Bureau d'Études TLS-SEC
BE Empoisonnement ARP

Avant-Propos

Une grande partie des idées et du texte de ce TP sont tirés de :
`[1] http://sid.rstack.org/static/articles/j/o/u/Jouer_avec_le_protocole_ARP_dadb.html`
page web réalisée par Cédric Blancher, hacker et expert en sécurité très reconnu dans la communauté française décédé subitement en 2013.

1 Introduction

L'objectif de ce TP c'est d'étudier et mettre en place les attaques de type empoisonnement ARP. Pour ce faire on se placera dans le contexte proposé par Cédric Blancher dans [1]. On jouera le rôle d'un ordinateur appelé joker voulant réaliser différentes attaques dans un réseau ou se trouvent les ordinateurs batman et robin. La passerelle s'appelle batcave-gw.

N'initiez aucune communication même pas un ping tant qu'on ne vous le demande pas.

1.1) Définissez un plan d'adressage, configurez la passerelle et attribuez des adresses IP aux différents acteurs.

1.2) Ajoutez les lignes nécessaires au fichier `/etc/hosts` pour utiliser les noms de domaine batman, robin, joker et batcave-gw dans vos commandes au lieu des adresses IP.

2 Étude du cache arp

2.1) Affichez les contenus du cache de la machine batman en tapant sur un terminal `arp -a`. Normalement il devrait être vide.

2.2) Faites un ping vers robin. Affichez à nouveau le contenu du cache arp. Normalement une entrée faisant la correspondance entre l'adresse IP de robin et son adresse MAC doit apparaître.

Il est également possible d'afficher la table ARP par la commande `ip neigh show`. Cette commande donne en plus l'état de l'entrée (REACHABLE, DELAY, STALE, INCOMPLETE, PROBE).

- INCOMPLETE - On est en attente d'une résolution d'adresse classique qui a été initiée mais pas complétée.
- REACHABLE - L'information dans la table est d'actualité, le voisin en question a été atteint il y a au plus quelques dizaines de secondes).
- STALE - L'entrée est obsolète on ne tentera pas de résoudre l'adresse tant que du trafic ne sera pas a envoyer à nouveau.
- DELAY - L'entrée est obsolète mais du trafic a été envoyé récemment au voisin et on est en attente de voir si le noyau arrive à voir que les messages sont reçus correctement (auquel cas on passera à l'état REACHABLE) ou pas (auquel cas on enverra une requête ARP en unicast et on passera à l'état PROBE).
- PROBE - L'entrée est obsolète et des requêtes ARP en unicast ont été envoyées. Si on a une réponse on passe en REACHABLE et sinon on envoie des requêtes ARP en broadcast et passe en INCOMPLETE.

2.3) Utilisez la commande `arp -d` suivie de l'adresse IP de robin pour effacer l'entrée associée dans la table.

Vous remarquerez que l'entrée ne disparaît pas, elle passe à l'état FAILED. Le noyau linux fait tout ce qu'il peut pour ne pas effacer les informations ARP car elles sont considérées précieuses. L'entrée ne disparaîtra complètement qu'au bout de quelques minutes (au plus une dizaine).

Une entrée, même en état FAILED est présente dans la table et donc les requêtes ARP la concernant sont considérées comme des actualisations d'entrées et pas comme des créations. Ceci a une importance en sécurité puisque les réponses ARP non sollicitées sont ignorées pour une création mais pas pour une actualisation. Quand vous voudrez faire des tests sur des créations d'entrées il ne suffira pas d'effacer une entrée avec `arp -d`, il faudra travailler sur une IP absente du cache ou attendre dix minutes pour réussir à supprimer une entrée pour de vrai.

2.4) Faites un ping vers robin depuis batman. Affichez la table arp. Déconfigurez l'interface réseau puis reconfigurez la. Affichez la table arp. Qu'est que vous en déduisez ?

3 Utilisation de arp-sk

Vous pouvez consulter le manuel de cette commande avec `arp-sk --help`. Avec les idées vues en cours et cette documentation on vous demande de commencer par faire quelques empoisonnements de base pour vous entraîner.

Empoisonnement par actualisation

3.1) Faites un ping depuis batman vers robin. Écoutez depuis joker, normalement vous ne devriez voir rien passer.

3.2) Utilisez arp-sk depuis joker avec l'option -r pour corrompre le cache de batman et associer à l'IP de robin l'adresse MAC de joker.

3.3) Refaites un ping depuis batman et écoutez depuis joker. Normalement joker devrait voir les ping request arriver mais batman ne devrait pas avoir des ping reply. Pourquoi ?

3.4) Activez le routage sur joker. Recommencez le ping et l'écoute. Ça devrait marcher maintenant. Essayez de vous expliquer l'un l'autre pourquoi le routage arrange les choses.

Amélioration de l'attaque

À ce point l'attaque ne marche qu'à moitié. Quand vous faites un ping de batman vers robin, vous avez des alertes qui s'affichent (parlant d'une redirection). Ces alertes apparaissent régulièrement au début puis de moins en moins souvent. Elles sont dues à des messages ICMP redirect qui demandent à batman de ne pas passer par joker pour atteindre robin.

3.5) Observez le trafic au niveau de batman. Qui est à l'origine de ces messages ICMP ? Pourquoi ces messages sont envoyés ? Regardez comment varie l'adresse MAC destination des ping request de batman en fonction des messages ICMP et des messages ARP reçus. Essayez de comprendre les changements et de vous les expliquer entre vous.

Ces messages sont très pénalisants pour notre attaque. On peut bloquer leur envoi sur une interface en ajoutant une règle à iptables, ou en mettant à zéro `/proc/sys/net/ipv4/conf/{all,ethX}/send_redirects`, ethX étant l'interface par laquelle on ne veut pas qu'ils soient envoyés.

3.6) Modifiez la configuration de joker pour que les envois de messages ICMP redirect cessent. Refaites un empoisonnement et un ping. Il ne devrait plus y avoir d'alerte et pratiquement tous les pings devraient être envoyés à batman.

3.7) Cherchez dans une capture d'une minute au niveau de batman l'adresse MAC de robin dans les destinations des ping request. Vous devriez trouver des messages dans ce cas. En regardant le trafic est-ce que vous êtes capables de comprendre pourquoi ? Comment éviter cela ? Faites le nécessaire pour que ce problème cesse et que l'attaque ARP marche de façon systématique.

Visibilité de l'attaque et TTL

À partir de ce moment nous considérons que l'attaque ARP réalisée est un Man in the Middle, c'est à dire que vous aurez corrompu les caches de batman et de robin.

3.8) Faites un traceroute depuis batman vers robin lorsque l'empoisonnement n'a pas lieu puis lorsqu'il a lieu. Qu'est-ce que vous en déduisez ?

3.9) Comment vous pouvez utiliser iptables pour résoudre ce problème ?

Visibilité de l'attaque (avancé)

Si batman regarde son trafic en détail il va pouvoir avoir plusieurs indices de l'attaque. Le premier est que arp-sk envoie régulièrement des réponses ARP pour maintenir la fausse association dans le cache, ce qui n'est pas un comportement habituel. Le deuxième n'apparaîtra que si batman essaye de contacter joker. Si joker est une machine occulte dans le réseau ceci est peu probable mais sinon batman va potentiellement pouvoir voir dans sa table ARP ou son trafic que l'adresse MAC de robin et de joker c'est les mêmes !

Pour être le plus invisible possible le mieux ça serait d'utiliser deux interfaces, une avec une adresse MAC pour les communications de joker et une autre, occulte, pour l'empoisonnement de robin. Il est possible d'avoir deux adresses MAC sur une même interface en utilisant un bridge et ebtables mais la configuration est complexe et au delà des objectifs de ce TP.

3.10) Avec deux interfaces on pourrait réduire le nombre de réponses ARP qu'on envoie à batman pour maintenir l'entrée dans le cache. Arrêtez l'envoi régulier de réponses ARP pour l'empoisonnement, mais continuez le ping. Observez les requêtes ARP, en provenance de batman, qu'apparaissent au bout d'un moment (attention à la destination). Expliquez pourquoi avec deux adresses MAC on pourrait envoyer moins de réponses ARP mais pas les faire complètement disparaître.

Empoisonnement par création

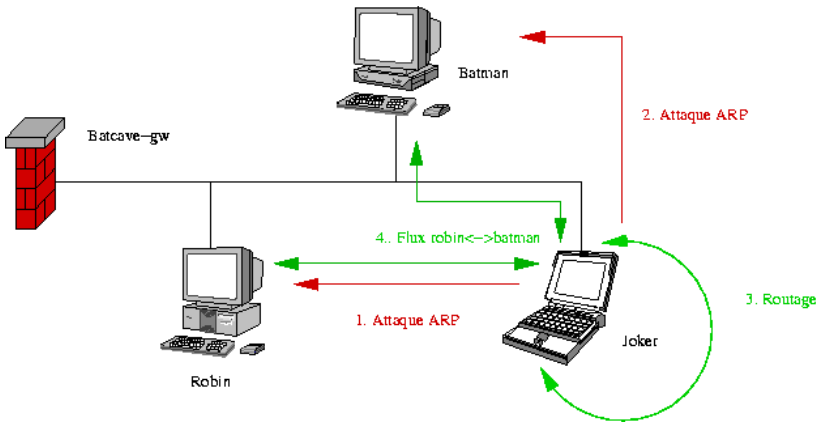
3.11) Faites ce qu'il faut pour que la table arp de batman soit vide. Essayez de créer depuis joker une fausse entrée dans la table ARP de batman associant l'adresse IP de robin à l'adresse MAC de joker en envoyant des réponses ARP comme dans la question 3.2. Ça ne devrait pas marcher. Pourquoi ?

De façon générale quand on ne sait pas si une entrée existe ou pas on envoie premièrement une requête ARP. Si on respecte le protocole, cette requête est envoyée en diffusion et l'attaque est très visible. Si on ne respecte pas le protocole et on envoie la requête en unicast, on diminue l'étendue de visibilité, mais on envoie un message assez suspect du fait qu'on ne respecte pas le protocole. Du coup, quand on fait une attaque ARP, on envoie généralement une premier message de type requête en unicast au cas où l'entrée n'existerait pas et après on envoie régulièrement des paquets de type réponse ARP pour maintenir l'entrée dans le cache.

3.12) Faites un empoisonnement complet avec création et maintien de l'entrée ARP. Servez-vous de l'aide de arp-sk pour voir comment envoyer la requête en unicast en indiquant l'adresse destination au niveau de la couche liaison.

4 Attaques

4.1) Faites un man in the middle complet comme indiquée dans la figure ci-dessous, tirée de [1].



4.2) Ajoutez une règle à iptables pour faire un déni de service ciblé interdisant à batman de se connecter sur robin par ssh mais laissant passer tout le reste. Essayez de le faire simplement puis de façon plus la plus discrète que vous pouvez imaginer.

Interception et modification de trafic

4.3) Batman est mort. Débranchez cet ordinateur et déconfigurez-le. On va l'utiliser pour simuler un ordinateur sur Internet. Branchez-le à **l'interface extérieure** de batcave-gw, appelez-le google, et donnez-lui l'adresse que vous voulez.

4.4) Lancez Apache sur l'ordinateur google et copiez le fichier `pages_web/google/index.html` du répertoire tpsecu dans `/var/www/html/`. Ouvrez un navigateur sur robin et sur la barre d'adresse tapez google. Vous devriez obtenir une page web qui a une certaine ressemblance avec celle çde Google Inc.

4.5) Comme pour google, mettez en place un serveur web chez joker mais utilisez comme page d'accueil `pages_web/owned/index.html`.

4.6) Faites un MITM avec joker entre robin et batcave-gw. Utilisez iptables avec la cible REDIRECT pour que lorsque robin demande la page google il tombe sur celle de joker.

4.7) Imaginez qu'il y a un autre ordinateur à l'extérieur de la batcave appelé yahoo. Supposez que robin essaye de se connecter à son serveur web. Est-ce que l'interception à toujours lieu ? Pourquoi ?

En pratique au lieu d'un serveur web, un attaquant mettrait en place un proxy web avec des capacités d'introspection dans HTTP. Ainsi il pourrait mettre des règles sur quel trafic il laisse passer, quel trafic il modifie et quel trafic il redirige. La configuration d'un tel proxy est sans grand intérêt pour ce TP et donc on en restera là.

Usurpation d'adresse IP

Robin est l'administrateur de la batcave. Comme tel il est le seul à pouvoir traverser le firewall sans restrictions. Les autres membres de la batcave ne peuvent sortir que pour faire du web (ports 80 et 443). Joker peut usurper facilement l'IP de robin, mais s'il ne fait pas d'empoisonnement ARP il ne recevra pas les réponses aux paquets envoyés. Il pourrait empoisonner juste batcave-gw, et si robin était un ordinateur très protégé ceci aurait un sens. Ceci dit on a vu qu'en pratique les attaques ARP marchent mieux si on fait un MITM, du coup à moins qu'il y ait une très bonne raison, pour usurper l'adresse IP de robin, joker empoisonnera et la passerelle et robin.

4.8) Configurez batcave-gw en accord avec ce qui a été dit. Essayez directement depuis joker de vous connecter sur google en ssh. Ça ne devrait pas marcher alors que depuis robin ça devrait marcher.

4.9) Positionnez joker en MITM entre robin et batcave-gw et activez le routage. Vérifiez que depuis robin vous pouvez continuer à interagir avec google (par ping, navigation web, connection SSH, etc.).

4.10) Si vous essayez de vous connecter depuis joker sur google en SSH sans rien faire de particulier ça ne va pas marcher. Pourquoi ? Essayez de réfléchir sur ce qu'il faudrait faire pour que joker arrive à se faire passer par robin. A priori vous utiliseriez quoi pour implémenter cette stratégie ?

Ce qu'on veut dans ce cas c'est que pour certaines applications, l'adresse IP de joker soit remplacée par celle de robin dans l'envoi des messages associés. Pour les réponses on souhaite pouvoir séparer celles qui concernent les requêtes initiées par joker de celles initiées par robin. Pour celles concernant les requêtes initiées par joker on veut qu'avant le routage l'adresse de robin soit remplacée par celle de joker. En bref, face à "l'extérieur" robin et joker vont avoir une même adresse alors qu'à "l'intérieur" ils auront des adresses différentes. Quel mécanisme capable d'afficher une adresse unique et pourtant séparer les communications des uns et des autres connaissez vous ? ? ?

4.11) Faites le nécessaire pour que joker continue à avoir un trafic normal engendré par lui SAUF quand il voudra faire du SSH auquel cas il se fera passer par robin. Testez qu'il peut traverser le firewall en 22 et pas en 80.

5 Compléments d'information : Ettercap

Le logiciel Ettercap permet de réaliser des empoisonnements ARP très simplement. Si on le souhaite, il peut empoisonner les caches de deux hôtes pour se mettre en MITM entre eux ou empoisonner tous les membres d'un réseau pour contrôler tout le trafic (bien sûr si le réseau est grand ceci sera très manifeste et très coûteux).

Ettercap permet de faire de l'écoute en cherchant des mots clés classiques (password, login, etc.) mais aussi de faire des attaques plus poussées par l'intermédiaire de modules. Ainsi, il peut faire de la redirection vers un proxy comme on a fait, mais aussi des empoisonnements DNS, et même des attaques de SSH downgrade (altérer la négociation pour faire choisir le protocole SSHv1 si les hôtes l'acceptent et attaquer ce protocole qui est facilement cassable). Le nombre de modules d'Ettercap est très important, et il y a même un module qui permet de détecter les attaques ARP.

L'étudiant curieux pourra installer Ettercap et explorer son interface graphique, très simple à utiliser, ou son interface en ligne de commande. L'étudiant très curieux observera par écoute locale quel est la stratégie de Ettercap pour comparer avec ce qu'on a fait en TP. L'étudiant très très curieux explorera comment sont faits les modules (c'est extrêmement simple) et essayera d'en faire d'autres par lui même.