Exercices de sécurité informatique

Jean BENOIT
Direction Informatique - Université de Strasbourg
16, rue René Descartes
67 000 STRASBOURG

Résumé

Enseigner la sécurité des systèmes d'information est un problème difficile : le domaine est immense et la théorie seule ne permet pas de sécuriser efficacement des systèmes. Le jeu semble être une solution intéressante d'apprentissage par la pratique.

Un exercice de sécurité est une simulation par équipe qui place des participants dans une situation plus ou moins réaliste où ils doivent attaquer et/ou défendre des systèmes. Il existe plusieurs types de compétition avec des règles et des scénarios parfois très complexes, conçus pour les besoins de différentes communautés (hackers, étudiants, militaires).

L'intérêt d'un exercice de sécurité est à la fois technique et humain. L'acquisition ou le renforcement des concepts essentiels de sécurité, la compréhension fine des mécanismes d'attaques, mais aussi la maîtrise des outils techniques améliorent les connaissances des participants. Par ailleurs, le jeu active des compétences personnelles comme la capacité à travailler en équipe et à prendre des décisions avec des informations partielles dans une période de temps contrainte.

L'organisation d'un exercice de sécurité exige beaucoup de préparation. Il est nécessaire de définir le public cible, les objectifs pédagogiques et le contenu détaillé de l'exercice : scénario, règles du jeu, durée. Son implémentation va demander la mise au point d'une plate-forme technique ainsi qu'un ensemble d'éléments logistiques.

Mots clefs

formation, serious game, sécurité appliquée, sensibilisation par la pratique, plate-forme d'apprentissage, défense d'infrastructure, jeu

1 Introduction

Faire un exercice de sécurité informatique est une manière originale et performante d'enseigner la sécurité par la pratique. Dans une première partie, nous allons voir que les exercices peuvent être très divers et satisfaire des objectifs différents. Dans un deuxième temps, nous examinerons ce qu'ils apportent sur le plan technique et humain et pourquoi ce sont des outils extrêmement pertinents pour améliorer la sécurité. Enfin, nous détaillerons les différents aspects à considérer pour pouvoir organiser des exercices de sécurité.

2 Qu'est-ce qu'un exercice de sécurité informatique ?

Un exercice de sécurité fournit un lieu pour pratiquer des stratégies, des outils, des techniques et des bons réflexes. Plus précisément, un exercice de sécurité informatique consiste à *jouer dans des environnements simulés*, sans impact sur des systèmes ou des environnements réels. Les participants sont des personnes impliquées dans la sécurité. Les objectifs sont :

— de comprendre comment ces environnements fonctionnent (outils, protocoles, interactions entre les éléments du systèmes, points de fragilité etc.) ,

— de s'exercer à des *techniques spécifiques* de sécurité (cryptographie, rétro-ingéniérie, analyse forensique, tests de pénétration, défense des systèmes etc.)

— de pratiquer une *démarche de sécurité plus générale*, et notamment d'apprendre à réagir face à des situations réelles : attaques, compromission ...

— de tester les limites d'un environnement afin d'en améliorer la sécurité ; le jeu simule un environnement proche de la réalité (une application, un ensemble de systèmes ou même une organisation),

— de se divertir, en résolvant des défis techniques dans le cadre d'une compétition.

Si ces objectifs paraissent distincts, ils contribuent conjointement à forger une solide culture de la sécurité, aussi bien au niveau opérationnel qu'au niveau de la conception de systèmes sécurisés.

L'idée directrice est que personne ne maîtrise la sécurité informatique en disposant uniquement de connaissances théoriques : les techniques doivent être *appliquées de manière répétée* pour que la compréhension des problèmes soit bien ancrée et que les mesures de sécurité soient opérantes.

Par exemple, lors d'une attaque, il faut savoir où chercher l'information (capture de paquet, fichier de log ...) et savoir l'interpréter. Cela suppose de disposer des connaissances techniques nécessaires pour comprendre la situation, d'être capable de corréler des informations en provenance de différentes sources, et d'être apte à décider des actions à mener.

Les incidents de sécurité sont souvent différents, éventuellement complexes, et certains types d'incident sont peu fréquents. Lorsqu'un incident a lieu, les personnes en charge de les traiter n'en tirent que des enseignements partiels : durant l'incident, en raison de l'urgence, l'analyse du problème est parfois superficielle et un dé-briefing formel post-incident n'est pas effectué systématiquement. Il parait donc judicieux de simuler ces situations plus régulièrement pour que les acteurs développent leurs compétences et renforcent leur capacité de réaction.

2.1 Différents types d'exercice

On recense de nombreux types d'exercices. Il est difficile de les classer précisément car beaucoup de paramètres entrent en jeu : les objectifs, le public visé, le niveau de difficulté, la dimension, la durée, la logistique nécessaire (donc le temps de préparation et le coût global) etc. Le côté ludique est souvent mis en avant. Certains exercices se pratiquent dans le cadre d'une compétition. D'autres sont entièrement tournés vers la coopération. La pratique peut être individuelle ou en équipe.

2.1.1 Exercice sur table

Appelé *tabletop exercise* en anglais, l'exercice ressemble beaucoup à un jeu de rôle de type «Donjons et Dragon». Autour d'une table, plusieurs participants jouent un rôle (directeur général, responsable informatique, RSSI, etc.) pour résoudre une crise simulée [1]. L'objectif est de tester les procédures et le fonctionnement général d'une organisation en cas d'incident majeur. Ce jeu ne fait pas appel à des compétences techniques très avancées, contrairement aux exercices suivants.

2.1.2 Exercice défensif

Un individu ou une équipe dispose d'une infrastructure de réseaux, de serveurs et d'applications à protéger. Des attaques sont lancées sur l'infrastructure, qu'il s'agit de défendre. Les éléments de l'infrastructure peuvent être volontairement vulnérables. Les participants se voient confier une infrastructure déjà opérationnelle mais qu'ils ne connaissent pas. Ou alors, ils construisent une infrastructure, avec plus ou moins

de contraintes de budget et de temps, selon des spécifications données. Les attaques peuvent être lancées de manière automatisée ou manuelle. L'efficacité de la défense peut être mesurée en comptant le nombre de vulnérabilités corrigées, le nombre d'attaques arrêtées etc. On y ajoute souvent la mesure de la disponibilité : les vulnérabilités doivent êtres bloquées tout en maintenant le service accessible.

2.1.3 Exercice offensif

Un individu ou une équipe est chargé d'attaquer une infrastructure informatique dédiée pour en trouver les vulnérabilités. Le temps utilisé et le nombre de vulnérabilités trouvées permettent de comparer les performances respectives des participants.

2.1.4 Exercice d'attaque/défense et CTF

Un exercice d'attaque/défense mêle les deux types d'exercice précédents. Il est fréquent d'affecter des rôles à chaque équipe : l'équipe attaquante sera l'équipe rouge (red team) et l'équipe qui défend sera l'équipe bleue (blue team). Mais il est également possible que chaque équipe joue les deux rôles simultanément. Dans ce cas, l'exercice est dit *full-spectrum* (champ complet).

L'archétype de l'exercice d'attaque/défense est une compétition appelée *Capture the Flag* ou *CTF*. C'est la transposition dans un environnement informatique du jeu de plein air du même nom. Dans le jeu en plein air, deux équipes s'affrontent : chaque équipe possède un drapeau. Elle doit capturer le drapeau de l'équipe adverse, tout en protégeant le sien.

Dans l'exercice de sécurité informatique, le drapeau, *flag*, est un jeton cryptographique : c'est simplement un nombre tiré aléatoirement (par exemple «3d8e8620fd9142615261cc42928ad268»). Ces *flags* sont mis en place par les organisateurs sur les infrastructures des participants. Ils peuvent être capturés en attaquant un service. L'exploitation d'une vulnérabilité donne accès au *flag* (par exemple, il est contenu dans un fichier, uniquement accessible si le service a pu être compromis). La capture d'un *flag* prouve donc que le service a été attaqué avec succès.

Chaque équipe dispose d'une infrastructure vulnérable (en général la même pour toutes les équipes, pour qu'elles soient sur un pied d'égalité). L'exercice est à la fois défensif et offensif. C'est un face-à-face entre équipes :

— Chaque équipe doit protéger son infrastructure, en empêchant les équipes adverses de la pénétrer pour y prendre des *flags*. Elle doit donc trouver les vulnérabilités et les corriger.

— Elle doit également attaquer l'infrastructure des autres équipes pour capturer un maximum de *flags*. Comme son infrastructure est la même que celle des autres, elle peut essayer d'exploiter des vulnérabilités sur ses propres services, avant de lancer la même attaque sur l'infrastructure adverse.

Un système de score permet de suivre les actions d'attaque et de défense, soit lorsque une équipe soumet le *flag* capturé au système de calcul des scores, soit par la détection automatique de *flags* dans la capture du trafic réseau (le trafic ne doit pas être chiffré). De nombreuses variantes existent : la capture d'un *flag* rapporte des points. L'écrasement d'un *flag* par le *flag* d'une autre équipe peut également rapporter des points (dans ce cas, le système de score doit surveiller les machines et détecter l'écrasement). Les mêmes vulnérabilités peuvent être exploitées plusieurs fois si elles ne sont pas corrigées, mais un même *flag* ne peut être capturé qu'une fois. Dans le cours du jeu, les *flags* peuvent être régénérés par les organisateurs. Ces possibilités dépendent des règles du jeu.

Toute une culture *hacker* s'est développée autour des CTF. Des sites maintiennent des tableaux de classement couvrant toutes les compétitions majeures [2], affichant le score cumulé des équipes à travers leurs différentes participations. Certains publient des solutions sur le web (les CTF *writeups* [3]). Certaines compétitions proposent des primes et des équipes se professionnalisent. La participation est devenue internationale,

avec des nombreuses équipes Chinoises, Sud-coréennes, Japonaises, Russes, Européennes et Américaines. La victoire, ou simplement la résolution d'un challenge difficile est une source de fierté et de prestige dans la communauté (revendiquer des *bragging rights* est un trait typique de la culture hacker). Certains individus sont même très connus dans ce milieu (en général par leur pseudonyme) et font souvent carrière dans le domaine de la sécurité. Le succès des CTF va grandissant : au début, c'était une pratique confidentielle issue d'une communauté obscure, les hackers. En effet, ces derniers s'identifient à une contre-culture revendiquant le droit d'utiliser Internet comme un outil émancipateur. La compétition est aujourd'hui en passe de devenir une pratique reconnue, assimilable aux tournois de jeux vidéo, y compris pour les gains, même si les CTF, en raison de leur plus faible notoriété, sont moins bien sponsorisés et donc très modestement dotés (les prix les plus importants sont de l'ordre de quelques millier d'euros [4]).

2.1.5 Défi de type *Jeopardy*

Figure 1 - Un défi de type Jeopardy

Ce type d'exercice est dit «statique». Contrairement aux exercices dit «dynamiques» décrits précédemment, (attaque, défense, CTF etc.) les défis de type *jeopardy* ne se déroulent pas en temps réel et il n'y a pas d'interactions directes entre les équipes. Il peut donc être pratiqué à distance, en général via une application web. Du matériel (un fichier binaire, une url etc.) et des questions sont fournies. Les participants ont entre plusieurs heures et plusieurs semaines pour résoudre ces problèmes et soumettre leurs solutions. Ces compétitions sont souvent utilisées pour tester les capacités des participants et les qualifier pour un CTF dynamique. Le nom *Jeopardy* vient du jeu télévisé, car différentes questions sur différents thèmes sont posées avec un niveau de difficulté croissant 1. Les défis peuvent être de diverses natures :

— cryptographique : un message secret doit être décodé ; il constitue le *flag* à soumettre

— rétro-ingéniérie : un exécutable fourni doit être *cracké* pour révéler le *flag*

— forensique : le *flag* est dissimulé dans un fichier, une image disque ou une capture réseau par exemple

Dans un CTF attaque-défense dynamique (en temps-réel), il est possible d'avoir une partie de la compétition consacrée à des questions de type *Jeopardy*, qui vont compter pour le score global des équipes.

2.1.6 Exercices mixtes et exercices à grande échelle

Il est possible de mêler différents types d'exercices. Par exemple, il existe des exercices à grande échelle qui impliquent des experts techniques, des juristes, des membres de CERT, des représentants des gouvernements de plusieurs pays. Durant plusieurs jours se déroule un scénario où des participants s'exercent à la décision et à la coordination dans un contexte de crise pendant que des experts sécurité défendent des systèmes et des réseaux contre des attaques simulées (par exemple, l'exercice *Baltic Cyber Shield* organisé par l'OTAN, suite aux cyber-attaques contre l'Estonie).

2.2 Quelques exemples d'exercices

2.2.1 DEF CON CTF

Figure 2 - Des équipes à l'œuvre durant DEF CON 15 CTF

La compétition la plus connue est le DEF CON CTF. Créé par Jeff Moss, DEF CON est un rassemblement de hackers qui a lieu chaque année à Las Vegas depuis 1993. Au début, des terminaux étaient détournés de leur usage par certains participants pour afficher du texte sur l'écran d'autres utilisateurs. À partir de 1996 *(DEF CON IV)* a été lancée la première compétition, de type offensif, avec des règles du jeu formalisées. Chaque équipe devait pirater des serveurs identiques. Un panel de juge accordait des points aux équipes en fonction de leurs succès.

Jusqu'en 2001, DEF CON CTF était très chaotique car il s'agissait davantage de pirater la compétition que les serveurs, et les règles du jeu évoluaient constamment. Les règles de la plupart des CTF interdisent de conduire des attaques de déni de service sur l'infrastructure du jeu, sous peine de disqualification. Dès 2002, le jeu s'est stabilisé : chaque compétition était précédée d'une phase de qualifications. Un système de calcul de score automatique relativement équilibré a été mis au point, et la topologie du réseau a été établie (réseau en étoile [figure 2] autour d'un routeur central faisant de la translation d'adresse) [5]. Depuis, le jeu se focalise sur les applications [6] [7] et évolue vers plus de complexité et de difficulté. Des applications spécifiques sont écrites pour chaque édition du jeu. L'objectif est d'en exploiter les vulnérabilités.

Le niveau de difficulté de la compétition est extrêmement élevé : les challenges peuvent porter sur des architectures matérielles exotiques spécialement créées pour l'occasion et pour lesquelles aucun debugger ou dés-assembleur ne fonctionnent. Chaque équipe doit inclure des spécialistes de différents domaines : rétro-ingéniérie *(reverse-engineering)*, cryptographie, etc. Certaines années, les règles autorisaient le recours à des personnes extérieures intervenant à distance. Ainsi, une équipe a pu compter plus de 40 membres. *DEF CON CTF* est un défi de taille, qui met à rude épreuve la résistance (la compétition dure plusieurs jours) et le talent des participants. Certains s'entraînent spécifiquement avant l'événement, pour parfaire la maîtrise de leurs techniques et de leurs outils. Durant la compétition, il est souvent nécessaire de développer très rapidement des programmes pour l'analyse du comportement des services et l'exploitation des vulnérabilités. La résolution des challenges exige des capacités techniques très poussées et une virtuosité intellectuelle que peu de hackers possèdent.

2.2.2 Un club de *hacking* : Comsoc

Des étudiants de l'Université du Texas à Austin ont monté un laboratoire de sécurité [8] dans les locaux de leur amicale en 2001, en réutilisant des ordinateurs destinés à être jetés. Leur objectif était d'apprendre la sécurité informatique en installant et en testant des systèmes. Le club s'est organisé autour de formations, d'ateliers pratiques et de différents projets d'infrastructure et de sécurité. Le développement des activités a rapidement conduit à créer une compétition de piratage offensif au sein du club, inspiré du DEF CON CTF.

C'est un jeu à petite échelle, qui tourne en continu durant l'année universitaire, avec des phases de jeu d'une semaine à deux mois ou plus [9]. À chaque phase de jeu, un organisateur met en place un système cible et donne des objectifs d'attaques aux participants. La phase de jeu se conclut lorsque tous les objectifs sont atteints. Le réseau et l'environnement du jeu sont derrière une passerelle uniquement accessible en SSH, et aucun accès vers l'extérieur n'est autorisé depuis le jeu. Une règle stricte a dû être mise en place, interdisant aux étudiant d'utiliser les techniques d'attaque en dehors du réseau dédié à la compétition. L'inconvénient principal de cette activité est qu'elle n'est pas intégrée à un cursus officiel. Cette expérience a contribué à la création de la première compétition inter-universitaire américaine au niveau national.

2.2.3 CDX

Suite à la construction d'un laboratoire sécurité destiné à la formation des étudiants, l'IWAR Range [10], l'académie militaire de West Point a eu l'idée de développer une compétition [11] [12]. Introduit en 2001, le CDX *(Cyber Defense Exercice)* [13] voit s'affronter des cadets de cinq académies militaires américaines. Dans le cadre d'un projet durant plusieurs mois, les étudiants doivent construire un réseau comportant des services opérationnels selon des spécifications données, avec des contraintes budgétaires. Les réseaux des différentes écoles sont reliés par des VPN à un point central. Une équipe attaquante *(red team)* composée d'experts externes tente de pénétrer l'infrastructure gérée par les étudiants. C'est un jeu essentiellement défensif où les participants doivent minimiser l'impact des attaques, maintenir l'infrastructure en conditions opérationnelles, évaluer la posture de sécurité, et faire les rapports réguliers des incidents et des corrections apportées. Un panel de juge accorde des points sur la rapidité d'intervention, la couverture des défenses, et l'exhaustivité des rapports.

Les principes de l'exercice sont proches d'autres entraînements conduits par des militaires. À travers le réalisme de la simulation, l'objectif est de développer des compétences spécifiques : planification, mise en œuvre de mesures de sécurité, capacité d'analyse, réaction à incident etc. Ce qui est assez remarquable, c'est que la totalité de cet exercice long et difficile s'inscrit dans le programme pédagogique sous le nom de *capstone project*, autrement dit un *chef-d'œuvre*. L'intention des enseignants est claire : au moyen-âge, le système du compagnonnage exige à travers un rituel initiatique que le futur compagnon démontre son excellence et prouve qu'il est digne d'intégrer la corporation en construisant un chef-d'œuvre. Les étudiants doivent donc synthétiser et intégrer toutes les connaissances apprises pour produire un objet technique fonctionnel et dont ils connaissent parfaitement les rouages.

2.2.4 CCDC

Les deux initiatives précédentes, comsoc et CDX, ont été à la base d'une compétition annuelle inter-universitaire. Un groupe de travail réunissant les hackers du comsoc, les fondateurs du CDX et d'autres parties prenantes s'est tenu à San Antonio en 2004 [14] pour tenter de déterminer la structure d'une telle compétition. La première *Collegiate Cyber Defense Competition (CCDC)* a eu lieu en 2005. La compétition est devenue nationale (*NCCDC*). Elle s'est structurée en un championnat divisé en 10 grandes régions. Chaque région organise des qualifications pour sélectionner l'université qui participera à la compétition nationale.

C'est un jeu essentiellement défensif [15]. Chaque équipe reçoit une infrastructure déjà paramétrée. Le score tient compte de la sécurité et de la disponibilité des services. Des points de pénalité sont donnés tant que le service ne fonctionne pas (de manière similaire à un contrat de service). Au cours du jeu, des tâches spécifiques d'administration (nommées *injects*) sont demandées en un temps limité (ajout d'un service par exemple) dont la réalisation rapporte des points supplémentaires. Destinée aux étudiants de premier cycle, le niveau de difficulté est bien inférieur au DEF CON CTF.

2.2.5 Cyber Europe

Cyber Europe est un exercice pan-européen de type *tabletop*. Le premier a eu lieu en 2010 [16]. 22 états membres de l'Union Européenne ont pris part à l'exercice, organisé par l'ENISA *(European Union Agency for Network and Information Security - Agence européenne chargée de la sécurité des réseaux et de l'information)*. L'objectif était de renforcer la collaboration et la coordination de la réponse à un incident majeur à l'échelle européenne. L'exercice a permis de tester les capacités de communication entre différents organismes de sécurité (CERTs nationaux, agences nationales de sécurité etc.) au niveau européen. La technicité était assez faible. Le plus difficile pour les personnes non-techniques impliquées a sans doute été de faire une requête DNS et une requête à la base Whois pour trouver le point de contact d'un autre pays. Mais l'exercice a eu le mérite de faire prendre conscience du manque de pratique, de l'absence de standards et de procédures, de la diversité de fonctionnement des structures et des difficultés de coordination à cette échelle. Il peut contribuer à construire un climat de confiance entre les acteurs et à faire comprendre la nature d'une crise de grande ampleur.

3 À quoi sert un exercice de sécurité ?

Les exercices peuvent être très différents dans leurs objectifs et dans leur nature. Cependant, ils ont tous en commun un aspect d'apprentissage et d'amélioration des participants. Ils développent notamment des compétences techniques mais aussi des aptitudes humaines. Utilisés avec les étudiants, les exercices sont également un excellent moyen d'évaluer leur motivation et de repérer les talents.

3.1 Aspects techniques

3.1.1 Comprendre

Les exercices permettent de faire le *lien entre la théorie et la pratique*. On peut imaginer à quoi ressemble une attaque de déni de service, comment se comporte un service vulnérable, et quelles sont les conséquences d'un excès de privilège. Mais le réalisme d'une simulation rend cela beaucoup plus clair. On voit une attaque se dérouler en vrai. On constate des points de fragilité du système. On découvre notamment la complexité de la réalité technique et l'interdépendance des services, qu'on étudie souvent séparément. Par exemple, en tentant de sécuriser fortement la configuration d'un serveur de nom, d'affiner des règles de filtrage de paquet, ou de réduire radicalement les privilèges d'un utilisateur, le fonctionnement de l'ensemble de l'infrastructure peut se retrouver bloqué. Chaque action faite sur un système a des conséquences. C'est une leçon que seule la pratique rend concrète.

La pratique renforce la *connaissance des techniques et des concepts*. On découvre que certains outils sont trop superficiels ou trop lents sur un système réel. On apprend à utiliser de nouveaux outils dont on ignorait l'existence. La combinaison de plusieurs outils doit être réalisée sur le coup, alors qu'elle n'a jamais été imaginée auparavant. Les concepts eux-mêmes sont mis en doute. Sous Unix, dans un répertoire autorisé à la modification pour tous, peut-on supprimer un fichier appartenant à l'administrateur ? Ces cas limites sont rarement couverts en cours, et vite oubliés s'ils sont abordés. Le feedback immédiat de l'exercice répond à ces questions de manière rapide et définitive. Il est vraisemblable que l'enseignement des systèmes d'exploitation ou des protocoles réseau sous l'angle de la sécurité rende ces sujets beaucoup plus intéressants encore. La simulation dans un environnement contrôlé est rassurante, car elle autorise toutes les expérimentations sans crainte de casser ou d'impacter un système de production.

L'identification de *patterns d'attaque* est également une notion importante, qui ne peut acquise par la seule théorie. En observant des éléments dans les logs, sur le système de fichier, dans les paquet réseaux, des motifs se dessinent. À force de répétition, un ensemble de traces simples amènent à une compréhension plus

large d'une situatin. Des intuitions deviennent des quasi-certitudes. Par exemple, un fichier ou un identifiant qui n'était pas présent précédemment amène un questionnement, une recherche plus approfondie. Un autre indice trouvé, par exemple un processus suspect, qui semble lié apporte une confirmation de l'idée initiale : l'ensemble de ces éléments mettent en évidence une compromission.

3.1.2 Agir

La pratique ne développe pas seulement des concepts et un outillage technique mais aussi des capacités à agir.

Dans une compétition, la prise de *décision technique* est une expérience délicate. En effet, chaque action nécessite d'intégrer tous les paramètres, de calculer les conséquences et de trouver un compromis. Seules les décisions techniques précédentes (et l'éventuel échec en résultant) et la maîtrise progressive des concepts et des interactions forment cette capacité à juger et à décider. L'apprentissage par l'erreur est efficace.

La correction d'un système en production est une manœuvre difficile. L'approche correcte est d'appliquer une procédure de test et de valider chaque action avant d'agir sur le vrai système. L'expérience répétée de cette démarche donne l'assurance à celui qui effectue l'opération que l'impact sur la disponibilité sera faible. Cette aptitude peut être exercée en participant à un exercice où le score pénalise l'indisponibilité des services.

La pratique de l'exercice améliore la *vitesse d'exécution*. Certaines actions deviennent des réflexes car elles ont été faites de très nombreuses fois. Par exemple, la découverte d'un élément textuel (une signature d'attaque, une option de configuration inhabituelle) conduit rapidement à former une requête dans un moteur de recherche. Le parcours cursif *(skimming)* des résultats de cette recherche ou d'une page de documentation liée amène très vite à la réponse pertinente. Un autre réflexe qui se développe est celui d'imaginer un algorithme et d'écrire très vite un petit script approprié à l'espace du problème à résoudre. Souvent, l'exploration d'une vulnérabilité (*scan* de port, essais de mot de passe en force brute, exploitation d'un dépassement de tampon etc.), d'un fichier de log, ou d'un ensemble de données à corréler est plus opportunément faite par un programme. C'est ce qu'on appelle de la paresse bien appliquée.

Enfin, l'outil principal développé et exercé est *une démarche*. La mise en œuvre d'un plan cohérent structure les actions. Dérouler le plan à chaque exercice garantit que les choses les plus importantes ont été faites et donne une certaine assurance. La démarche doit inclure l'habitude du doute systématique : est-ce que j'ai bien compris le problème ? Quelles sont les possibilités que j'ai oubliées ? Par exemple, la démarche initiale dans un exercice défensif pourrait être : inventaire des services, application d'une politique de filtrage, application d'une politique de mise à jour, analyse et sécurisation des configurations etc. La première intervention va permettre de protéger la plate-forme sur une grande partie du périmètre contre des attaques les moins ciblées. Dans un deuxième temps, il convient d'examiner les risques résiduels et de creuser toutes les éventualités.

3.2 Aspects humains

Au delà des outils, l'expérience de l'exercice est également formatrice du côté de l'humain. Certain participant montrent de compétences non-techniques intéressantes : [17] une approche générale dans la résolution de problèmes, le fait de penser en dehors des sentiers battus, la persistance et la détermination, le développement des stratégies multiples, la confiance en soi, le fait d'être prêt à tout essayer et prendre le risque d'avoir tort, etc.

3.2.1 Travail en commun

Le travail en équipe avec un objectif commun apprend aux participants à se coordonner, à communiquer et à s'adapter au modèle mental forcément différent de ses partenaires. La planification et la répartition des tâches obligent à expliciter sa stratégie en faisant preuve de pédagogie et à accepter un regard extérieur critique.

Il faut s'entraîner soi-même à intégrer des informations nombreuses de diverses sources, puis les synthétiser pour les partager avec les autres. Il faut également accepter d'apprendre des autres. La confrontation de ses idées avec des inconnus, des personnes ayant des niveaux et des pratiques différentes amène une réelle richesse des échanges.

3.2.2 Attitude

Il convient d'adopter une *attitude ouverte*. Même si on pense être suffisamment expérimenté, chaque jeu est différent. Face à l'imprévu, la capacité à trouver une solution s'appuie sur son expérience et son imagination. L'approche des hackers est riche d'enseignement [18]. Ils privilégient la connaissance en profondeur du fonctionnement des choses («regarder sous le capot»), la connaissance des cas limites et le développement d'outils spécifiques pour examiner les états complets d'un système et pour les modifier. Les hackers sont à l'origine d'articles très précis et fournissant des connaissances directement utilisables sur des failles, mais aussi sur des techniques pour combattre certaines classes de vulnérabilité. Par exemple, sur les dépassements de tampon *(buffer overflow)*, le texte le plus cité sur le sujet n'est pas une publication académique mais l'article de 1996 d'Aleph One dans le magazine électronique Phrack [19]

Cultiver *l'humilité* conduit à l'envie de s'améliorer. Le feedback de l'exercice, par le biais du score, permet de s'auto-évaluer. Souvent, les participants constatent l'étendue de leur ignorance. Le domaine et les types d'attaque leur sont inconnus. Dans la mesure où il s'agit d'un jeu, ce n'est pas grave, bien au contraire. C'est l'occasion d'apprendre des choses nouvelles, de découvrir ses capacités à réagir et à résister au stress. Il faut accepter de faire des choix dans un environnement où l'information est partielle, avec la possibilité de se tromper. La motivation est une propriété intrinsèque de jeu. Le réalisme, l'attrait de la nouveauté (la situation n'a jamais été vécue) et l'aspect compétitif concourent à rendre l'expérience d'apprentissage immersive et divertissante.

4 Comment organiser un exercice de sécurité ?

Même si les exercices sont des outils très formateurs, leur mise en place est une affaire complexe. Certaines personnes ayant déjà organisé des exercices recommandent avec humour de ne pas le faire, car c'est une tâche colossale...

4.1 Préparation

4.1.1 Définition des objectifs

C'est une étape indispensable : il faut définir précisément les objectifs et les publics ciblés, Il faut aussi choisir le type d'exercice adapté aux besoins.

Plusieurs objectifs sont possibles : former et sensibiliser différentes populations, tester une organisation, évaluer le niveau technique des personnes dans une logique d'amélioration, etc. Les publics potentiels sont des étudiants, des professionnels de la sécurité, des responsables, etc.

S'il s'agit de tester une organisation, par exemple dans le but d'évaluer si les procédures fonctionnent correctement en cas de crise, un exercice sur table est approprié (c'est d'ailleurs une mesure recommandée dans la norme ISO 27035 [20] au chapitre 5.8).

Des administrateurs systèmes qui n'ont jamais subi certains types d'attaques peuvent y être sensibilisés par le jeu. Une mise en situation dans un exercice défensif peut leur enseigner les différentes choses à vérifier et les actions à entreprendre.

Pour former des étudiants à la sécurité, il faut au préalable leur donner les concepts et les outils. De manière générale, inscrire les exercices dans le cadre d'une compétition rend les choses plus intéressantes. Il faut mentionner que des compétitions existent déjà dans d'autres domaines de l'informatique (programmation, robotique, Intelligence Artificielle ...) et suscitent beaucoup d'engouement. Les récompenses éventuelles (coupe, trophée...) renforcent la motivation. Différent types d'exercices peuvent être utilisés : exercices défensifs, exercices offensifs, attaque/défense.

Développer des capacités offensives chez des étudiants est en général perçu comme dangereux. La dangerosité est à rapporter aux éléments suivants :

— D'un point de vue pédagogique, enseigner l'attaque a des avantages [21], comme une meilleure compréhension des mécanismes ou une motivation et une curiosité plus importantes.

— Un expert d'un domaine doit savoir réaliser des attaques ; par exemple, un serrurier sait aussi bien monter que casser une serrure.

— Les informations permettant de conduire les attaques sont déjà disponibles sur Internet, et les attaquants malveillant y ont déjà accès. Cela permet donc un équilibre des forces.

Il faut bien sûr encadrer les exercices de type offensif en introduisant un minimum de déontologie, en informant sur les conséquences de chaque action et sur le fait que l'attaque est détectable. Lors de la démonstration de chaque attaque, il faut également enseigner la contre-mesure. Dans certaines circonstances, il est conseillé de faire signer une charte. Mais le dialogue et la responsabilisation sont essentiels.

4.1.2 Organisation et planification

La préparation d'un nouvel exercice nécessite beaucoup de temps et de ressources. Il faut s'y prendre plusieurs mois à l'avance, en fonction de la complexité du scénario et du nombre de participants. Parmi les différents éléments à préparer, il y aura sans doute :

— la définition des objectifs et du public cible,

— le budget prévisionnel,

— la recherche éventuelle de sponsors,

— la communication autour de l'exercice,

— l'écriture du scénario,

— la formalisation détaillée des règles (éligibilité des participants, quelles actions sont autorisées ou interdites, comment sont calculés le score et les pénalités, les aspects juridiques, etc) ,

— l'infrastructure de l'exercice (architecture, dimensionnement, achat et préparation des machines physiques et virtuelles, mise en place des équipements réseau et du câblage, documentation, développements spécifiques, mise aux points des vulnérabilités, du système de calcul des scores, tests fonctionnels, tests de charge etc.),

— la constitution des équipes (sondage éventuel du niveau des participants, répartition des participants, ...),

— l'organisation éventuelle d'une phase de qualification (qui peut être elle-même un ensemble d'exercices statiques à préparer, avec sa logistique et son infrastructure propre),

— tous les autres aspects logistiques (locaux, matériel, repas, recrutement des personnes volontaires pour aider à l'organisation, etc.).

Un comité en charge de l'organisation est obligatoire à partir d'une certaine taille. Chaque élément peut prendre entre plusieurs jours et plusieurs semaines de préparation. Il faut noter que le contenu développé pour un événement ne peut pas être réutilisé, car les solutions sont accessibles publiquement à l'issue de la compétition. Si l'exercice s'inscrit dans un cursus pédagogique, il faut planifier de long mois à l'avance les cours introduisant les concepts nécessaires et les séances d'entraînement préparatoires régulières.

4.2 Contenu

Les scénarios peuvent être très variés, du plus simple au plus complexe.

4.2.1 Un scénario simple

Un exemple de scénario simple : les *CERT Games*. C'est une compétition défensive en temps réel, préparée par Comcert.pl, à destination des professionnels. Les équipes qui y participent sont dans la situation suivante : il faut administrer deux machines qui hébergent des services de messagerie, des applications web, un DNS et du partage de fichier. Les serveurs ont une vingtaine de vulnérabilités (un mot de passe faible, un problème de droits etc.) Chaque équipe gère un couple de machines et peut intervenir sur n'importe quelle partie du système, mais les services doivent rester accessibles, sous peine de pénalité. Avant la première attaque, les équipes disposent de deux heures pour évaluer le niveau de sécurité des machines et appliquer les mesures qu'elles jugent nécessaires. Plusieurs vagues d'attaques automatisées sont ensuite lancées. Le score, composé de la note de sécurité et de disponibilité, est affiché en temps réel. Ce scénario de base peut être amélioré en ajoutant d'autre type de failles [22, p.106] et notamment en installant d'anciennes versions d'applications [23] comportant des vulnérabilités.

4.2.2 Un scénario complexe

Un exemple de scénario très original mais plus complexe : le jeu conçu pour l'édition 2010 [24] de l'*iCTF (international Capture The Flag)* [25] par l'équipe de Giovanni Vigna de l'*UCSB* (Université de Californie à Santa Barbara). Cette compétition distribuée, accessible à distance à des étudiants comptait 72 équipes et 900 participants. Le jeu est de type offensif en temps réel. Pour résumer les aspects les plus fondamentaux des règles : les équipes font partie d'une coalition dont l'objectif est de renverser le gouvernement de Lytia, un état voyou imaginaire. Plusieurs applications interagissent entre elles : un IDS capable de détecter et de bloquer les attaques (*IDS* qu'on peut désactiver temporairement en soudoyant le service *Briber*), un *botnet* sur lequel il faut être connecté pour pouvoir accéder aux services vulnérables (il faut envoyer régulièrement au *botnet* un certain nombre d'unités d'une monnaie fictive dont la quantité est limitée). Les services vulnérables pouvaient être attaqués uniquement en suivant un timing précis. Le bon moment pour attaquer le service correspondait à un état dans un graphe (des réseaux de pétri, fournis avant le début du jeu). Cet état devait être déduit des transitions révélées par un autre service, LytiaLeaks. Les règles obligent les équipes à prendre des décisions stratégiques sur l'utilisation du temps et des ressources. Bien sûr, un certain nombre d'équipes n'a pas compris comment fonctionnaient les parties essentielles du jeu.

4.2.3 Difficulté/facilité de l'exercice

Tous les scénarios ne sont pas aussi complexes. Le niveau de difficulté de l'exercice aura une influence directe sur sa jouabilité et l'implication des participants. Un exercice trop facile suscitera l'ennui et un

exercice trop difficile générera de la frustration ou des abandons. Il faut veiller à équilibrer la quantité d'informations fournies et les informations qu'il faut découvrir [26]. Dans les CERT Games par exemple, un dévoilement progressif de la nature des vulnérabilités permet d'éviter le découragement en milieu et fin du jeu. Une autre possibilité de rendre la complexité graduelle est, dans un jeu offensif, de ne donner accès qu'à la première machine ou à la première partie du jeu. Dans cette première partie, des tâches à accomplir (exploitation de vulnérabilités, recherche cryptographique ou forensique) permettent de découvrir des informations pour déverrouiller la deuxième partie ou un accès à une deuxième machine, etc.

La composition des équipes influe également sur la difficulté du jeu. Il est recommandé d'évaluer le niveau des participants avant l'exercice pour former des équipes homogènes. Il est intéressant de mêler des personnes de très bon niveau avec des personnes moins fortes, notamment pour des exercices destinés aux étudiants. Dans ce contexte, il faut parfois ajouter des règles spécifiques pour que les personnes les moins expérimentées de l'équipe participent pleinement [27] S'il y a des équipes spécifiques (par exemple une *red team*, équipe attaquant les autres dans un jeu défensif) les rôles doivent être répartis judicieusement. L'équipe attaquante peut être constituée par des personnes qui ont participé au jeu les années précédentes.

En général, la création de leurre augmente la difficulté. Par exemple, il y a plusieurs *flags* à protéger dans un jeu attaque/défense, mais certains seulement sont valides et seuls les organisateurs du jeu savent lesquels. L'équipe qui défend doit les protéger tous.

Une mesure essentielle pour augmenter la difficulté du jeu est le masquage des adresses réseau. Dans des exercices défensifs ou offensifs, on peut avoir un trafic de contrôle spécifique (supervision, soumission des *flags*, scores etc.) en complément du trafic lié aux attaques. Il est important de rendre ces trafics de contrôle impossible à distinguer du trafic des attaques. Dans le cas contraire, les attaques sont plus faciles à trouver. Plusieurs approches sont possibles :

— réécrire les paquets réseau : champs TTL, adresse IP source (avec une translation d'adresse simple, ou en faisant varier aléatoirement les adresses source),

— utiliser la même plate-forme pour mener les attaques et lancer les interrogations de supervision.

Il peut être pertinent d'introduire des délais aléatoires car la régularité des tests est également détectable.

4.2.4 Score

Le calcul informel et manuel du score peut être pratiqué à petite échelle uniquement (c'était le cas du *CDX* où des juges [11] accordaient des points), mais l'automatisation du score est préférable pour des questions d'efficacité et d'impartialité. Les algorithmes de calcul du score doivent être particulièrement soignés. Ils doivent être simples et lisibles pour les participants. Le calcul doit être robuste et rendre la triche difficile. Par exemple, il est recommandé d'enregistrer les événements de score dans un journal pour pouvoir les rejouer (plutôt que de maintenir simplement la valeur courante du score). Sinon, en cas de panne de la plate-forme (par exemple, suite à une attaque intentionnelle sur le système de score), si la valeur du score est perdue, il sera difficile de la reconstituer. Dans certains CTF, les actions effectuées sur le système sont enregistrées, afin de calculer des scores de manière plus élaborée mais plus complexe : en effet, cela permet, au delà de savoir si la vulnérabilité a été corrigée, de déterminer comment elle a été corrigée. De manière générale, enregistrer et conserver les traces (*logs*, paquets réseau) est une idée pertinente. Certaines traces sont même utilisables pour faire des recherches. [24, p.6]

4.3 Plate-forme

4.3.1 Infrastructure de virtualisation

Il y a plusieurs approches pour créer une plate-forme qui hébergera l'exercice. L'architecture la plus fréquente est basée sur un réseau interne dédié et des machines virtuelles ou des conteneurs (l'utilisation des

containers facilite la mise au point dans la mesure où le système de fichier du container est accessible depuis le système hôte).

La sécurité et la solidité de la plate-forme doivent être éprouvées. Il faut veiller à ce que les machines virtuelles des équipes soient isolées entre elles, mais aussi isolées du reste du réseau [28]. Il faut éviter à tout prix que des attaques puissent fuiter vers Internet ou que des machines vulnérables faisant partie de l'exercice soient compromises par des attaques externes. L'accès à la plate-forme se fait souvent via un VPN. Au préalable, il faut générer, imprimer et mettre sous enveloppe cachetée (physique ou électronique) les identifiants et les mots de passe VPN et administrateur des machines. Au moment précis où l'exercice commence, on procédera à la distribution de ces informations aux équipes.

L'exercice se déroule sur une courte période pendant laquelle aucune interruption n'est tolérée. La plate-forme obéit aux mêmes règles que toute infrastructure critique. La robustesse de la conception du système s'appuiera sur des techniques classiques : redondance (réseau doublé, machines de rechange), possibilité de restaurer une image en cas de problème, etc. Des tests doivent être effectués régulièrement lors du développement des composants. Il faut aussi tester les interactions entre les composants lors de leur intégration entre eux. Il faut s'intéresser à la disponibilité en cas de forte charge. La plate-forme doit être testée et calibrée par rapport au dimensionnement de l'exercice (nombre d'équipes, nombre de machines, charge réseau liées aux attaques). Les règles sont un moyen de limiter certain problèmes de ce type :

— interdiction des attaques DoS,

— restriction de la bande passante utilisée (soit par des pénalités portées au score, soit par des dispositifs techniques limitant activement le débit).

4.3.2 Cyber range

Pour proposer des exercices régulièrement, certaines institutions se dotent d'une plate-forme qu'on appelle *cyber range* [29]. Ce terme, issu des militaires américains, s'inspire du mot *shooting range*, stand de tir ; il s'agit donc d'un terrain d'entraînement spécifiquement appliqué à la sécurité informatique. C'est un investissement important (à partir de dizaines de milliers d'euros et jusqu'à plusieurs millions d'euros). Un *cyber range* est composé d'un ensemble d'éléments : serveurs, équipements réseaux, logiciels (hyperviseurs, systèmes d'exploitation, applications, générateurs de trafic), contenus pédagogiques (cours, exercices)... La configuration de la plate-forme est modulaire : certains des composants cités sont optionnels. Des plus, les équipementiers spécialisés dans les *cyber range* proposent différentes prestations autour des plate-formes : mise à jour des contenus, conception sur-mesure, développements spécifiques, etc.

La pérennité d'un tel investissement dépend de sa capacité à évoluer : intégration de nouvelles versions de système d'exploitation, de nouvelles vulnérabilités, de nouvelles classes d'attaques, voire de nouvelles émulations matérielles (par exemple, *SCADA* pour simuler des réseaux industriels) etc.

La plupart implémente un simulateur d'environnement, ce qui permet notamment de composer des exercices. Dans une interface graphique, il suffit de glisser-déplacer des icônes représentant un système d'exploitation, un pare-feu ou un routeur, de les connecter entre elles par des liaisons virtuelles. Au démarrage de la simulation, des machines et des réseaux virtuels implémentant ces fonctions sont lancés automatiquement.

Le gouvernement Japonais a annoncé son intention de former plusieurs milliers de professionnels à la sécurité informatique avant les jeux Olympiques de Tokyo en 2020 [30] [31]. Plusieurs universités se sont équipées de *cyber range*. L'université de Kyushu [32] a notamment choisi d'investir dans une plate-forme commerciale et de former les étudiants en informatique dans le but de sélectionner les meilleurs étudiants pour en faire des experts.

À Taïwan, le *NCHC (National Center for High-performance Computing)* a développé une plate-forme mutualisée [33] utilisée par les étudiants de différentes universités taïwanaises pour des travaux pratiques

et des compétitions de type attaque-défense. La plate-forme est basée sur des outils *open source* comme *Ezilla*, un outil de déploiement et *OpenNebula*, un outil de gestion de *cloud*.

En France, le centre commun de gestion de crise cybernétique de l'ENSIBS (Vannes, université Bretagne sud) a mis en place un *cyber range* [34] destiné à la formation d'étudiants et de professionnels.

Les *cyber range* prennent progressivement de l'importance. Différentes initiatives sont apparues dans les dernières années. Il y a vraisemblablement une prise de conscience que les étudiants et les professionnels ont de sérieux besoins de formations appliquées.

5 Conclusion

Encore méconnus aujourd'hui, les exercices de sécurité constituent un moyen efficace pour augmenter le niveau de sécurité d'une organisation. Leurs principaux bénéfices sont d'améliorer les compétences techniques mais ils se focalisent aussi sur les aspects humains. L'humain est l'élément essentiel dans tout dispositif de sécurité, à la fois à l'origine des risques et porteur des solutions. Les retours d'expérience des participants sont encourageants : les exercices éveillent la curiosité et suscitent un véritable plaisir d'apprendre et de comprendre. Il est permis d'imaginer que, parce qu'elle a des répercussions très concrètes sur les façons de faire, la pratique des exercices se diffusera plus largement.

Annexe

Bibliographie

[1] Nina Wilhelmson et Thomas Svensson. *Handbook for planning, running and evaluating information technology and cyber security exercises*. Center for Asymmetric Threat Studies (CATS) - Swedish National Defence College, 2011.

[2] CTFTIME.ORG. https://ctftime.org/.

[3] CTFs. https://github.com/ctfs.

[4] Chris Eagle. Keynote - CTF - All the Cool Kids are doing it. Dans *Code Blue Conference*, Février 2013. https://www.youtube.com/watch?v=t1w08LLKn5Y.

[5] Riley "Caezar" Eller. Capture The Flag Games - Measuring Skill with Hacking Contests. Dans *Black Hat Asia*, 2004. https://www.blackhat.com/presentations/bh-asia-04/bh-jp-04-pdfs/bh-jp-04-eller/bh-jp-04-eller.pdf.

[6] Kenshoto. WarGamez Redux. Dans *DEF CON 17*, 2009. https://www.youtube.com/watch?v=O8oJ5CMXo3k.

[7] Alexander Taylor. defcon-vm. https://github.com/fuzyll/defcon-vm.

[8] Georges Chamales et Adam Pridgen. The Success of the UT IEEE Communications Society. Dans *The 8th Colloquium for Information System Security Education (CISSE)*, 2004. https://cisse.info/resources/archives/category/2-papers?download=2:s1p02-2004.

[9] Ryan Smith et Georges Chamales. *University of Texas Cyber Security Exercise*, page 24. 2004. https://cspri.seas.gwu.edu/sites/cspri.seas.gwu.edu/files/Exploring%20a%20National%20Cyber%20Security%20Exercise%20for%20Colleges%20and%20Universities_Hoffman_2004.pdf#page=27.

[10] Joseph Schafer, Daniel J. Ragsdale, John R. Surdu et Curtis A. Carver. The IWAR range : a laboratory for undergraduate information assurance education. Dans *Proceedings of the sixth annual CCSC northeastern conference on The journal of computing in small colleges*, pages 223–232, 2001.

[11] Wayne J. Schepens, Daniel J. Ragsdale, John R. Surdu et Joseph Schafer. The Cyber Defense Exercise : An Evaluation of the Effectiveness of Information Assurance Education. Dans *The journal of information security*, 2002.

[12] Wayne J. Schepens et J.R. Jame. Architecture of a Cyber Defense Competition. Dans *IEEE International Conference on Systems, Man and Cybernetics*, volume 5, pages 4300–4305, 2003.

[13] Donald Welch, Daniel Ragsdale et Wayne Schepens. Training for Information Assurance. Dans *Computer*, volume 35, pages 30–37. IEEE, 2002.

[14] Daniel Ragsdale Lance J. Hoffman. *Exploring a National Cyber Security Exercise for Colleges and Universities*. Tech Report No. CSPRI-04-08. Cyber Security Policy and Research Institute (CSPRI), 2004. https://cspri.seas.gwu.edu/sites/cspri.seas.gwu.edu/files/Exploring%20a%20National%20Cyber%20Security%20Exercise%20for%20Colleges%20and%20Universities_Hoffman_2004.pdf.

[15] Gregory B. White et Dwayne Williams. The Collegiate Cyber Defense Competition™. Dans *Proceedings of the 9th Colloquium for Information Systems Security Education (CISSE)*, 2005.

[16] ENISA. *Cyber Europe 2010 Report*. 2011. https://www.enisa.europa.eu/publications/ce2010report/at_download/fullReport.

[17] D.G. Oblinger. The Next Generation of Educational Engagement. Dans *Journal of Interactive Media in Education.*, volume 2004(1), p.Art. 10., 2004.

[18] Sergey Bratus. What hackers learn that the rest of us don't, 2009. http://www.cs.dartmouth.edu/~sergey/hacker-methodology.pdf.

[19] Aleph One. Smashing the Stack for Fun and Profit. Dans *Phrack*, volume 7, Novembre 1996. http://www.phrack.com/issues.html?issue=49&id=14.

[20] *ISO/IEC 27035 :2011 Technologies de l'information – Techniques de sécurité – Gestion des incidents de sécurité de l'information*. Organisation internationale de normalisation, Genève, Suisse., 2011.

[21] Martin Mink et Rainer Greifeneder. Evaluation of the Offensive Approach in Information Security Education. Dans Kai Rannenberg, Vijay Varadharajan et Christian Weber, éditeurs, *SEC 2010. IFIP Advances in Information and Communication Technology*, volume 330 de *Security and Privacy - Silver Linings in the Cloud*, pages 203–214. Springer, 2010. https://link.springer.com/content/pdf/10.1007%2F978-3-642-15257-3_18.pdf.

[22] Mike O'Leary. Small-scale cyber security competitions. Dans *Proceedings of the 16th Colloquium for Information Systems Security Education (CISSE)*, pages 103–110, Juin 2012. https://cisse.info/resources/archives/category/29-papers?download=302:p16-2012.

[23] Old Apps. http://www.oldapps.com/.

[24] Adam Doupe, Manuel Egele, Benjamin Caillat, Gianluca Stringhini, Gorkem Yakin, Ali Zand, Ludovico Cavedon et Giovanni Vigna. Hit 'em where it hurts : A live security exercise on cyber situational awareness. Dans *Proceedings of the Annual Computer Security Applications Conference (ACSAC)*, 2011.

[25] The UC Santa Barbara iCTF Competition. https://ictf.cs.ucsb.edu/.

[26] Richard Weiss, Frankly Turbak, Jens Mache, Erik Nilsen et Michael E. Locasto. Finding the balance between guidance and independence in cybersecurity exercises. Dans *USENIX Workshop*

on Advances in Security Education, 2016. https://www.usenix.org/system/files/conference/ase16/ase16-paper-weiss.pdf.

[27] CANVAS : a regional assessment exercise for teaching security concepts. Dans *Proceedings of the 12th Colloquium for Information Systems Security Education (CISSE)*, pages 66–71, Juin 2008. https://cisse.info/resources/archives/category/10-papers?download=105:s04p02-2008.

[28] Giovanni Vigna. Teaching Hands-on Network Security : Testbeds and Live Exercises. Dans *Journal of Information Warfare*, volume 2, issue 3, pages 8–24, 2003.

[29] Lutte informatique defensive : le role-clé des cyber range en matière d'entraînement. Dans *Observatoire du Monde Cybernétique*. Compagnie Européenne d'Intelligence Stratégique (CEIS), Mars 2017. http://www.defense.gouv.fr/content/download/502443/8528007/file/OBS_Monde%20cybern%C3%A9tique_201703.pdf.

[30] Gouvernement du Japon. CYBERSECURITY STRATEGY, Septembre 2015. https://www.nisc.go.jp/eng/pdf/cs-strategy-en.pdf.

[31] Mihoko Matsubara. Japan's Cybersecurity Effort towards Tokyo 2020 and Its Global Significance. Dans *Asia Pacific & Japan RSA Conference*, Juillet 2017. https://published-prd.lanyonevents.com/published/rsaap15.5861_ap17/sessionsFiles/3157/PGR-F03-Japan%E2%80%99s-Cybersecurity-Effort-Towards-Tokyo-2020-and-Its-Global-Significance.pdf.

[32] Koji Okamura. Cybersecurity Training with Cyber Range. Dans *CNMS 2016 - Campus Network Monitoring and Security workshop*, Avril 2016. https://www.cesnet.cz/wp-content/uploads/2015/12/160425-oka-OKAMURA.pdf.

[33] Yi-Lun (Serena) Pan. The Lightweight Approach to Build Cloud CyberSecurity Exercise. Dans *OpenNebula Conf*, Octobre 2016. https://www.youtube.com/watch?v=HQiiATKpyzU.

[34] Cyber Security Center - Centre de formation, de recherche et d'entraînement à la cyberdéfense. http://www.cyber-security-center.com/Accueil-257-0-0-0.html.