegasus utilise un réseau de serveurs d'anonymisation (appelé par NSO Pegasus Anonymizing Transmission Network — PATN) entre les téléphones compromis et les serveurs de Commande & Contrôle ou les serveurs d'infection, permettant de masquer l'identité du client utilisant Pegasus. Au cours de nos recherches, nous avons développé des empreintes de ces serveurs d'anonymisation afin d'identifier et de relier les activités d'un grand nombre de domaines et serveurs, et ce pour plusieurs générations d'infrastructures de NSO. Une empreinte consiste à trouver un set d'informations spécifiques à la configuration de ce serveur, et chercher d'autres serveurs partageant cette configuration. Par exemple, une des empreintes que nous avons développées reposait sur la liste d'algorithmes

de chiffrement supportés par TLS (un fonctionnement assez proche de JARM). Cette liste de domaines et serveurs nous a ensuite permis d’attribuer ces attaques à l’infrastructure de Pégasus (voir [13] pour plus de détails).

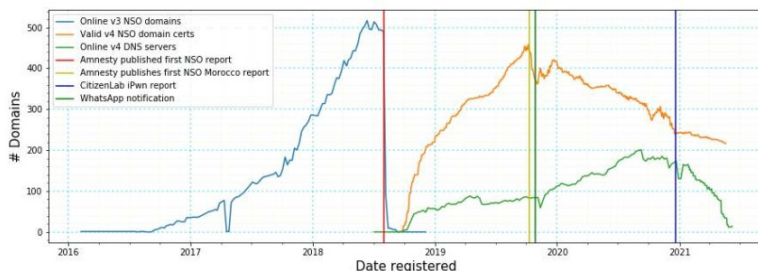


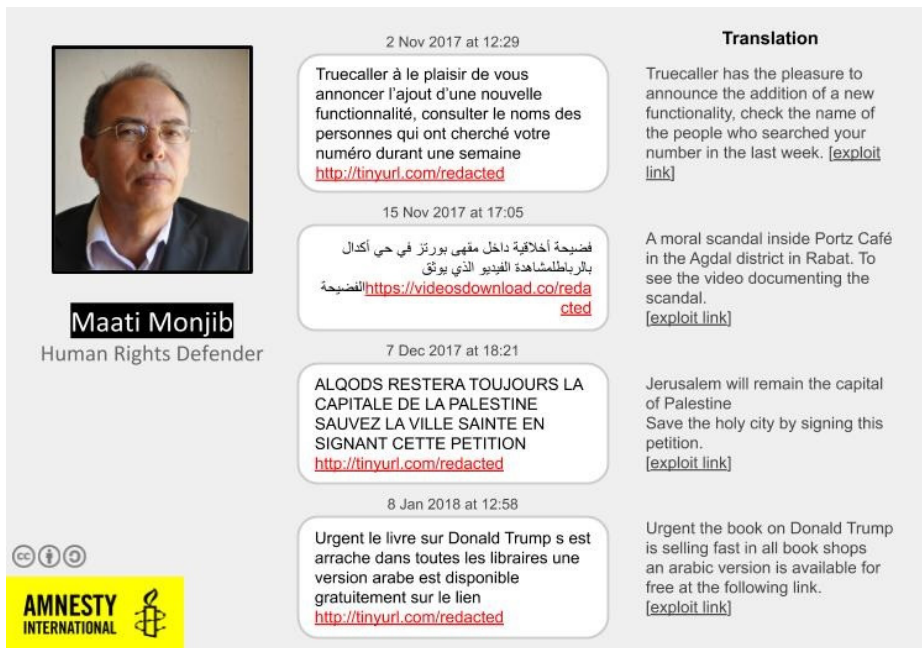
Fig. 2. Evolution du nombre de domaines de l’infrastructure de Pégasus (source : rapport Amnesty [13])

Enfin, un backend est installé chez le client final de NSO, et permet de mener des attaques et de récolter les informations des téléphones compromis. Les informations accessibles par NSO Group sur les personnes ciblées par Pégasus est soumis à controverse : NSO Group prétend ne pas savoir qui est ciblé par ses clients car ces informations ne sont disponibles que sur les serveurs installés chez eux, mais a plusieurs fois nié que Pégasus avait été utilisé pour cibler certaines personnes. En particulier, NSO Group a publiquement nié le fait que Pégasus ait été utilisé pour cibler le journaliste Jamal Khashoggi et ses proches ; or, nous avons démontré durant le projet Pégasus que deux de ses proches avaient bien été ciblées avant et après son assassinat [27].

2.2 Modes d’infection

Attaques par SMS Jusqu’en 2019, la plupart des attaques de Pégasus identifiées reposaient sur l’envoi de liens par SMS. Un clic sur un de ces liens conduisait à l’exploitation du navigateur suivie d’une élévation de privilège afin d’installer Pégasus (seule une chaîne d’exploits de ce type a été identifiée par le Citizen Lab en 2016 [21]).

En termes de recherche, tous les rapports de cette période (par exemple au Mexique [22]) se sont basés sur la recherche de SMS malveillants et une attribution à l’infrastructure de Pégasus via une empreinte de celle-ci.



2 Nov 2017 at 12:29

Truecaller à le plaisir de vous annoncer l'ajout d'une nouvelle fonctionnalité, consulter le noms des personnes qui ont cherché votre numéro durant une semaine
<http://tinyurl.com/redacted>

15 Nov 2017 at 17:05

فضيحة أخلاقية داخل مقهى بورتز في حي أكدال بالرباط لمشاهدة الفيديو الذي يوثق للفضيحة
<https://videodownload.co/redacted>

7 Dec 2017 at 18:21

ALQODS RESTERA TOUJOURS LA CAPITALE DE LA PALESTINE SAUVEZ LA VILLE SAINTE EN SIGNANT CETTE PETITION
<http://tinyurl.com/redacted>

8 Jan 2018 at 12:58

Urgent le livre sur Donald Trump s est arrache dans toutes les libraires une version arabe est disponible gratuitement sur le lien
<http://tinyurl.com/redacted>

Translation

Truecaller has the pleasure to announce the addition of a new functionality, check the name of the people who searched your number in the last week. [\[exploit link\]](#)

A moral scandal inside Portz Café in the Agdal district in Rabat. To see the video documenting the scandal. [\[exploit link\]](#)

Jerusalem will remain the capital of Palestine Save the holy city by signing this petition. [\[exploit link\]](#)

Urgent the book on Donald Trump is selling fast in all book shops an arabic version is available for free at the following link. [\[exploit link\]](#)

AMNESTY INTERNATIONAL

Fig. 3. Attaques par SMS contre le DDH marocain Maati Monjib (source : rapport Amnesty [11])

Vulnérabilités dans des applications Aux alentours de 2018/2019, NSO Group a commencé à déployer largement des techniques d'infection ne demandant pas d'interaction avec la personne ciblée (couramment appelées attaques 0-click). Ces attaques sont principalement basées sur des vulnérabilités logicielles dans des applications. En octobre 2019, WhatsApp a révélé que NSO Group avait utilisé une faille dans l'application (CVE-2019-3568) pour cibler 1400 personnes entre avril et mai 2019, notamment plus d'une centaine de DDH, avocat-es, universitaires et journalistes [6].

Lors de notre investigation, nous avons identifié des traces forensiques d'exploitation de différentes applications sur iPhone, principalement iMessage mais également Apple Photo et Apple Music [13]. Nous avons notamment identifié une vulnérabilité 0-day dans iMessage exploitée de début 2021 à juillet 2021 pendant notre investigation. Cette vulnérabilité que nous avons baptisée Mégalodon a également été identifiée par le Citizen Lab (sous le nom FORCEDENTRY [23]) qui a trouvé des traces forensiques permettant à Apple de la corriger en septembre 2021 dans iOS 14.8 (CVE-2021-30860).

Injection de trafic Lors de nos recherches sur l'utilisation de Pégasus au Maroc, nous avons identifié deux cas d'attaques par injection de trafic [11, 12]. Par une analyse forensique des téléphones infectés, nous avons observé des redirections lors de la visite de certains sites en HTTP vers des domaines de l'infrastructure de Pégasus, suivies de traces d'infections par Pégasus.

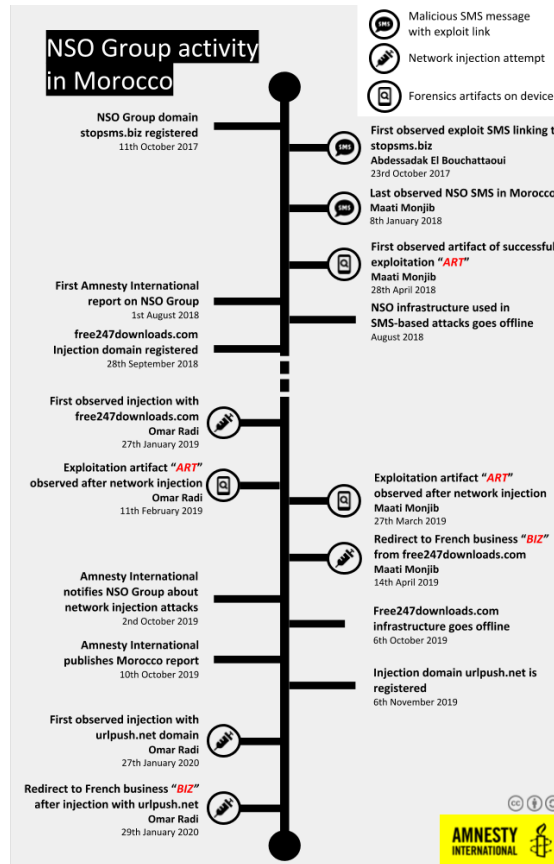


Fig. 4. Attaques utilisant Pégasus au Maroc (source : rapport Amnesty [12])

Cette injection de trafic peut-être réalisée par deux moyens : soit en déployant un système d'injection de trafic sur le réseau marocain (ou à la frontière entre le réseau marocain et les différents réseaux upstream), soit en utilisant un système de type IMSI Catcher. Les IMSI Catchers (parfois appelés Stingrays) sont des fausses antennes-relais mobiles permettant l'identification et parfois l'interception de trafic de téléphones mobiles

dans une zone géographique précise. L'utilisation d'injection de trafic pour infecter un téléphone avec Pégasus a seulement été démontrée au Maroc, et nous ne savons pas si cette technique est ou a été utilisée par d'autres clients de NSO Group.

2.3 Impact de Pégasus

Il serait incomplet de ne voir le problème de la surveillance ciblée illégale que par le prisme d'une attaque contre la vie privée. Ces attaques se passent toujours dans un contexte extrêmement tendu en matière de droits humains, et s'ajoutent à tout un arsenal répressif utilisé contre ceux qui luttent pour les droits humains ou une presse indépendante. Ainsi, la première personne identifiée comme ayant été une cible de Pégasus, le militant émirati Ahmed Mansoor, est en prison depuis mars 2017 pour menace à l'ordre public malgré les demandes de libération faites par de nombreuses organisations. De même, beaucoup de journalistes ciblé-es par Pégasus font déjà leur travail dans des pays où la liberté de la presse est régulièrement attaquée, comme au Maroc, au Mexique ou en Inde. Enfin, il est tentant mais faux de considérer que ce problème ne touche que les « pays du sud », car nous savons maintenant que des journalistes, militant·es et opposant·es politiques ont été ciblé-es par Pégasus en France et en Europe (Hongrie, Pologne et Espagne notamment).

Si ces attaques représentent un problème grave en termes de liberté d'expression, il faut également bien comprendre les conséquences personnelles qu'elles ont pour les personnes ciblées. Se rendre compte qu'une partie intime de sa vie a été espionnée peut provoquer un choc psychologique important dont il peut être difficile de se remettre.

3 Faire de l'analyse forensique de smartphones

Du fait de l'évolution de ces attaques vers des attaques sans clic, nous avons dû développer une méthodologie d'analyse forensique pour identifier des traces de Pégasus sur smartphone.

L'analyse forensique de smartphones pose plus de problèmes techniques que l'analyse d'ordinateurs en raison de leur architecture plus fermée. Deux problèmes majeurs se sont posés à nous : d'une part, comment accéder à des données utiles sur un smartphone ? Et, d'autre part, comment identifier des traces de Pégasus ?

L'accès aux données pose plus de problèmes qu'on ne le pense au premier abord. La situation idéale serait de pouvoir rooter ou jailbreaker un

téléphone et récupérer les données afin de les analyser. Malheureusement, ces méthodes posent des risques de fiabilité (notamment sur Android), des risques en termes de dommages sur le téléphone (briquer un téléphone est peu courant, mais cela arrive), et enfin laissent des traces sur le téléphone qui peuvent le rendre moins sécurisé et moins utilisable (beaucoup d'applications — notamment bancaires — refusent de fonctionner sur un téléphone qui a été rooté ou jailbreaké). Il est donc difficile d'utiliser une technique aussi intrusive de manière systématique pour vérifier un grand nombre des téléphones de DDH.

Il faut donc regarder les autres types de données disponibles, et nous nous sommes notamment intéressés aux backups. Si cette fonctionnalité a été peu à peu abandonnée sous Android, elle est encore centrale sur les iPhone et les backups faits, par exemple, avec iTunes enregistrent beaucoup de données, y compris des informations spécifiques au fonctionnement du système.

La recherche de traces de Pégasus a nécessité un travail de fourmi en analysant un par un des fichiers système, la plupart du temps non documentés, au fur et à mesure que nous analysions des téléphones potentiellement piratés. La communauté forensique internationale se consacrant majoritairement à un travail de police cherchant à analyser les faits et gestes du propriétaire de l'iPhone, il existe en réalité peu de ressources sur la détection d'intrusions pour smartphone d'un point de vue forensique. Nous avons donc au fur et à mesure des cas, développé une nouvelle liste d'artefacts forensiques, en ayant parfois une compréhension partielle de certains champs ou fichiers générés par le système.

Si cette recherche a été fructueuse sur iPhone en se basant notamment sur les données nombreuses disponibles dans les backups, elle n'a malheureusement pas abouti sous Android et nous continuons à chercher une méthode aussi efficace pour ce système d'exploitation. La suite de cet article portera donc exclusivement sur l'analyse d'iPhone, même si le Mobile Verification Toolkit fournit tout de même des moyens d'analyse de téléphones Android (voir notamment [7, 8]).

4 Méthode d'analyse d'iPhone

Lors de la publication du projet Pégasus, nous avons publié une méthode décrivant comment nous avons pu traquer Pégasus sur plusieurs années [13], ainsi qu'un outil appelé le Mobile Verification Toolkit [16] (MVT pour les intimes).

Notre méthode d'analyse d'iPhone se déroule en quatre étapes :

- faire un backup chiffré du téléphone à l'aide d'iTunes ou de libimobiledevice [1]
- récupérer une partie des données de ce backup pour l'analyser
- déchiffrer le backup
- analyser les données avec le Mobile Verification Toolkit.

Si la méthodologie décrite ici correspond bien à celle utilisée pendant le projet Pégasus, plusieurs artefacts découverts depuis ont été ajoutés à cet article afin de fournir une méthodologie plus exhaustive.

4.1 Anatomie d'un backup

La recherche d'artefacts nous a tout d'abord demandé de comprendre la structure de fichiers d'un backup. Un backup est constitué de :

- un fichier *Info.plist* contenant des informations sur les applications installées
- un fichier *Manifest.plist* contenant des informations sur le backup, notamment les clés de chiffrement si celui-ci est chiffré
- un fichier *Status.plist* contenant des informations sur le backup lui-même (date, UUID)
- un fichier *Manifest.db* regroupant des informations sur la liste des fichiers présents sur le backup, notamment leur chemin et domaine
- un certain nombre de fichiers nommés par leur hash, et triés par dossiers nommés à partir des deux premiers caractères du hash. Par exemple, le fichier *DataUsage.sqlite* que nous verrons plus loin, a pour hash *0d609c54856a9bb2d56729df1d68f2958a88426b* et est donc stocké dans *0d/0d609c54856a9bb2d56729df1d68f2958a88426b*.

Un backup chiffré contient plus d'informations, notamment privées, comme les données de l'application WhatsApp.

4.2 Artefacts forensiques pour iOS

Une fois le format de ce backup analysé, nous avons pu petit à petit découvrir des artefacts utiles à la détection de logiciels malveillants sur le téléphone.

DataUsage.sqlite Le fichier *DataUsage.sqlite* présent dans le backup (id : *0d609c54856a9bb2d56729df1d68f2958a88426b*) est une base de données stockant des informations relatives à l'utilisation du réseau de données (2/3/4/5G) par application afin d'identifier des applications gourmandes en données communiquées. Il s'agit en réalité d'une mine d'or pour l'analyse,

car il garde trace de tous les processus exécutés sur le téléphone depuis la dernière réinitialisation. Il est même restauré en cas de sauvegarde, ce qui nous a permis par exemple de trouver des traces d'infections sur des téléphones n'étant plus en possession de leur propriétaire.

Cette base de données SQLite a deux tables principales, une table *ZPROCESS* contenant des listes de processus avec leur domaine et leurs dates de premières et dernières utilisations. Une seconde table *ZLIVEUSAGE* contient la quantité de données utilisée, ainsi qu'une date d'utilisation.

Voici par exemple une entrée de *DataUsage* extraite par MVT :

```
1 {
2   "first_isodate": "2019-08-29 23:23:13.935593",
3   "isodate": "2019-08-29 23:23:20.705853",
4   "proc_name": "mDNSResponder/com.apple.AppStore",
5   "bundle_id": "com.apple.AppStore",
6   "proc_id": 38,
7   "wifi_in": 0.0,
8   "wifi_out": 0.0,
9   "wwan_in": 437.0,
10  "wwan_out": 138.0,
11  "live_id": 7,
12  "live_proc_id": 38,
13  "live_isodate": "2019-08-29 23:23:13.933432"
14 },
```

Listing 1. Extrait d'un fichier *DataUsage*

Cette entrée nous indique que le 29 août 2019, le processus *mDNSResponder/com.apple.AppStore* qui appartient à l'application *com.apple.AppStore* a envoyé 138 octets de données et reçu 437 octets. Le *bundle_id* est important ici : il nous indique qu'il s'agit d'un processus appartenant à une application et non ayant des droits système élevés. Dans certains cas, un bug dans iOS fait que seuls les 8 premiers caractères du nom de processus sont enregistrés.

Au cours de notre enquête, nous avons identifié 75 processus [17] utilisés par différents composants de Pégasus, le plus souvent ayant des noms proches de processus système, comme par exemple *seraccountd* ou *rlaccountd*.

Nous avons également remarqué que certaines versions de Pégasus récentes suppriment leurs noms de processus de cette base de données. Dans certains cas, cette suppression se fait de manière incomplète, seule l'entrée dans la table *ZPROCESS* est supprimée, laissant une entrée dans la table *ZLIVEUSAGE* orpheline. D'expérience, nous avons trouvés de très rares cas dans lesquels ce phénomène se produit sans infection de Pégasus

(probablement en raison d'un bug lors d'une restauration de sauvegarde), il s'agit donc d'un indicateur important de la présence possible de Pégasus mais qui ne suffit pas à démontrer une infection.

Enfin, certaines versions de Pégasus suppriment correctement les processus en rapport avec Pégasus de ces deux tables. Néanmoins, les identifiants des entrées de la table étant incrémentaux, il est possible de voir qu'il y a eu des suppressions. On peut alors retrouver une période de temps d'exécution du processus supprimé entre la première date d'exécution du processus précédent et du suivant. Cet indicateur est utile mais cependant pas totalement fiable, car nous avons remarqué des processus légitimes supprimés automatiquement par le téléphone de manière assez régulière.

Manifest.db Comme nous l'avons vu précédemment, la base de données *Manifest.db* est un fichier central des backups, puisqu'il contient la liste des fichiers présents dans le backup. Mais il contient en réalité des informations extrêmement utiles à une analyse, notamment le domaine de création des fichiers, ainsi que les dates de création, modification et changement de statut.

C'est grâce à cet artefact que nous avons pu identifier la création de certains fichiers par Pégasus lors d'une infection, par exemple le fichier *Library/Preferences/com.apple.CrashReporter.plist* qui permet de désactiver l'envoi de rapports de crash à Apple, ou encore le fichier *Library/Preferences/roleaccountd.plist*. Dans les deux cas, ces fichiers sont créés en *RootDomain*, soit le niveau de privilège du système.

Voici un exemple de fichier créé par Pégasus et listé dans la base *Manifest.db* et extrait par MVT :

```
1 {
2   "file_id": "6edc4862c937e60d235878f03a201e11de26b642",
3   "domain": "RootDomain",
4   "relative_path": "Library/Preferences/roleaccountd.plist",
5   "flags": 1,
6   "created": "2019-04-04 05:33:12.000000",
7   "modified": "2019-04-04 05:33:12.000000",
8   "status_changed": "2019-12-18 22:14:22.000000",
9   "mode": "0o100600",
10  "owner": 0,
11  "size": 262
12 },
```

Listing 2. Extrait de données d'un fichier Manifest.db

IDStatusCache Le fichier IDStatusCache présent dans les backups (id *6b97989189901ceaa4e5be9b7f05fb584120e27b*) répertorie les recherches de

comptes iCloud faites par des applications. Il enregistre par exemple lorsque iMessage vérifie qu'un numéro de téléphone est bien associé à un compte iCloud. Lors de notre enquête, nous avons découvert qu'il contenait des traces de recherches de comptes iCloud utilisés par NSO Group afin d'exploiter des vulnérabilités dans des applications.

Voici un exemple d'une entrée dans ce fichier extraite par MVT montrant une recherche de compte que nous attribuons à Pégasus quelques secondes avant l'exécution d'un processus lié à Pégasus (le package *com.apple.madrid* correspond à l'application iMessage) :

```
1 {  
2   "package": "com.apple.madrid",  
3   "user": "mailto:emmadavies8266@gmail.com",  
4   "isodate": "2019-09-10 06:09:04.634913",  
5   "idstatus": 1,  
6   "occurrences": 1  
7 },
```

Dans certains cas, l'e-mail utilisé par Pégasus a deux caractères remplacés par l'octet 0, comme par exemple *e\x00\x00adavies8266@gmail.com* au lieu de *emmadavies8266@gmail.com*. Nous n'avons jamais observé de caractère nul dans des entrées légitimes, cette modification a pu venir soit d'une tentative d'obfuscation, soit d'un effet secondaire de l'exploitation.

Lors de notre enquête, nous avons remarqué que ces comptes iCloud (nous en avons listé 17 durant le projet Pégasus [17]) semblent être spécifiques à un client de NSO Group. Par exemple, nous n'avons vu l'adresse email *emmadavies8266@gmail.com* ci-dessus que sur les téléphones d'András Szabó et Szabolcs Panyi, deux journalistes hongrois (vous pouvez vous reporter à l'annexe B de notre méthodologie [13] pour plus d'informations).

Pour des raisons assez obscures, cet artefact a été supprimé des backups par Apple dans la version iOS 14.8, quelques semaines seulement après la publication du projet Pégasus, nous privant d'un artefact précieux lors de nos analyses.

OS Analytics AD Daily Un fichier nommé *com.apple.osanalytics.addaily.plist* contient une liste des processus lancés sur le téléphone ainsi que la quantité de données utilisée en Wifi et en Data. Découvert après la publication du projet Pégasus et disponible dans les backups (id *f65b5fafc69bbd3c60be019c6e938e146825fa83*), il constitue un artefact important pour compléter la liste des processus disponibles dans le fichier DataUsage. Seul bémol : il ne contient pas d'information sur le bundle du processus permettant de déterminer si le processus était lancé en tant que système ou par une application.

```
1 {
2   "package": "healthappd",
3   "ts": "2021-03-24 12:00:38.452758",
4   "wifi_in": 685214.0,
5   "wifi_out": 221935.0,
6   "wwan_in": 0.0,
7   "wwan_out": 0.0
8 },
```

Listing 3. Exemple de données extraites de `com.apple.osanalytics.addaily.plist`

SMS et messages WhatsApp Comme indiqué précédemment, une partie des attaques de Pegasus utilise des liens envoyés par SMS, il est donc utile de regarder les messages reçus par SMS et autres applications de messagerie. Les données de SMS sont disponibles dans le fichier `private/var/mobile/Library/SMS/sms.db` qui figure dans les backups sous l'id `3d0d7e5fb2ce288813306e4d4636395e047a3d28`. Les données WhatsApp sont elles disponibles dans les fichiers `private/var/mobile/Containers/Shared/AppGroup/*/ChatStorage.sqlite` présents dans les backups sous l'id `7c7fba66680ef796b916b067077cc246adacf01d` (se reporter au code source de MVT pour voir le détail des requêtes SQL).

Historique de navigation L'historique de navigation est évidemment une information importante pour identifier soit des clics sur un lien envoyé par SMS, soit une potentielle attaque par injection de trafic. Le navigateur Safari est installé et utilisé par défaut sur iPhone mais Chrome ou Firefox sont assez régulièrement utilisés, il est donc utile de récupérer les données de ces trois navigateurs.

Les historiques de navigation de Chrome, Firefox et Safari sont inclus dans les backups iOS :

- pour Safari dans le fichier `Library/Safari/History.db` (l'id change en fonction des appareils)
- pour Chrome dans le fichier `Library/Application Support/Google/Chrome/Default/History` (id `faf971ce92c3ac508c018dce1bef2a8b8e9838f1`)
- pour Firefox dans le fichier `private/var/mobile/profile.profile/browser.d` (id `2e57c396a35b0d1bcbc624725002d98bd61d142b`).

Ces données ne sont en général stockées que pour une durée limitée (quelques mois tout au plus pour Safari par exemple).

Dans plusieurs analyses que nous avons faites, nous n'avons pas eu accès aux informations de l'historique au moment de l'infection en raison du délai

entre l'infection et l'analyse. Cependant les navigateurs gardent d'autres traces de la navigation et notamment un historique des favicons (icônes de sites web) téléchargés. Si cette donnée ne donne qu'une vue partielle de la navigation, elle est par contre conservée pendant une plus longue période et constitue un artefact additionnel de l'historique de navigation. Malheureusement, le fichier Favicon de Safari n'est pas disponible dans les backups et uniquement accessible par Jailbreak :

- pour Safari : dans les fichiers *private/var/mobile/Library/Image Cache/Favicons/Favicons.db* ou *private/var/mobile/Containers/Data/Application/*/Library/Image Cache/Favicons/Favicons.db* par jailbreak
- pour Chrome : *Library/Application Support/Google/Chrome/Default/Favicons* (id *55680ab883d0fdcffd94f959b1632e5fbbb18c5b*)
- pour Firefox : *profile.profile/browser.db* (id : *2e57c396a35b0d1bc6c624725002d98bd61d142b*).

LocationD Le fichier *locationd/clients.plist* contient des informations sur les applications ayant demandé à utiliser la géolocalisation du téléphone, incluant des process système potentiellement malveillants. Il s'agit donc d'un artefact utile pour identifier une infection. Ce fichier est présent dans les backups sous l'id *a690d7769cce8904ca2b67320b107c8fe5f79412* et comprend la date de début d'utilisation de la géolocalisation.

```

1  {
2      "Whitelisted": false,
3      "SupportedAuthorizationMask": 5,
4      "BundlePath": "/System/Library/PrivateFrameworks/
      FindMyDevice.framework",
5      "AuthorizationUpgradeAvailable": false,
6      "Authorization": 4,
7      "Registered": "",
8      "Executable": "",
9      "ConsumptionPeriodBegin": "2021-10-06 12:47:09.339589",
10     "package": "com.apple.locationd.bundle-/System/Library/
      PrivateFrameworks/FindMyDevice.framework"
11 },

```

Listing 4. Exemple de données extraites de clientsplist

Raccourcis Le malware Prédator développé par l'entreprise Cytrox et analysé par le Citizen Lab en décembre 2021 [24] utilise des raccourcis iOS pour être persistant au redémarrage du téléphone. Un raccourci est créé sur le téléphone avec une automatisation pour activer cette tâche lors du lancement d'une application sur le téléphone. Cette tâche télécharge

alors du code javascript depuis un serveur géré par Cytrox et l'exécute sur le téléphone.

Les informations sur les raccourcis présents sur le téléphone sont stockées dans le fichier *private/var/mobile/Library/Shortcuts/Shortcuts.sqlite* qui est présent dans les backups (id *5b4d0b44b5990f62b9f4d34ad8dc382bf0b01094*). Il s'agit donc d'un indicateur utile pour identifier ce type de persistance.

```
1  {
2      "shortcut_id": 2,
3      "shortcut_name": "Automation 71F559AF-A383-46AA-8A14-
4          D4D82C95139F",
5      "modified_date": "2022-04-02 16:55:11.325908",
6      "description": "Open URLs",
7      "isodate": "2022-04-02 16:44:23.281346",
8      "parsed_actions": 1,
9      "action_urls": [
10         "https://amnesty.org"
11     ]
12 }
```

Listing 5. Exemple de données extraites d'un raccourci

Profils de configuration Enfin, le malware Prédator installe également un profil de configuration sur le téléphone [24]. Les profils de configurations permettant d'accéder à de nombreuses fonctionnalités sur un iPhone, il s'agit d'un artefact utile pour identifier des attaques (avec ou sans malware). Les données sur les profils de configurations sont stockées dans le dossier *Library/ConfigurationProfiles/* et disponibles dans les backups sous le domaine *SysSharedContainerDomain-systemgroup.com.apple.configurationprofiles*.

5 Le Mobile Verification Toolkit

Afin de simplifier l'analyse de backups, nous avons développé et publié un logiciel : le Mobile Verification Toolkit [16] (ci-après appelé MVT).

MVT permet à la fois d'aider à l'analyse d'un téléphone (par exemple en déchiffrant un backup d'iPhone), d'extraire les données ainsi que d'identifier de potentielles traces malveillantes à partir d'indicateurs de compromission (IOC).

L'analyse d'artefacts est organisée en modules, comme le module *SMS* ou *OSAnalyticsADDaily*. Les modules sont regroupés par type d'analyse (par exemple les modules pour backup) et sont lancés les uns après les

autres lors d'une l'analyse. Chaque module peut réaliser trois tâches : tout d'abord extraire des informations d'un artefact (ces informations peuvent être stockées en JSON par la suite), ensuite transformer ces résultats sous la forme de données datées permettant l'établissement d'une chronologie, et enfin vérifier la présence d'indicateurs de compromission.

Des indicateurs de compromissions (IOC) peuvent être transmises à MVT sous la forme de fichiers au format STIX2 [9], regroupant par exemple des domaines, adresses email, hashes ou noms de processus. Une commande *download-iocs* permet de télécharger plusieurs fichiers existants (dont un pour Pégasus) et de les précharger lors d'une analyse.

Il faut noter que la licence de MVT ne permet son utilisation qu'avec le consentement du ou de la propriétaire du téléphone. Même si le forensique a des utilisations légitimes, nous connaissons aussi la menace qu'il fait peser sur de nombreux DDH lors de saisies de leur matériel informatique par des agences étatiques, et avons voulu limiter l'utilisation de MVT dans ce cadre-là.

Voici à quoi pourrait ressembler l'analyse d'un iPhone avec MVT et libimobiledevice (les logs ont été raccourcis pour plus de visibilité) :

```

1  $ idevicebackup2 backup -d FOLDER
2  Backup directory is "backup3"
3  Started "com.apple.mobilebackup2" service on port 49257.
4  Negotiated Protocol Version 2.1
5  Starting backup...
6  Backup will be encrypted.
7  Requesting backup from device...
8  Full backup mode.
9  [=====] 18% Finished
10 Receiving files
11 [=====] 28% Finished
    /23.9 MB)
12 Receiving files
13 [=====] 58% Finished
    /84.8 MB)
14 [SNIP]
15 Sending 'c4ef0ab5293610788fd86e641ef9cd8f072c4b08/Status.plist' (189
    Bytes)
16 Sending 'c4ef0ab5293610788fd86e641ef9cd8f072c4b08/Manifest.plist'
    (80.9 KB)
17 Sending 'c4ef0ab5293610788fd86e641ef9cd8f072c4b08/Manifest.db' (3.3
    MB)
18 Received 1106 files from device.
19 Backup Successful.

```

Listing 6. Extraction du backup avec libimobiledevice

```

1  $ mvt-ios decrypt-backup -p [PASSWORD] -d decrypted FOLDER
2  MVT - Mobile Verification Toolkit

```



```

3           https://mvt.re
4           Version: 1.4.9
5
6 INFO      [mvt.ios.cli] Your password may be visible in the process
7           table because it was supplied on the command line!
8 INFO      [mvt.ios.decrypt] Decrypting iOS backup at path backup2
9           with password
10 INFO     [mvt.ios.decrypt] Decrypted file Library/Cookies/Cookies.
           binarycookies [AppDomain-ch.icoaching.typewise] to decrypted2
           /14/14cba36499f91a10ab77753fc7c5b36ce587320e
11 INFO     [mvt.ios.decrypt] Decrypted file Library/Preferences/ch.
           icoaching.typewise.plist [AppDomain-ch.icoaching.typewise] to
           decrypted2/22/222294a11b55fb6581548ecd803be1af959c4295
12 [SNIP]

```

Listing 7. Déchiffrement du backup avec MVT

```

1 $ mvt-ios check-backup -o results decrypted
2
3           MVT - Mobile Verification Toolkit
4           https://mvt.re
5           Version: 1.4.9
6
7 INFO      [mvt.ios.cli] Checking iTunes backup located at: decrypted
8 INFO      [mvt.ios.cli] Parsing STIX2 indicators file at path [PATH]/
           raw.githubusercontent.com_AmnestyTech_investigations_master_2021
           -07-18_nso_pegasus.stix2
9 INFO      [mvt.ios.cli] Loaded 1499 indicators from "Pegasus"
           indicators file
10 INFO     [mvt.ios.cli] Parsing STIX2 indicators file at path [PATH]/
           raw.githubusercontent.com_AmnestyTech_investigations_master_2021
           -12-16_cytrox_cytrox.stix2
11 INFO     [mvt.ios.cli] Loaded 330 indicators from "Cytrox"
           indicators file
12 INFO     [mvt.ios.cli] Loaded a total of 1829 unique indicators
13 INFO     [mvt.ios.modules.backup.backup_info] Running module
           BackupInfo...
14 INFO     [mvt.ios.modules.backup.backup_info] Build Version: 18C66
15 INFO     [mvt.ios.modules.backup.backup_info] Device Name: iPhone
16 [SNIP]
17 INFO     [mvt.ios.modules.backup.manifest] Running module Manifest
           ...
18 INFO     [mvt.ios.modules.backup.manifest] Found Manifest.db
           database at path: decrypted/Manifest.db
19 INFO     [mvt.ios.modules.backup.manifest] Extracted a total of 3738
           file metadata items
20 INFO     [mvt.ios.modules.backup.manifest] The Manifest module
           produced no detections!
21 INFO     [mvt.ios.modules.mixed.osanalytics_addaily] Running module
           OSAnalyticsADDaily...
22 INFO     [mvt.ios.modules.mixed.osanalytics_addaily] Found com.apple
           .osanalytics.addaily plist at path: decrypted/f6/
           f65b5fafc69bbd3c60be019c6e938e146825fa83
23 INFO     [mvt.ios.modules.mixed.osanalytics_addaily] Extracted a
           total of 120 com.apple.osanalytics.addaily entries

```

```
24 WARNING [mvt.ios.modules.mixed.osanalytics_addaily] Found a known
    suspicious process name "actmanaged" matching indicators from "
    Pegasus"
25 [SNIP]
26 WARNING [mvt.ios.cli] The analysis of the backup produced 1
    detections!
```

Listing 8. Analyse du backup avec MVT

On voit dans les logs de cette commande que MVT charge tout d’abord un certain nombre d’indicateurs (1829 ici), puis commande l’analyse qui se fait module par module. Le premier module par exemple, *backup_info*, nous donne des indications sur le téléphone (nom, IMEI, applications installées, etc.). Les modules suivants continuent leur exécution jusqu’au module *osanalytics_daily* qui indique avoir identifié un processus nommé *actmanaged* et connu dans les indicateurs comme étant un processus de Pégasus.

À la suite de cette analyse, le dossier *results* contient un certain nombre de fichiers JSONs créés par les différents modules, ainsi qu’une chronologie des évènements dans le fichier *timeline.csv*. Si un indicateur a été détecté par un module, il sera alors copié dans un fichier JSON distinct terminé par *_detected* (par exemple *datausage_detected.json*) et reporté dans un fichier *timeline_detected.csv*.

6 Résultats

Lors du projet Pégasus, nous avons analysé 67 téléphones de journalistes ou DDH, et identifié des traces de Pégasus sur 37 d’entre eux. Ce chiffre peut paraître assez bas mais il faut bien comprendre qu’un certain nombre de ces téléphones étaient des téléphones Android pour lequel l’analyse est beaucoup moins fiable, et qu’une partie des personnes avaient changé de téléphone depuis leur attaque potentielle. Si nous ne comptons que les iPhone qui étaient déjà utilisés au moment d’une attaque potentielle, nous avons alors trouvé des traces de Pégasus sur plus de 75 % d’entre eux.

Nous avons publié en annexe de notre méthodologie [14] une liste des traces d’infection trouvées sur ces différents téléphones, issues directement des chronologies générées par MVT. Le tableau 1 montre le détail des traces forensiques trouvées sur le téléphone d’András Szabó, journaliste hongrois au média Direkt36. On retrouve dans ce tableau les artefacts décrits plus haut : création de fichiers connus pour être des traces de Pégasus, suppression d’entrées dans la table *ZPROCESS* de la base de

données *DataUsage.sqlite*, ou encore recherche de comptes iCloud par iMessage venant du fichier *IDStatusCache*.

Date (UTC)	Event
2019-06-13 11 :15 :40	File created : Library/Preferences/com.apple.CrashReporter.plist from RootDomain
2019-06-13 11 :15 :53	File created : Library/Preferences/roleaccountd.plist from RootDomain
2019-06-13 12 :39 :40	Process record deleted from ZPROCESS (IN : 3.69 MB, OUT : 27.39 MB)
2019-06-15 08 :06 :27	Process record deleted from ZPROCESS (IN : 0.32 MB, OUT : 0.56 MB)
2019-07-25 09 :31 :09	Process record deleted from ZPROCESS (IN : 7.80 MB, OUT : 6.43 MB)
2019-08-16 10 :13 :19	Process record deleted from ZPROCESS (IN : 18 MB, OUT : 29.81 MB)
2019-09-15 15 :30 :44	Process record deleted from ZPROCESS (IN : 1.27 MB, OUT : 3.34 MB)
2019-09-17 06 :33 :24	Process record deleted from ZPROCESS (IN : 2.00 MB, OUT : 5.57 MB)
2019-09-24 13 :26 :15	iMessage lookup for account jessicadavies1345@outlook.com
2019-09-24 13 :26 :51	iMessage lookup for account emmadavies8266@gmail.com
2019-09-24 13 :32 :10	Process : roleaccountd (IN : 0.02 MB, OUT : 0.003 MB)
2019-09-24 13 :32 :11	Process : roleaccountd
2019-09-24 13 :32 :13	Process : stagingd (IN : 4.03 MB, OUT : 0.19 MB)
2019-09-24 13 :32 :23	Process : stagingd
2019-09-26 14 :32 :25	Process record deleted from ZPROCESS (IN : 1.16 MB, OUT : 2.81 MB)
2019-10-24 05 :40 :33	Process record deleted from ZPROCESS (IN : 12.81 MB, OUT : 46 MB)

Tableau 1. Détails des traces d’infection sur le téléphone du journaliste hongrois András Szabó

Sur les traces de 41 téléphones publiées en juillet 2021, 27 contiennent des traces venant de *idstatuscache*, 23 contiennent des traces venant de *DataUsage*, 22 venant de *Manifest.db* et 8 venant de SMS (le fichier OS Analytics Daily, les raccourcis et profils de configurations n’ont pas été analysés durant le projet).

7 Conclusion

Dans cet article, nous avons vu en détail la méthodologie de forensique que nous avons développée au sein du Security Lab d’Amnesty International pendant le projet Pégasus. Au-delà des aspects techniques,

ces révélations ont mis en lumière les abus incessants contre les DDH et journalistes rendus possibles par l'industrie de la surveillance dont NSO Group est la figure centrale. Du Mexique à l'Inde, en passant par la Hongrie ou le Maroc, les outils de NSO Group ont été constamment utilisés pour cibler et pirater des journalistes ou militant·es luttant pour les droits humains dans des contextes extrêmement difficiles.

Ces révélations doivent maintenant entraîner des actions de la part des États, les organisations internationales et les grandes entreprises de la technologie. L'ajout par le département du commerce états-uniens de NSO Group et Candiru sur une liste des entités connues pour des activités malveillantes [28], le procès intenté par Apple [2] contre NSO Group ainsi que les notifications envoyées aux personnes ciblées sont des évolutions extrêmement positives dans la lutte contre ces abus. On ne trouvera néanmoins pas de solution pérenne sans accords internationaux sur ce sujet. Amnesty, comme d'autres organisations, appelle à un moratoire sur l'utilisation, la vente et le transfert de technologies de surveillance jusqu'à ce qu'un cadre réglementaire approprié en matière de droits humains soit mis en place [18].

Au sein de nos communautés de sécurité informatique, où la limite entre sécurité offensive et surveillance peut-être poreuse, il nous faudra également questionner le rôle que nous jouons en tant que hacker·euses et ingénieur·es dans le développement de telles technologies.

Références

1. libimobiledevice. <https://libimobiledevice.org/>.
2. Apple. Apple sues nso group to curb the abuse of state-sponsored spyware. <https://www.apple.com/newsroom/2021/11/apple-sues-nso-group-to-curb-the-abuse-of-state-sponsored-spyware/>, 2021.
3. Bloomberg. Spyware vendor finfisher claims insolvency amid investigation. <https://www.bloomberg.com/news/articles/2022-03-28/spyware-vendor-finisher-claims-insolvency-amid-investigation>, 2022.
4. Security Without Borders. Reports on Targeted Surveillance of Civil Society. <https://securitywithoutborders.org/resources/targeted-surveillance-reports.html>, 2021.
5. NSO Group. Pegasus - product description. <https://www.documentcloud.org/documents/4599753-NSO-Pegasus.html>, 2012.
6. The Guardian. Whatsapp sues israeli firm, accusing it of hacking activists' phones. <https://www.theguardian.com/technology/2019/oct/29/whatsapp-sues-israeli-firm-accusing-it-of-hacking-activists-phones>, 2019.
7. Claudio Guarnieri. A Primer On Android Forensics. <https://nex.sx/tech/2022/01/28/a-primer-on-android-forensics.html>, 2022.

8. Claudio Guarnieri. Diving Deeper in Android System Diagnostics and Remote Forensics. <https://nex.sx/tech/2022/02/04/diving-deeper-in-android-system-diagnostics.html>, 2022.
9. OASIS Cyber Threat Intelligence. Introduction to stix. <https://oasis-open.github.io/cti-documentation/stix/intro.html>, 2021.
10. Amnesty International. Amnesty International Among Targets of NSO-powered Campaign. <https://www.amnesty.org/en/latest/research/2018/08/amnesty-international-among-targets-of-nso-powered-campaign/>, 2018.
11. Amnesty International. Morocco : Human Rights Defenders Targeted with NSO Group’s Spyware. <https://www.amnesty.org/en/latest/research/2019/10/Morocco-Human-Rights-Defenders-Targeted-with-NSO-Groups-Spyware/>, 2019.
12. Amnesty International. Moroccan Journalist Targeted With Network Injection Attacks Using NSO Group’s Tools. <https://www.amnesty.org/en/latest/research/2020/06/moroccan-journalist-targeted-with-network-injection-attacks-using-nso-groups-tools/>, 2020.
13. Amnesty International. Forensic Methodology Report : How to catch NSO Group’s Pegasus. <https://www.amnesty.org/en/latest/research/2021/07/forensic-methodology-report-how-to-catch-nso-groups-pegasus/>, 2021.
14. Amnesty International. Forensic Methodology Report : Pegasus Forensic Traces per Target. <https://www.amnesty.org/en/latest/research/2021/07/forensic-methodology-report-appendix-d/>, 2021.
15. Amnesty International. Massive data leak reveals Israeli NSO Group’s spyware used to target activists, journalists, and political leaders globally. <https://www.amnesty.org/en/latest/news/2021/07/the-pegasus-project/>, 2021.
16. Amnesty International. Mobile verification toolkit. <https://github.com/mvt-project/mvt>, 2021.
17. Amnesty International. NSO Group Pegasus Indicator of Compromise. https://github.com/AmnestyTech/investigations/tree/master/2021-07-18_nso, 2021.
18. Amnesty International. Demandez la fin de la surveillance ciblée des défenseur-e-s des droits humains. <https://www.amnesty.org/fr/petition/targeted-surveillance-human-rights-defenders/>, 2022.
19. Citizen Lab. From bahrain with love - finfisher’s spy kit exposed? <https://citizenlab.ca/2012/07/from-bahrain-with-love-finishers-spy-kit-exposed/>, 2012.
20. Citizen Lab. Communities @ risk - targeted digital threats against civil society. <https://targetedthreats.net/>, 2014.
21. Citizen Lab. The million dollar dissident - nso group’s iphone zero-days used against a uae human rights defender. <https://citizenlab.ca/2016/08/million-dollar-dissident-iphone-zero-day-nso-group-uae/>, 2016.
22. Citizen Lab. Bitter sweet - supporters of mexico’s soda tax targeted with nso exploit links. <https://citizenlab.ca/2017/02/bittersweet-nso-mexico-spyware/>, 2017.
23. Citizen Lab. Forcentry - nso group imessage zero-click exploit captured in the wild. <https://citizenlab.ca/2021/09/forcentry-nso-group-imessage-zero-click-exploit-captured-in-the-wild/>, 2021.

24. Citizen Lab. Pegasus vs. predator - dissident's doubly-infected iphone reveals cytrox mercenary spyware. <https://citizenlab.ca/2021/12/pegasus-vs-predator-dissidents-doubly-infected-iphone-reveals-cytrox-mercenary-spyware/>, 2021.
25. Lookout. Technical analysis of pegasus spyware. <https://info.lookout.com/rs/051-ESQ-475/images/lookout-pegasus-technical-analysis.pdf>, 2016.
26. Lookout. Pegasus for android - technical analysis and findings of chrysaor. <https://info.lookout.com/rs/051-ESQ-475/images/lookout-pegasus-android-technical-analysis.pdf>, 2017.
27. Le Monde. Affaire khashoggi : deux femmes proches du journaliste assassiné ont été surveillées par pegasus. https://www.lemonde.fr/projet-pegasus/article/2021/07/18/affaire-khashoggi-deux-femmes-proches-du-journaliste-assassine-ont-ete-surveillees-par-pegasus_6088655_6088648.html, 2021.
28. US Department of Commerce. Commerce adds nso group and other foreign companies to entity list for malicious cyber activities. <https://www.commerce.gov/news/press-releases/2021/11/commerce-adds-nso-group-and-other-foreign-companies-entity-list>, 2021.
29. Google Security. An investigation of chrysaor malware on android. <https://security.googleblog.com/2017/04/an-investigation-of-chrysaor-malware-on.html>, 2017.
30. Slate. How government-grade spy tech used a fake scandal to dupe journalists. <https://slate.com/technology/2012/08/moroccan-website-mamfakinch-targeted-by-government-grade-spyware-from-hacking-team.html>, 2012.
31. Forbidden Stories. Spying on Mexican Journalists : Investigating the Lucrative Market of Cyber-Surveillance. <https://forbiddenstories.org/spying-on-mexican-journalists-investigating-the-lucrative-market-of-cyber-surveillance/>, 2020.
32. Forbidden Stories. Le projet Pégasus. <https://forbiddenstories.org/fr/case/le-pegasus-project/>, 2021.
33. VICE. An Interview with Hacker Phineas Fisher as a Puppet. <https://www.youtube.com/watch?v=BpyC11Qm6Xs>, 2016.