SYNACKTIV

FORMATIONS

2026.04

Présentation

Synacktiv prend à cœur le partage de son expérience en cybersécurité, acquise au fil des années, en dispensant des formations inter-entreprises. Alliant enseignement théorique et travaux pratiques, nos formations ont été conçues pour offrir une expérience d'apprentissage unique et enrichissante, et s'adressent principalement à des professionnels de la sécurité des systèmes d'information : pentesteurs, experts en retro-ingénierie, analystes SOC, analystes CSIRT, administrateurs systèmes, architectes sécurité, développeurs, etc.

Chaque session est animée par deux formateurs expérimentés qui assureront une compréhension optimale tout en apportant des retours d'expérience concrets. Tout le matériel nécessaire à la réalisation des travaux pratiques sera fourni aux étudiants et chacun disposera d'un environnement individuel afin d'assurer une expérience d'apprentissage immersive. Les supports de cours seront transmis au format PDF, permettant aux participants de les consulter à tout moment et de les utiliser comme référence.

Les formations se déroulent dans nos locaux parisiens, dans un environnement professionnel et confortable qui favorisera la concentration des apprenants. Les déjeuners et boissons sont inclus, ainsi qu'un repas au restaurant le dernier jour de formation.

- 2 formateurs expérimentés
- 7 à 12 participants
- Minimum 50 % de pratique
- Travaux pratiques au sein de labs individuels
- Matériel fourni (ordinateurs portables)
- Supports de cours fournis
- Dans nos locaux 5 bd Montmartre, Paris 75002
- Repas et boissons inclus

Pentest

Offensive Security Fundamentals

Obtenez les compétences nécessaires à la compréhension des principales étapes d'une intrusion. Reconnaissance, applications web, systèmes Linux et Windows, étapes de post-exploitation, cette formation fournit un socle essentiel à tout professionnel de sécurité.

5 jours | Junior

Active Directory Intrusion Tactics: Entry Level

Découvrez les fondamentaux de la sécurité des environnements Active Directory au travers de cette formation offensive. D'un accès anonyme et jusqu'à la compromission complète des infrastructures, devenez autonome en intrusion de réseaux d'entreprises.

5 jours | Intermédiaire

DevOps & Linux Breach Tactics

Maîtrisez les techniques d'intrusion sur des infrastructures DevOps et Linux au travers de cette formation offensive. Compromission de micro-services, injection de pipelines CI/CD, et empoisonnement d'artefacts, devenez autonome en intrusion de réseaux d'entreprises.

5 jours | Avancé

Active Directory Intrusion Tactics: Advanced Level

Approfondissez vos compétences d'intrusion en environnements Active Directory avec cette formation de niveau confirmé. Découvrez les techniques d'exploitation avancées et maîtrisez la compromission de réseaux d'entreprises complexes.

5 jours | Avancé

Cloud Intrusion Tactics

Initiez-vous à la compromission de réseaux modernes avec cette formation sur les infrastructures cloud. GCP, AWS, Azure et Kubernetes, découvrez les mécanismes caractéristiques de ces technologies récentes, avec la posture d'un attaquant.

5 jours | Intermédiaire

Azure Intrusion Tactics

Affinez vos compétences en intrusion sur Azure avec cette formation avancée. Plongez dans Entra ID, les services M365, les ressources Azure, la CI/CD avec Azure DevOps, la gestion Intune et les environnements hybrides à travers des scénarios d'exploitation subtils et réalistes.

5 jours | Avancé

Pentest

Kubernetes Intrusion Tactics

Initiez-vous à la compromission des plateformes containerisées avec cette formation dédiée à Kubernetes. Comprenez les mécanismes d'authentification, d'autorisation, de réseau et d'évasion propres à Kubernetes, en adoptant la posture d'un attaquant pour identifier failles et pistes de remédiation.

2 jours | Junior

AWS Intrusion Tactics

Initiez-vous à la compromission des environnements AWS avec cette formation dédiée à l'offensive cloud. Comprenez IAM, exploitation des métadonnées EC2, abus de S3, techniques sur Lambda et mouvements latéraux, en adoptant la posture d'un attaquant pour identifier vulnérabilités et pistes de remédiation.

2 jours | Junior

Attacking Web Applications

Étudiez les mécanismes de sécurité des applications web modernes et les méthodes d'exploitation avancées permettant de les contourner. PHP, Java, Python et ASP.NET, maîtrisez la compromission d'applications web complexes.

5 jours | Intermédiaire

Practical Web 0-Day Hunting

Obtenez les compétences nécessaires à la recherche de vulnérabilités web Java, PHP et .NET. Étude de frameworks et outils d'analyse statique et dynamique, cette formation permet aux pentesteurs et développeurs d'optimiser leur recherche de vulnérabilité en boîte blanche.

5 jours | Avancé

Offensive CI/CD

Maîtrisez l'exploitation CI/CD sur GitHub et GitLab. De l'injection dans les pipelines au détournement de runners et à l'extraction de secrets, apprenez à exécuter des chaînes d'attaques complexes et à compromettre les infrastructures de développement modernes.

2 jours | Intermédiaire

Attacking Android Applications

Découvrez les méthodologies et techniques d'analyse des applications Android. Architecture des applications, points d'entrée, analyses statique et dynamique, maîtriser le pentest en environnement Android.

2 jours | Junior

Pentest

Password Cracking

Étudiez les méthodes d'optimisation du cassage de mots de passe avec les outils John et Hashcat. Règles de mutation, masques, attaques prince et siga, devenez un véritable expert des mots de passe.

1 jour | Junior

Reverse

Embedded Linux Exploitation

Obtenez la méthodologie essentielle à l'évaluation de sécurité de systèmes Linux embarqués. Analyse matérielle, émulation, fuzzing et exploitation de vulnérabilités : cette formation constitue un socle indispensable pour tout professionnel de la sécurité offensive.

5 jours | Junior

Hardware Intrusion

Apprenez à apprivoiser un PCB : reconnaître des composants, identifier des testpads et inférer puis interagir avec des protocoles (UART, JTAG/SWD, SDIO, SPI). Utiliser les outils et matériels actifs/passifs (analyseur logique, FT2232H, JTAGulator, OpenOCD).

5 jours | Intermédiaire

Android for Security Engineers

Découvrez de façon approfondie et à l'aide d'exercices pratiques, le fonctionnement d'Android et de ses mécanismes de sécurité.

5 jours | Intermédiaire

iOS for Security Engineers

Découvrez de façon approfondie et à l'aide d'exercices pratiques, le fonctionnement d'iOS et de ses mécanismes de sécurité.

5 jours | Intermédiaire

Advanced IDA

Familiarisez-vous avec les fonctionnalités avancées d'IDA, son API et son écosystème. Apprenez comment développer des scripts et plugins pour étendre ses fonctionnalités.

5 jours | Avancé

DMA Attacks

Maîtrisez les attaques matérielles et l'exploitation des accès mémoire directs (DMA). De la manipulation d'équipements de capture à la modification à la volée avec PCILeech et l'analyse forensic de la RAM, apprenez à compromettre physiquement des postes de travail.

4 jours | Intermédiaire

Forensics

Windows Forensics

Maîtrisez l'investigation numérique des systèmes Windows 10 et 11 en apprenant à identifier et caractériser les malveillances associées, autant dans le cadre d'un incident de sécurité que d'une recherche de compromission (levée de doute, hunting).

5 jours | Junior

Linux Forensics

Maîtrisez l'investigation numérique des systèmes Linux en apprenant à identifier et caractériser les malveillances associées, autant dans le cadre d'un incident de sécurité que d'une recherche de compromission (levée de doute, hunting).

5 jours | Junior

Mobile Forensics

Découvrez l'investigation numérique de système d'exploitation mobile Android et iOS en étudiant les techniques d'acquisition de données, la découverte d'applications malveillantes ou encore les artefacts de fonctionnement du téléphone.

5 jours | Junior

Kubernetes Forensics

Maîtrisez l'investigation numérique des environnements Kubernetes. Comprenez le fonctionnement interne des composants de clusters afin de répondre efficacement aux compromissions. Identifiez des menaces et maintenez l'ordre dans le cluster.

3 jours | intermédiaire

Malware Analysis

Découvrez l'analyse de code malveillant dans le cadre d'un incident de sécurité au travers de situations diverses et de cas réels de modes opératoires d'attaquants.

3 jours | Intermédiaire

Advanced Malware Analysis

Effectuer la rétro-ingénierie de menaces dans différents environnements et identifier des techniques pour lutter contre les codes malveillants obfusqués et dissimulés à l'aide de scripts.

3 jours | Avancé

Forensics

Cloud Forensics in AWS

Explorez AWS pour comprendre le fonctionnement de son système de journalisation. Apprenez à détecter les principales attaques sur ces environnements et découvrez des cas d'utilisation spécifiques pour reconstituer l'activité des acteurs malveillants.

3 jours | Junior

Cloud Forensics in Azure

Explorez Microsoft Azure, Entra ID et M365 pour comprendre le fonctionnement du système de journalisation. Découvrez comment détecter les principales attaques sur ces environnements et les cas d'utilisation spécifiques pour reconstituer l'activité des acteurs malveillants.

3 jours | Junior

Ransomware Investigation

Appréhendez les gestes de premiers secours et les mesures d'urgence sur un incident ransomware. Poursuivez votre compréhension de la situation par des méthodes d'investigation adaptées à cette menace et démarrez la remédiation.

3 jours | Intermédiaire

Active Directory: Hardening & Post-Compromise Recovery

Sécurisez le cœur de votre SI en maîtrisant le durcissement, la supervision et la remédiation Active Directory. Du déploiement du Tiering à la reprise de confiance post-intrusion, apprenez à bloquer les attaquants et à reconstruire un environnement sain.

5 jours | intermédiaire

Data Breach: Investigations, Crisis Management & Compliance

Maîtrisez la gestion de crise lors d'une fuite de données en alliant investigations techniques d'urgence (quick wins) et conformité juridique. Apprenez à qualifier l'incident, piloter la communication et respecter vos obligations légales face à divers scénarios de compromission.

1 jour | Junior

Développement

Agentic AI Red Teaming

Maîtrisez le développement d'architectures autonomes en concevant un agent Red Team capable de raisonner et d'agir. Apprenez à orchestrer des LLM et à intégrer des outils via le protocole MCP.

5 jours | Intermédiaire

Network Interception in Rust

Concevez de A à Z un outil d'interception Man-In-The-Middle performant en Rust pour maîtriser les attaques réseau locales (ARP/DHCP) et la manipulation de flux applicatifs chiffrés (HTTP/TLS).

3 jours | Intermédiaire

Advanced Rust

Maîtrisez les concepts avancés du langage Rust (multi-threading, asynchronisme, typage) en développant itérativement un scanner de fichiers performant, de bout en bout, jusqu'à son pilotage via une interface web.

5 jours | Intermédiaire

Infrastructure

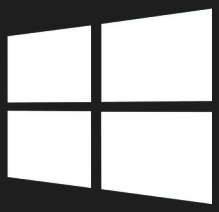
Linux Hardening

Apprenez à réduire drastiquement la surface d'attaque de vos systèmes Linux grâce aux mécanismes de défense natifs. De l'isolation applicative avec AppArmor et Podman au filtrage réseau granulaire via Nftables, construisez des environnements robustes capables d'endiguer toute compromission.

4 jours | Intermédiaire

Offensive Security Fundamentals

5 jours | Niveau junior



Description

La réalisation de tests d'intrusion permet une mise en situation réaliste des mécanismes de défense et représente par conséquent une étape clé dans la sécurisation des systèmes d'information. Cette formation d'introduction au pentest vise à fournir une compréhension approfondie de l'audit de sécurité en abordant les différentes étapes d'une intrusion.

Au cours de ces cinq jours de formation, les participants seront exposés à quatre modules de cours couvrant la reconnaissance, les applications web, les systèmes Linux et Windows, et les techniques de post-exploitation. Chaque module sera illustré par des travaux pratiques guidés permettant d'appliquer les notions théoriques enseignées. Enfin, la formation se conclura par une mise en situation réaliste sur un réseau d'entreprise.

- 5 jours (35 heures)
- 4 modules de cours couvrant les étapes principales d'un test d'intrusion
- Reconnaissance, applications web, Linux, Windows, post-exploitation
- 20 exercices d'application
- 1 intrusion guidée sur environnement d'entreprise complet (10 machines)

Public et prérequis

Cette formation a été conçue pour des personnes n'ayant aucune expérience préalable sur les tests d'intrusion. Elle s'adresse principalement aux pentesteurs débutants, administrateurs systèmes, architectes sécurité et développeurs, mais également à tout profil technique souhaitant enrichir son parcours professionnel avec une composante sécurité.

- Pentesteurs débutants
- Administrateurs systèmes
- Architectes sécurité
- Développeurs

Des connaissances basiques de l'environnement Unix et des langages web sont recommandées.

Contenu

Jour 1

Introduction aux méthodes de reconnaissance : énumération DNS et HTTP, scans de services. Présentation des principaux outils d'intrusion : Metasploit, Burp Suite. Vulnérabilités sur les applications web : injections SQL, XSS (Cross-Site Scripting), XXE (XML eXternal Entities), SSRF (Service-Side Request Forgery), upload de fichiers, désérialisation, avec différents exercices de mise en pratique.

Jour 2

Mise en pratique sur des applications web complexes : reconnaissance, exploitation et élévations de privilèges jusqu'à l'obtention d'un accès aux serveurs. Élévation de privilèges sur les systèmes Linux : fondamentaux (gestion des identités et des accès), reconnaissance et exploitation (permissions, configurations sudo, tâches planifiées, unités systemd, kernel) et technologies de conteneurisation (Docker, LXC/LXD).

Jour 3

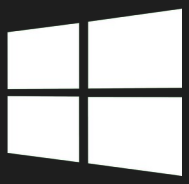
Élévation de privilèges sur les systèmes Windows : fondamentaux (gestion des identités et des accès, gestion des secrets), reconnaissance et exploitation (permissions, configurations de services, tâches planifiées, vulnérabilités publiques). Mise en pratique sur des serveurs depuis un accès non privilégié.

Jours 4 et 5

Étapes de post-exploitation : extraction de secrets (disque, mémoire), installation de portes dérobées et déplacements latéraux (rebond réseau, proxy SOCKS, forward de ports). Mise en situation sur un réseau d'entreprise.

Active Directory Intrusion Tactics: Entry Level

5 jours | Niveau intermédiaire



Description

Pour de nombreuses entreprises, l'Active Directory constitue le cœur de la gestion des identités et des accès. Son omniprésence au sein des systèmes d'information en fait une cible de choix pour les attaques informatiques et les tests d'intrusion sont un composant clé de sa défense contre les menaces.

Au cours de cette formation de cinq jours, vous acquerrez les compétences nécessaires à la réalisation d'un test d'intrusion Active Directory approfondi. En suivant les cinq modules d'apprentissage, les étudiants apprendront la méthodologie et les techniques utilisées par nos experts lors d'une intrusion, depuis un accès anonyme jusqu'à la compromission totale de l'environnement et la persistance des accès en son sein. Afin de mettre en pratique les concepts enseignés, les apprenants seront guidés au travers de deux environnements d'entreprise complets.

- 5 jours (35 heures)
- 5 modules de cours couvrant les étapes d'une intrusion réaliste + 1 module Azure
- 2 environnements d'entreprise avec plus de 30 machines

Public et prérequis

Cette formation est adaptée pour des personnes ayant des notions de sécurité offensive mais pas d'expérience préalable sur les environnements Active Directory. Elle s'adresse principalement aux pentesteurs, administrateurs systèmes et architectes sécurité, mais également à tout profil technique souhaitant enrichir son parcours professionnel avec une composante sécurité.

- Pentesteurs
- Administrateurs système
- Architectes sécurité

Des notions de sécurité offensive et de bonnes connaissances réseau et Unix sont recommandées.

Contenu

Jour 1

Bases théoriques des mécanismes de sécurité : fonctionnement des mécanismes d'administration (RPC, SMB, WMI, RDP, WinRM), gestion des identités et des accès, stockage des secrets, protocoles d'authentification réseau (NTLM, Kerberos), hiérarchie et liens de confiance Active Directory. Techniques de reconnaissance et d'exploitation depuis un accès anonyme : énumération, empoisonnement de protocoles réseau, relaying.

Jour 2

Reconnaissance sur le domaine depuis un accès non privilégié : extraction des objets (utilisateurs, groupes, machines, GPO) et cartographie avec BloodHound. Élévation de privilèges locale : énumération et exploitation (services locaux, tâches planifiées, ACLs, vulnérabilités publiques), techniques de contournement de l'UAC.

Jour 3

Élévation de privilèges au sein d'un domaine : extraction de secrets (registres, LSASS, DPAPI), rejeu d'authentification, kerberoasting, abus de chemins de contrôle. Contournement de restrictions logicielles : AppLocker, évasion de bureaux restreints (Citrix, Kiosque RDP).

Jour 4

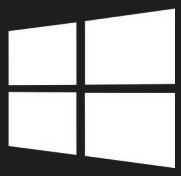
Étapes de post-exploitation depuis un accès privilégié sur le domaine : extraction de secrets (NTDS, DPAPI), forge de tickets (silver et golden tickets), manipulation d'ACL, persistance au sein de l'environnement et effacement des traces. Extension de la compromission : études des relations de confiance inter-domaines et inter-forêts, abus de délégation Kerberos.

Jour 5

Introduction à Azure : concepts fondamentaux (terminologie, gestion des identités et des accès), intégration avec l'Active Directory (synchronisation des identités, mécanisme Single Sign-On), étapes de reconnaissance et compromission depuis l'environnement on-premise.

Active Directory Intrusion Tactics: Advanced Level

5 jours | Niveau avancé



Description

Pour de nombreuses entreprises, l'Active Directory constitue le cœur de la gestion des identités et des accès. Son omniprésence au sein des systèmes d'information en fait une cible de choix pour les attaques informatiques et les tests d'intrusion sont un composant clé de sa défense contre les menaces.

Au cours de cette formation de cinq jours, vous approfondirez vos compétences d'intrusion en environnement Active Directory. Guidé par nos experts, étudiez des techniques avancées de reconnaissance, mouvements latéraux, élévation de privilèges, extraction de secrets et persistance. Afin de mettre en pratique les concepts enseignés, les apprenants seront mis en situation sur deux environnements d'entreprise complets, issus de scénarios de compromission réels.

- 5 jours (35 heures)
- 5 modules de cours couvrant les étapes d'une intrusion réaliste
- 2 environnements d'entreprise avec 30+ machines et des services comme ADCS et SCCM

Public et prérequis

Cette formation s'adresse aux profils techniques confirmés ayant déjà une expérience significative sur les environnements Active Directory. Elle constitue la suite logique du cursus Active Directory Intrusion Tactics: Entry Level. Profils cibles :

- Pentesteurs / red teamers
- Administrateurs système
- Architectes sécurité

Afin de profiter pleinement de cette formation, les participants doivent posséder une expérience préalable sur les points suivants :

- Utilisation courante de la suite Impacket, de NetExec et de BloodHound.
- Maîtrise des attaques basiques : poisoning LLMNR / NBNS, Kerberoasting et rejeu d'identifiants.
- Extraction des secrets relatifs aux environnements Windows (base SAM, processus LSASS).
- Pratique des rebonds et du tunneling (SOCKS / TUN) pour la navigation entre différents réseaux.

De bonnes connaissances en administration Windows, UNIX et une compréhension des protocoles NTLM et Kerberos sont recommandées pour aborder les modules avancés.

Contenu

Jour 1

Fondamentaux : mécanismes Active Directory, principes d'intrusion généraux et spécifiques à ces environnements. Reconnaissance et premières actions depuis un accès authentifié : méthodes de récupération d'information (ADIDNS, détection de services via analyses LDAP et GPO) utilisation avancée de BloodHound (Cypher queries).

Jour 2

Mouvements latéraux : empoisonnement ADIDNS, WinRM et JEA, extraction de secrets LAPS, gMSA/sMSA, abus de liens de confiance MS-SQL, relaying NTLM (dissection, relai cross-protocoles, WebDAV), coercing d'authentification, relai Kerberos, pivots inter-forêts.

Jour 3

Élévation de privilèges locale : access token et impersonation, étude des vulnérabilités potatoes. Élévation de privilèges sur le domaine : étude et abus des ACL, exploitation avancée de délégation Kerberos, ADCS ESC1 à 15, mécanismes et primitives d'exploitation SCCM, abus de groupes privilégiés, analyse de vulnérabilités publiques.

Jour 4

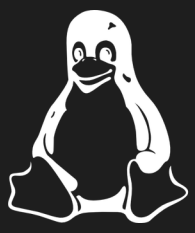
Extraction de secrets : méthodes et outils d'extraction LSASS, usurpation de tokens, analyse des secrets de bases de registres, implémentation DPAPI, extraction de bases KeePass.

Jour 5

Persistance : ADCS (certificats), tickets Kerberos (golden, diamond, sapphire), DSRM, golden gMSA, abus AdminSDHolder, création de skeleton key, délégation Kerberos, empoisonnement de GPO, DC Shadow.

DevOps & Linux Breach Tactics

5 jours | Niveau avancé



Description

Les environnements DevOps reposent massivement sur Linux pour propulser les architectures de micro-services, les infrastructures hybrides et les pipelines de déploiement continu. La sécurité de ces environnements repose sur des outils d'automatisation et de conteneurisation dont la maîtrise est devenue un enjeu critique pour les attaquants modernes.

Au travers de ces cinq jours de formation, les participants seront exposés à cinq modules de théorie détaillant une killchain complète : de l'accès initial par injection de pipelines CI/CD (GitLab, Jenkins) jusqu'à l'abus d'outils Infra-as-Code (Ansible, Terraform) pour compromettre l'infrastructure. Un module complémentaire sera également dédié aux systèmes durcis (AppArmor, SELinux). Ces notions seront appliquées tout au long de la semaine sur deux réseaux d'entreprise complexes, inspirés d'intrusions réellement menées par nos experts.

- 5 jours (35 heures)
- 5 modules de cours sur les étapes d'une intrusion réaliste + 1 module sur les systèmes durcis
- 2 environnements d'entreprise avec plus de 20 machines dont des services GitLab, Jenkins, Artifactory, AWX, HashiCorp Vault, Guacamole, KeyCloak et vSphere
- Distributions RedHat-like et Debian-like

Public et prérequis

Cette formation est adaptée pour des personnes ayant des notions de sécurité offensive mais pas d'expérience préalable sur l'intrusion d'environnements DevOps et Linux. Elle s'adresse principalement aux pentesteurs, administrateurs systèmes et architectes sécurité, mais également à tout profil technique souhaitant enrichir son parcours professionnel avec une composante sécurité.

- Pentesteurs / red teamers
- Administrateurs système
- Architectes sécurité

Des notions de sécurité offensive et de bonnes connaissances réseau et Unix sont recommandées.

Contenu

Jour 1

Concepts fondamentaux : gestion des identités et des accès, mécanismes de sécurité (ACL étendues, attributs standards et étendus, capabilities), fondamentaux réseau, conteneurisation (namespaces, cgroups, seccomp, Docker et LXC / LXD). Micro-services : Docker registry, Portainer, Traefik, Redis, détection de conteneurs, analyse d'images, recherche de secrets et backdooring. Authentification centralisée : Kerberos, OpenLDAP, SSSD et FreeIPA, reconnaissance, énumération et outillage.

Jour 2

Déplacements latéraux : rebond réseau, proxy SOCKS, forward de ports, TUN forwarding et outillage. CI/CD : Jenkins, injection de pipelines et compromission d'agents, Artifactory, dependency confusion et empoisonnement d'artefacts. Postes utilisateurs : KeePass, DBUS Secret Service API, agents SSH et GPG, navigateurs Firefox et Chrome, keyloggers.

Jour 3

Administration : PKI et signature de certificats, manipulations firewall (iptables / nftables), déroutage (DNAT) et spoofing (SNAT), systèmes de fichiers (NFS, Samba, FUSE), gestion des identités et accès (KeyCloak), bastion (Guacamole), packaging et empoisonnement (Deb et RPM). Hyperviseurs (vSphere) : architecture, ESXi et vCenter, méthodes d'authentification, techniques de post-exploitation (captures et contournement réseau, exfiltration optimisée), golden SAML.

Jour 4

GitLab : étude IAM, architecture et implémentation des runners, reconnaissance, accès initial via dump et analyses de projets (ThruffleHog, noseyparker), élévation de privilège via injection dans les pipelines, dump de secrets. Infra-as-Code : ansible, terraform, AWX et déploiements automatisés. HashiCorp Vault : architecture, authentification, politiques d'accès et utilisation.

Jour 5

Exploitation avancée : gestion des processus (sessions, groupes et cycle de vie), fonctionnement du TTY et injections, analyse de stacks PAM et backdooring. Compromission de systèmes durcis : implémentation et configuration des LSM AppArmor et SELinux, analyse et contournement du durcissement.

Cloud Intrusion Tactics

5 jours | Niveau intermédiaire

Description

Les technologies cloud sont progressivement intégrées dans le système d'information des entreprises. Elles apportent de nombreux mécanismes de sécurité parfois difficile à appréhender et forçant les attaquants à repenser leurs méthodes d'intrusion.

Au cours de cette formation de cinq jours, les participants seront exposés aux concepts des trois fournisseurs cloud principaux : GCP (Google), AWS (Amazon) et Azure (Microsoft). Après avoir étudié les fondamentaux qui leur sont communs, les spécificités d'implémentation seront détaillées et illustrées au travers d'environnements complets permettant de s'initier aux techniques d'intrusion cloud. Un module complémentaire sera également dédié aux infrastructures Kubernetes.

- 5 jours (35 heures) modulaires (découpage possible)
- 3 modules de cours sur GCP, AWS et Azure + 1 module dédié à Kubernetes
- 4 environnements complets et individualisés

Public et prérequis

Cette formation est adaptée pour des personnes ayant des notions de sécurité offensive mais pas d'expérience préalable sur les environnements cloud. Elle s'adresse principalement aux pentesteurs, administrateurs systèmes, architectes sécurité et développeurs, mais également à tout profil technique souhaitant enrichir son parcours professionnel avec une composante sécurité.

- Pentesteurs / red teamers
- Administrateurs système
- Architectes sécurité
- Développeurs

De bonnes connaissances réseau et Unix et des notions d'intrusion web sont recommandées.

Contenu

Jour 1

Fondamentaux : terminologie cloud, services d'infrastructure, topologie réseau, gestion des identités et des accès, mécanismes d'authentification (OAuth), rappels des mécanismes de sécurité Linux (namespaces, cgroups, seccomp, LSM), recherche d'accès en source ouverte.

Jour 2

Google Cloud Plateform : architecture (organisation, dossiers, projets, ressources, régions et zones), IAM (permissions, rôles, principaux et politiques), authentification (OAuth 2.0, JWT), utilisation de la CLI gcloud, méthodes de reconnaissance des services, abus de droits sur les buckets, implémentations App Engine et instance (abus des metadata), élévation de privilèges IAM, reconnaissance réseau (VPC, firewall, VPN, peerings), post-exploitation (délégation sur le domaine, rebond sur Workspace), analyse des évènements.

Jour 3

Amazon Web Services : architecture (organisation, comptes), IAM (types d'identité, assumation de rôle, politiques), utilisation de la CLI aws, méthodes de reconnaissance des services, énumération non-authentifiée des identités, abus de droits sur les buckets S3, EC2 (metadata, mouvements latéraux et empoisonnement des agents SSM), Lambdas (runtime API, persistance, exfiltration de données), Cognito (user et identity pools) élévation de privilèges IAM, reconnaissance réseau (VPC, network ACL, security groups), persistance (modification de politiques IAM, role chain juggling).

Jour 4

Azure : architecture (tenants, management groups, subscriptions), Entra ID (types d'identité, gestion des accès, rôles Entra vs Azure), synchronisation en environnement hybride (PHS, PTA, ADFS), reconnaissance non-authentifiée, utilisation de la CLI azure et module Az, reconnaissance authentifiée (ROADrecon, AzureHound), implémentation blob storage, key vault, machines virtuelles, mouvements latéraux (Vnet, bastions).

Jour 5

Kubernetes : architecture (conteneurs, pods, nodes, services internes), reconnaissance, authentification (mot de passe, certificats, tokens) et autorisations (node, ABAC, RBAC, WebHook), utilisation de la CLI kubectl, pod templates et contrôleurs, escapes (namespaces, PSP, PSA), concepts réseau (ingress, pod to pod, CNI, politiques).

Azure Intrusion Tactics

5 jours | Niveau avancé

Description

Azure est aujourd'hui un leader incontesté du cloud, omniprésent dans les infrastructures d'entreprises grâce à son intégration étroite avec les environnements on-premise, notamment via Active Directory. Cette adoption massive en fait une cible de choix pour les attaquants, qui doivent comprendre les spécificités de sa sécurité afin de mener des tests d'intrusion efficaces.

Au cours de cette formation de cinq jours, les participants approfondiront leurs compétences offensives sur Azure. Après une introduction aux principes fondamentaux de la sécurité, l'accent sera mis sur les services clés comme Entra ID, la suite Microsoft 365, les ressources Azure, la CI/CD via Azure DevOps, la gestion Intune et les environnements hybrides. Des scénarios d'exploitation réalistes et discrets permettront aux participants d'acquérir les techniques nécessaires pour compromettre ces infrastructures, tout en adoptant une approche furtive et ciblée.

- 5 jours (35 heures)
- 6 modules de cours couvrant une intrusion complète des services Azure
- 1 environnement réaliste, complet et individualisé

Public et prérequis

Cette formation est adaptée pour des personnes ayant des notions de sécurité offensive, mais pas d'expérience préalable sur les environnements Azure. Elle s'adresse principalement aux pentesteurs, administrateurs systèmes, architectes sécurité et développeurs, mais également à tout profil technique souhaitant enrichir son parcours professionnel avec une composante sécurité.

- Pentesteurs / red teamers
- Administrateurs système
- Architectes sécurité
- Développeurs

De bonnes connaissances réseau et Unix et des notions d'intrusion web sont recommandées.

Contenu

Jour 1

Fondamentaux : architecture du tenant, portails, outils. Entra ID : identités et rôles, framework OAuth 2.0 et implémentation Microsoft, jetons d'accès, concept d'applications, API Microsoft Graph, sécurité (politiques d'accès conditionnel, PIM, détections), méthodes d'authentification, MFA et contournements, journalisation, accès initial et discovery (AzureHound, ROADTools, Microsoft Graph CLI).

Jour 2

Microsoft 365 : analyse des services principaux de la suite bureautique (Teams, Outlook / Exchange, SharePoint / OneDrive, OneNote / Word / Excel), API spécifiques et leur utilisation offensive, gestion des accès avec Microsoft Graph vs. API spécifiques, journalisation.

Jour 3

Ressources Azure : architecture, reconnaissance et discovery (AzureHound / BloodHound), API ARM, CLI az et portails, machines virtuelles / VDI, conteneurs et registres, key vaults, app services, réseau (filtrage, interconnexions) et stockage, mouvements latéraux et post-exploitation, étude de la journalisation. Azure DevOps : architecture, concepts CI/CD, gestion des accès.

Jour 4

Azure DevOps : implémentation des agents, injection de pipelines, élévation de privilèges et post-exploitation (persistance et extraction de secrets). Intune : relations avec Entra ID, processus d'enrôlement, gestion des accès Intune vs. Entra ID, implémentation des services sur les appareils, post-exploitation (déploiement de scripts et applications), outillage furtif.

Jour 5

Environnements hybrides : méthodes de synchronisation (PHS, PTA, fédération), rebonds on-premise vers Intune et vice-versa, AZURESSO, ADFS, implémentation des agents Entra Connect/Cloud Sync, vol de cookies.

Kubernetes Intrusion Tactics

2 jours | Niveau junior

Description

Kubernetes est aujourd'hui une technologie omniprésente dans les infrastructures modernes. Utilisé pour automatiser le déploiement, la mise à l'échelle et la gestion des applications, il s'est imposé comme un pilier des environnements DevOps et cloud-native. Sa complexité et la diversité de ses composants en font cependant une surface d'attaque riche et souvent mal comprise, exposant les organisations à de nouvelles formes de vulnérabilités et de compromissions.

Au cours de cette formation de deux jours, les participants découvriront les mécanismes internes de Kubernetes et apprendront à en exploiter les failles dans une approche offensive. De l'énumération des composants à l'évasion de conteneurs, en passant par l'escalade de privilèges et les mouvements latéraux, la formation combine théorie et mise en pratique sur un cluster vulnérable. Elle s'adresse aux auditeurs, red teamers et équipes sécurité souhaitant comprendre les risques concrets liés à Kubernetes et renforcer la sécurité de leurs environnements.

- 2 jours (14 heures)
- Module de cours complet et environnement de pratique individuel
- 30% théorie / 70% pratique

Public et prérequis

Cette formation est adaptée pour des personnes ayant des notions de sécurité offensive, mais pas d'expérience préalable sur la technologie Kubernetes. Elle s'adresse principalement aux pentesteurs, administrateurs systèmes, architectes sécurité et développeurs, mais également à tout profil technique souhaitant enrichir son parcours professionnel avec une composante sécurité.

- Pentesteurs
- Administrateurs systèmes
- Architectes sécurité
- Développeurs

De bonnes connaissances réseau et Unix sont recommandées.

Contenu

Jour 1

Architecture et surface d'attaque : rappels sur les conteneurs (Docker, OCI), composants Kubernetes (API Server, etcd, Scheduler, Controller Manager, kubelet), topologie typique d'un cluster (control plane, worker nodes), services internes et exposés (kube proxy, dashboard, API externes), concepts réseau (ingress, pod to pod, CNI, politiques). Premières interactions : reconnaissance, authentification (mot de passe, certificats, tokens) et autorisations (node, ABAC, RBAC, WebHook), utilisation de la CLI kubectl.

!

Jour 2

Concepts avancés : pod templates et contrôleurs, escapes (namespaces, PSP, PSA), mouvements latéraux. Exercices pratiques et scénarios réalistes : exploitation d'un cluster vulnérable.

AWS Intrusion Tactics

2 jours | Niveau junior

Description

AWS est aujourd'hui une plateforme incontournable pour héberger et scaler les applications et les services des entreprises. Sa richesse fonctionnelle et la diversité des services proposés (compute, storage, serverless, managed services) créent une surface d'attaque importante et des modes d'exploitation spécifiques que les attaquants et les auditeurs doivent maîtriser.

Au cours de cette formation dédiée à AWS, les participants découvriront l'architecture AWS, les mécanismes d'IAM, les vecteurs d'accès courants (métadonnées EC2, abus de permissions, buckets S3 publics), ainsi que l'exploitation des services serverless (Lambda) et des mécanismes réseau (VPC, peering, security groups). La formation couvre la reconnaissance, l'accès initial, l'escalade de privilèges, les mouvements latéraux et la persistance et l'exfiltration, avec une approche minimisant l'empreinte pour assurer la discrétion. Pédagogie 100 % pratique avec laboratoires sur environnements AWS, outillage public, études de cas et exercices dirigés, destinée aux pentesters, auditeurs et équipes sécurité souhaitant renforcer leurs capacités d'évaluation offensive sur AWS.

- 2 jours (14 heures)
- Module de cours complet et environnement de pratique individuel
- 30% théorie / 70% pratique

Public et prérequis

Cette formation est adaptée pour des personnes ayant des notions de sécurité offensive, mais pas d'expérience préalable sur les environnements AWS. Elle s'adresse principalement aux pentesteurs, administrateurs systèmes, architectes sécurité et développeurs, mais également à tout profil technique souhaitant enrichir son parcours professionnel avec une composante sécurité.

- Pentesteurs
- Administrateurs systèmes
- Architectes sécurité
- Développeurs

De bonnes connaissances réseau et Unix et des notions d'intrusion web sont recommandées.

Contenu

Jour 1

Fondamentaux, reconnaissance et accès initial : architecture (organisation, comptes), IAM (types d'identité, assumation de rôle, politiques), utilisation de la CLI aws, méthodes de reconnaissance des services, énumération non-authentifiée des identités.
Exploitation et mouvements latéraux : abus de droits sur les buckets S3, EC2 (metadata, mouvements latéraux et empoisonnement des agents SSM), Cognito (user et identity pools) élévation de privilèges IAM.

!

Jour 2

Concepts avancés : Lambdas (runtime API, persistance, exfiltration de données), reconnaissance réseau (VPC, network ACL, security groups), persistance (modification de politiques IAM, role chain juggling), analyse de logs et détection (CloudTrail, CloudWatch, GuardDuty).
Exercices pratiques et scénarios réalistes : exploitation d'un environnement vulnérable.

Attacking Web Applications

5 jours | Niveau intermédiaire

Description

Les applications web représentent une grande partie de la surface d'attaque exposée sur Internet. Au fil de l'évolution des technologies, de nouvelles vulnérabilités et méthodes d'exploitation continuent d'apparaître, complexifiant par conséquent les étapes d'intrusion.

Lors de cette formation de cinq jours, les participants seront amenés à étudier le fonctionnement des mécanismes de sécurité implémentés dans les applications web récentes. Les différents exercices issus du retour d'expérience de nos experts leur permettront d'affiner leurs méthodes d'intrusion pour l'exploitation de vulnérabilités complexes. Enfin, les apprenants pourront appréhender les spécificités des langages et frameworks Java, PHP, Python et ASP.NET, à l'aide de modules dédiés.

- 5 jours (35 heures) modulaires (découpage possible)
- 9 modules de cours dont Java, PHP, Python et ASP.NET
- Plus de 30 exercices pratiques

Public et prérequis

Cette formation est adaptée pour des personnes ayant une expérience préalable en techniques intrusion web. Elle s'adresse principalement aux pentesteurs et développeurs.

- Pentesteurs
- Développeurs

De bonnes connaissances réseau et Unix sont également recommandées.

Contenu

Jour 1

BurpSuite : utilisation avancée, limitations, raccourcis et mécanismes d'automatisation, extensions (AuthMatrix, Hackvertor, ActiveScan++). Reconnaissance : énumération DNS, vhosts, fuzzing, identification des composants web.

Jour 2

Mécanismes de sécurité fondamentaux : authentification (OAuth, JWT, SAML), gestion des sessions (cookies, tokens, viewstates), réinitialisation de mot de passe, contrôle d'accès, gestion des entrées utilisateur. Exploitation avancée : XXE, SSRF, injections, SSTI, prototype pollution, attaques cryptographiques, GraphQL, spécificités des environnements cloud.

Jour 3

Java : reconnaissance et identification de frameworks (extensions, endpoints, en-têtes, interfaces d'administration), exploitation de vulnérabilités spécifiques (XXE, injections HQL, désérialisation, expression languages, JNDI, path traversals).

Jour 4

PHP : reconnaissance et identification de frameworks (endpoints, erreurs, en-têtes), fonctions de sécurité (gestion des sessions, sanitization), exploitation de vulnérabilités spécifiques (type juggling, stream wrappers et filtres, désérialisation et conception de POP chains complexes, XXE), post-exploitation (exécution fileless, contournements de disable_functions).

Jour 5

Python Django : exposition de la surface d'attaque (mode debug, signature des cookies, injection de templates DTL et Jinja2). ASP.NET : rappels fondamentaux, reconnaissance, exploitation de comportements spécifiques (désérialisation, ViewState, Web.config, SSTI (Razor), XXE).

Practical Web 0-Day Hunting

5 jours | Niveau avancé

Description

La complexité des applications web modernes nécessite une forte compréhension des mécanismes natifs des langages utilisés. Les méthodes d'analyse de code source permettent d'optimiser la recherche de vulnérabilités lors d'une intrusion.

Au cours de cette formation de cinq jours, vous acquerrez les compétences nécessaires à l'identification de vulnérabilités complexes au sein du code source d'applications PHP, Java et .NET. En s'appuyant sur de nombreux cas pratiques sur des frameworks répandus tels que Spring ou Symfony, les participants apprendront à optimiser leur recherche à l'aide d'outils d'analyse statique et dynamique.

- 5 jours (35 heures) modulaires (découpage possible)
- 3 modules couvrant les spécificités des langages PHP , Java et .NET.
- Cas pratiques sur des vulnérabilités 1-day
  - Patch diffing
  - Analyse teintée (CodeQL / Semgrep)
  - Instrumentation et débogage
  - Étude des principaux frameworks (Symfony, Spring, etc.)
  - Découverte des vulnérabilités et création d'exploits fonctionnels

Public et prérequis

Cette formation est adaptée pour des personnes ayant de bonnes connaissances des technologies web et des vulnérabilités associées. Elle s'adresse principalement aux pentesteurs et développeurs souhaitant améliorer leur méthode de recherche.

- Pentesteurs
- Développeurs

De bonnes connaissances en sécurité des applications web sont nécessaires (OWASP top 10). Des notions de réseau et Unix sont également recommandées.

Contenu

Jour 1

Méthodologie : approches top-down, bottom-up et hybrides, analyses statique et dynamique, outillage, analyse de l'architecture de l'application et son environnement. PHP : rappels, fonctionnement du langage, mécanismes de sécurités, pièges, étude de frameworks connus et analyse de leurs mécanismes de défense et des fonctionnalités menant à des vulnérabilités, mise en place de l'environnement d'analyse (IDE, Xdebug, configuration PHP, Semgrep).

Jour 2

PHP : vulnérabilités classiques et spécificités d'exploitation liées à PHP, fonctions menant à des vulnérabilités et méthodologie de recherche. Seront notamment étudiées : les injections SQL, les exécutions de code, les type juggling, la déserialisation, les wrappers et les filtres.

Jour 3

Java : étude d'applications Java classiques, structure d'une application (composants Class, JAR, JSP, configurations), formats (WAR, EAR), configuration web.xml (mapping URI, filtres, hooks, contraintes de sécurité), application des approches top-down et bottom-up, outillage. Spécificité des serveurs web : Tomcat, Jetty, WebLogic, Glassfish, WildFly. Instrumentation et analyse de code Java : mise en place d'un environnement d'audit de code, utilisation d'un IDE, débogage, instrumentation, décompilation (jd-gui, procyon), analyse teintée à l'aide de CodeQL, de l'exécution simple de l'outil à l'écriture de requêtes spécifiques.

Jour 4

Java Spring : injections de dépendances, de Beans, de Controllers, de Mappings et d'Annotations. Analyses de vulnérabilités classiques et spécificités d'exploitation liées à Java : LFI, IDOR, XXE et désérialisation. Mécanismes mis en place par Java pour empêcher les vulnérabilités de désérialisation (JEP 290, JEP 396) ainsi que leurs contournements.

.NET : environnement .NET, .NET Framework, .NET Core, ASP.NET et leurs spécificités. Serveur IIS : fonctionnement, configuration, architecture, analyse d'un déploiement classique et des points clés à auditer. Étude d'applications web en .NET afin de mettre en place un environnement d'audit de code, les manières de les décompiler et de trouver les éléments clés pour l'identification de vulnérabilités.

Jour 5

.NET déserialisation et marshalling : fonctionnement, configuration et exploitation. Comprendre les gadgets existants pour savoir en trouver de nouveaux. .NET Remoting : identification des configurations vulnérables.

Offensive CI/CD

2 jours | Niveau intermédiaire

Description

Les environnements d'intégration continue et de déploiement continu (CI/CD) sont devenus essentiels au développement logiciel moderne. Les développeurs s'appuyant fortement sur l'automatisation pour la création, le test et le déploiement d'applications, ces plateformes sont devenues des cibles de premier plan. Sécuriser ces environnements et comprendre leurs vulnérabilités représente aujourd'hui un défi majeur pour les professionnels de la sécurité et les attaquants modernes.

Durant cette formation de deux jours, les participants découvriront une méthodologie offensive complète ciblant les deux solutions CI/CD les plus populaires : GitHub et GitLab. La formation est divisée en deux modules distincts, explorant en profondeur les concepts architecturaux, l'exploitation abusive des pipelines, le détournement de runners et la post-exploitation. À travers des environnements d'entreprise réalistes, les participants s'exerceront à des chaînes d'attaque complexes, allant de l'accès non authentifié aux dépôts via l'injection dans les pipelines jusqu'à la compromission complète de l'organisation et l'extraction de secrets.

- 2 jours (14 heures)
- 2 modules de formation intensifs suivant les étapes réalistes d'une intrusion CI/CD
- Environnements d'entreprise dédiés simulant des organisations GitHub et GitLab complexes
- Analyse approfondie des vulnérabilités réelles et des outils d'exploitation

Public et prérequis

Cette formation s'adresse aux personnes ayant des notions de sécurité offensive et souhaitant comprendre comment attaquer et sécuriser les environnements CI/CD. Elle est principalement destinée aux auditeurs, ingénieurs DevSecOps, administrateurs système et architectes de sécurité, mais aussi à tout profil technique souhaitant enrichir son parcours professionnel.

- Pentesteurs / Red Teamers
- Ingénieurs DevOps / DevSecOps
- Administrateurs système
- Architectes de sécurité

Une bonne maîtrise d'UNIX/Linux, de solides connaissances en réseaux et un intérêt marqué pour la sécurité offensive sont fortement recommandés.

Contenu

Jour 1 : GitHub

Architecture et gestion des identités et des accès (IAM) : Concepts fondamentaux des organisations GitHub, des dépôts et de la gestion des identités (utilisateurs, agents et applications GitHub). Authentification et autorisation : Jetons d'accès personnels (PAT) et étendues, clés SSH et système de rôles/ACL (qui peut créer des branches, effectuer des commits, des fusions ou consulter les journaux). Pipelines : Fonctionnement interne de GitHub Actions, définitions YAML, modules disponibles et stockage des variables (gestion des secrets et implications des branches protégées). Exploitation des pipelines : Injection de code malveillant via des entrées non fiables, injections d'expressions, artefacts dangereux et manipulation de l'environnement. Exploration des vecteurs d'attaque dans la chaîne d'approvisionnement, tels que le détournement de dépôts et l'abus de Dependabot. Runners et OIDC : Runners hébergés vs auto-hébergés. Architecture, enregistrement et techniques de détournement des Runners. Fonctionnement interne d'OIDC, génération de jetons et exploitation des liaisons de revendications faibles. Outils et automatisation : Utilisation d'Octoscan pour l'analyse statique des vulnérabilités et de NordStream pour l'extraction automatisée des secrets CI/CD.

Jour 2 : GitLab

Architecture et ACL : hiérarchies de projets et de groupes spécifiques à GitLab, et modèles d'autorisation avancés. Mécanismes d'authentification : flux d'identifiants/mots de passe, jetons d'accès personnels (PAT) et étendues, clés SSH et fournisseurs d'identité externes. Pipelines : configuration YAML de GitLab CI/CD, variables d'environnement spécifiques et implications de sécurité des variables et branches protégées. Implémentation et détournement des runners : enregistrement de l'agent GitLab, authentification, hiérarchie du système de fichiers et modes d'exécution (par exemple, Shell, Docker, Kubernetes). Techniques de détournement, d'exploitation et de pivotement à partir de ces runners (persistance, vol d'identité du runner, exploitation de Docker-in-Docker). Post-exploitation : interaction avec la console Ruby de GitLab, extraction avancée de secrets et accès direct aux dépôts depuis le système de fichiers sous-jacent. Défense et opérations : analyse des journaux de pipeline, identification des alertes de sécurité et suivi des activités CI/CD malveillantes.

Attacking Android Applications

2 jours | Niveau junior

Description

Android est l'un des systèmes d'exploitation pour mobile les plus répandus sur le marché et sur lequel de nombreuses applications sont développées. Cet écosystème définit des normes d'implémentation, de communication, de stockage et des mécanismes de sécurité qui lui sont propre et que les développeurs doivent respecter.

Au cours de cette formation de deux jours, les participants découvriront les spécificités d'implémentation des applications Android et étudieront les méthodologies et techniques employées pour les analyser.

- 2 jours (14 heures)
- 5 modules de cours
- 9 applications Android avec des exercices pratiques

Public et prérequis

Cette formation est adaptée pour des personnes ayant des notions de sécurité offensive mais pas d'expérience préalable sur l'audit d'applications Android. Elle s'adresse principalement aux pentesteurs et développeurs Android.

- Pentesteurs
- Développeurs Android

Des notions de sécurité offensive et des connaissances réseau et Unix sont recommandées.

Contenu

Jour 1

Fondamentaux : fonctionnement d'une application et de l'écosystème Android (services, intents, keystore, format des APK, fichier de cache, shared prefs, mécanisme de backup). Analyse statique : analyse des permissions et des interactions avec le système et les autres applications, présentation des outils d'analyse et explication des artefacts courant donnant de l'information sur les activités d'une application.

Jour 2

Analyse dynamique : architecture d'une application au runtime, mécanisme d'interception et d'instrumentation de code Java, présentation de Frida et d'Objection pour automatiser des contournements classiques ou obtenir de l'information. Cas pratiques : mise en applications sur des applications Android.

Password Cracking

1 jour | Niveau junior

Description

Les mots de passe constituent encore aujourd'hui un composant essentiel de la sécurité des systèmes d'informations. Lors des intrusions, différents types d'empreintes de mots de passe sont récupérés et pouvoir les casser dans un temps restreints peut s'avérer décisif.

Cette formation a pour objectif de présenter les techniques et les outils permettant de casser le plus rapidement des empreintes de mots de passe. Un historique des évolutions du stockage des mots de passe sera également présenté, afin de mettre en lumière les mauvais exemples et erreurs commises dans des projets répandus.

- 1 jour (6 heures)
- Techniques d'optimisation du cassage de mots de passe
- Datasets fournis

Public et prérequis

Cette formation est adaptée pour des personnes n'ayant pas de connaissances préalables sur le cassage de mots de passe. Elle s'adresse principalement aux pentesteurs, administrateurs systèmes, et développeurs.

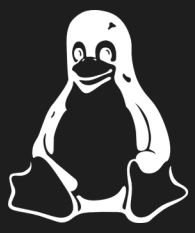
- Pentesteurs
- Administrateurs système
- Développeurs

Contenu

Théorie sur le stockage et la génération de mots de passe : type de stockage, fonctions de hachage, attaques sur les fonctions, génération de candidats, technologies de calcul. Historique des algorithmes. Série d'exercices pratiques : identification des algorithmes dans du code source, prise en main de John the Ripper (modes de génération de candidats, développement de règles de dérivation et de filtres de candidats basés sur une politique de mots de passe, formats dynamiques, implémentation ou modification d'un format natif), prise en main de Hashcat (génération de candidats avancée avec combinaison prince, mutations génétiques siga et génération de règles).

Embedded Linux Exploitation

5 jours | Niveau junior



Description

Obtenez la méthodologie essentielle à l'évaluation de sécurité de systèmes Linux embarqués. Analyse matérielle, émulation, fuzzing et exploitation de vulnérabilités : cette formation couvre le processus complet de recherche de vulnérabilités et constitue un socle indispensable pour tout professionnel de la sécurité offensive.

Objectifs

- Identifier les composants matériels et les interfaces critiques d'un équipement Linux embarqué
- Accéder à l'équipement via l'UART, interrompre son processus de démarrage et extraire son firmware
- Analyser et émuler des composants du firmware en utilisant QEMU et la conteneurisation
- Effectuer une analyse statique et assister la recherche de bugs avec des scripts
- Mettre en place et lancer une campagne de fuzzing avec AFL++ sur un programme compilé
- Analyser et exploiter des vulnérabilités classiques
- Mettre en œuvre des mécanismes de persistance sur un système embarqué compromis

Public et prérequis

Cette formation est destinée aux personnes travaillant dans la sécurité informatique, le développement de systèmes embarqués et intéressées par la sécurité offensive des systèmes Linux embarqués. Les personnes suivant la formation doivent avoir une base solide dans les domaines suivants :

- Ligne de commande Linux : être à l'aise avec la navigation dans le système de fichiers, la gestion des processus et l'utilisation des outils en ligne de commande courants.
- Bases en réseaux : compréhension des concepts TCP/IP de base, des ports et des protocoles comme HTTP.
- Connaissances en programmation : familiarité avec la lecture et l'écriture de scripts simples en Python et une compréhension de base des concepts de programmation en C.
- Concepts de rétro-ingénierie : des connaissances de base en architecture informatique, en langage d'assemblage ARM et une expérience avec un désassembleur.

Contenu

Jour 1

Prise en main et extraction : recherche d'informations en source ouverte, définition de la surface d'attaque et analyse des composants matériels. Compréhension de la séquence de démarrage et première extraction du firmware.

Jour 2

Analyse et émulation du firmware : techniques d'émulation logicielle, configuration de l'environnement et identification des processus cibles à analyser. Extraction et émulation des services logiciels importants à partir du firmware.

Jour 3

Analyse statique et exploitation : analyse statique avec des outils comme Ghidra et semgrep pour l'analyse de différences et la recherche de vulnérabilités. Méthodologie d'exploitation et écriture d'un premier script pour une injection de commande.

Jour 4

Fuzzing : introduction au fuzzing avec AFL++. Création d'un corpus d'entrées, développement d'un harness et lancement d'une campagne de fuzzing sur une cible binaire.

Jour 5

Exploitation avancée et persistance : analyse des crashs issus du fuzzer et identification d'une vulnérabilité exploitable. Construction d'une chaîne ROP, écriture d'un script d'exploitation et étude des techniques de persistance sur le système.

Android for Security Engineers

5 jours | Niveau intermédiaire

Description

Android est l'un des systèmes d'exploitation pour mobile les plus répandus sur le marché. Bien qu'il soit basé sur Linux, il se démarque par des composants spécifiques qui l'éloignent de l'OS traditionnel. Au cours de cette formation, les participants découvriront l'architecture d'Android et les interactions entre ses différents composants internes. Le système permet d'exécuter des applications tierces tout en protégeant les données de l'utilisateur final.

Les composants clés du système seront décortiqués, y compris le processus de démarrage et les mécanismes de sécurité. Les formateurs détailleront les évolutions des versions à partir d'Android 10 et évoqueront certaines particularités des constructeurs. Au cours des chapitres, les notions présentées seront mises en pratique au travers d'exercices concrets.

À la fin de cette formation, les participants auront une compréhension approfondie d'Android et seront en mesure d'être autonomes dans tout travail de recherche sur cet écosystème.

- 5 jours (35 heures)
- 15h théorie / 20h pratique

Public et prérequis

Android for Security Engineers est une formation de niveau avancé conçue pour les ingénieurs sécurité souhaitant mener des travaux de recherche sur ce système.

- Pentesteurs
- Développeurs Android
- Ingénieurs sécurité

De bonnes connaissances en développement C ainsi que des connaissances de base sur les systèmes Linux sont recommandées.

Contenu

Jour 1

Architecture globale d'Android, chaîne de démarrage, système de mise-à-jour, modèle de sécurité et root d'un téléphone.

Jour 2

Format des applications (APK) et présentation des outils de compilation et de debug (exercices avec Frida).

Jour 3

Android Runtime, mécanisme d'IPC (Binder) et présentation de la bibliothèque Bionic (libc Android).

Jour 4

Cycle de vie d'une Application : installation, démarrage, exécution et arrêt. Exploration des traces/logs pouvant être présents sur un appareil. Chiffrement des données utilisateurs.

Jour 5

Exercice final : modification d'un environnement Android via les modules Magisk et mise en pratique des notions vues pendant la semaine. Analyse des spécificités du noyau Linux pour Android.

iOS for Security Engineers

5 jours | Niveau intermédiaire

Description

iOS est un des systèmes d'exploitation les plus répandus sur le marché, offrant un modèle de sécurité à l'état de l'art.

Au cours de cette formation, les participants aborderont l'écosystème et les briques fondamentales du système d'exploitation iOS. Ils découvriront l'utilisation de la chaîne de compilation macOS afin de déployer un programme, puis les outils de débogage et de diagnostic.

Les fondamentaux du reverse-engineering d'applications et des services systèmes seront abordés dans un second temps : le fonctionnement d'Objective-C, les mécanismes d'IPC (mach, XPC, NSXPC) et les API du noyau. Des exemples pratiques et des exercices guideront les participants tout au long du training. Enfin, les mesures de sécurités logicielles et matérielles propres à iOS seront couvertes, tant dans l'espace noyau qu'utilisateur.

- 5 jours (35 heures)
- 18h théorie / 17h pratique

Public et prérequis

iOS for Security Engineers est une formation de niveau intermédiaire, conçue pour les ingénieurs sécurité souhaitant mener des travaux de recherche sur ce système.

- Pentesteurs
- Développeurs iOS
- Ingénieurs sécurité

De bonnes connaissances en développement C et des bases en rétro-ingénierie sont recommandées. Une licence IDA Pro avec le décompilateur Hex-Rays pour ARM64 est un plus.

Contenu

Jour 1

Introduction : présentation de l'environnement de travail, développement sur les plateformes Apple (iOS et macOS), utilisation des outils de diagnostic, introduction à l'écosystème Apple.

Jour 2

Introduction au reverse-engineering sur les plateformes Apple : extraction de mise à jour, formats de fichiers importants et outils, découverte et expérimentation du fonctionnement interne d'Objective-C, introduction au kernel XNU.

Jour 3

Mécanismes Mach : explications et exercices autour de l'API IPC de XNU, présentation et exercices sur l'implémentation de l'API Mach pour l'interaction avec les objets noyau, utilisation de Frida pour instrumenter des services.

Jour 4

Reverse-engineering de services Mach : théorie et exercices pratiques autour de XPC et NSXPC, les abstractions utilisées pour les communications inter-processus. Panorama de l'utilisation des pointeurs signés sur les plateformes Apple.

Jour 5

Sécurité de XNU : présentation du framework MACF, explications du fonctionnement d'AMFI et des politiques d'isolation (sandbox), description des mécanismes de défense en profondeur de XNU, contre-mesures matérielles de sécurité dans le noyau, mitigations des vulnérabilités noyau. Cas d'étude sur l'envoi de données de diagnostiques.

Advanced IDA

5 jours | Niveau avancé

Description

Hex-Rays est un des acteurs majeurs dans le développement d'outils pour la rétro-ingénierie. Leur produit IDA s'est imposé au fil des années comme la référence en la matière. Cependant, le manque de documentation et de ressources rendent parfois difficile son maniement.

L'objectif de cette formation est de se familiariser avec IDA (son interface, ses fonctionnalités, son API et son écosystème) au travers de plusieurs modules théoriques et pratiques. Les participants apprendront également comment développer des scripts et plugins pour étendre les fonctionnalités d'IDA et son décompilateur.

- 5 jours (35 heures)
- 8h théorie / 27h pratique

Public et prérequis

Cette formation de niveau avancé est conçue pour les chercheurs en sécurité et experts en rétro-ingénierie souhaitant changer d'environnement ou se perfectionner dans l'utilisation d'IDA.

- Chercheurs en sécurité
- Experts en rétro-ingénierie

De bonnes connaissances en assembleur (x86-x64, ARM) ainsi qu'en programmation Python sont fortement recommandées. Une licence IDA Pro (non fournie) est indispensable.

Contenu

Jour 1

Introduction à IDA : terminologie, architecture et présentation de l'outil
Découverte du SDK et de l'API Python : notions élémentaires

Jour 2

Prise en main des fonctionnalités disponibles via différents exercices. Analyse statique : désassembleur, FLIRT, IDS, Type Info Library. Analyse dynamique : débuggeur, traceur et instrumentation binaire

Jour 3

Programmation avancée (partie1) : présentation détaillée du SDK et mise en pratique par du scripting pour automatiser les tâches complexes.

Jour 4

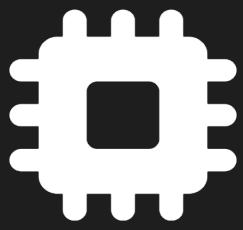
Programmation avancée (partie2) : développement de plugins, loader et extension de processeur pour mettre en pratique les TP précédents.

Jour 5

Extension du décompilateur : présentation de l'API Hex-Rays, manipulation du microcode et de l'AST, extension et amélioration des outils créés lors de la session.

Hardware Intrusion

5 jours | Niveau intermédiaire



Description

L'objectif de cette formation est de monter en compétences sur l'analyse de sécurité hardware. Elle s'adresse autant aux novices qu'à ceux ayant un niveau intermédiaire.

À l'issue de cette formation les étudiants doivent connaître les principes de base en électronique et soudure. Ils sauront reconnaître les différents composants d'un PCB et rechercher des informations pertinentes dans des datasheets de composants type System on Chip (SoC) ou Flash externe pour en tirer parti (mise en RST, fonctionnalité de debug).

Enfin, ils sauront identifier les éventuels points de tests, inférer puis interagir avec les protocoles les plus courants (UART, JTAG/SWD, SDIO, SPI).

Lors de la formation, les étudiants apprendront également à utiliser des matériels et outils utiles à l'analyse (analyseurs logiques & Logic2, sondes à base de FT2232H & OpenOCD/flashrom).

- 5 jours (35 heures)
- 17h théorie / 18h pratique

Public et prérequis

L'initiation à l'intrusion matérielle est une formation de niveau débutant à intermédiaire conçue pour les pentesteurs, les chercheurs en sécurité et les équipes de sécurité.

- Pentesteurs
- Chercheurs en sécurité
- Équipes sécurité

Des connaissances de base en électricité et électronique (savoir se servir d'un multimètre, connaître la loi d'Ohm) sont recommandées.

Contenu

Jour 1

Notions fondamentales sur les composants : PCB, SoC, Flash, résistances, condensateur, transistor, oscillateur à quartz et PMIC.

Jour 2

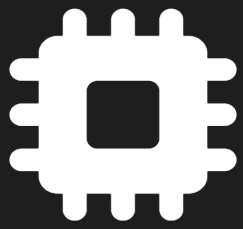
Rappels théoriques : électricité, sécurité, électronique analogique et numérique.

Jour 3 à 5

Protocoles courants : théorie (caractéristiques, variation, utilité dans l'analyse de sécurité, forme du signal) et pratique (identifier les ports intéressants, connaître et savoir utiliser le matériel et les outils permettant de s'y connecter). Soudure : principe, matériel et bonnes pratiques.

DMA Attacks

4 jours | Niveau intermédiaire



Description

L'accès physique à une machine ouvre des vecteurs d'attaque conséquents, notamment via les accès directs à la mémoire (DMA). Ces attaques permettent à un composant matériel externe de lire et d'écrire directement dans la RAM de la cible, contournant ainsi le système d'exploitation, l'authentification et les protections logicielles classiques.

Au cours de cette formation de quatre jours, les participants découvriront l'architecture matérielle des systèmes modernes et apprendront à exploiter les connectiques (PCI Express, M.2, etc.) pour réaliser des dumps de mémoire. La formation couvre l'utilisation d'outils matériels spécifiques (PCIScreamer, USB3380) et le framework PCILeech pour injecter du code ou contourner des authentifications à la volée. Un pan important de la formation est également dédié à la rétro-ingénierie pour l'écriture de signatures personnalisées, ainsi qu'à l'analyse forensic des dumps mémoire à l'aide de Volatility.

- 4 jours (28 heures)
- Manipulation de matériel spécifique d'interception (PCIScreamer, USB3380)
- Répartition équilibrée entre l'exploitation offensive (PCILeech) et l'analyse de mémoire (Volatility)
- Exercices pratiques sur cibles Windows et Linux

Public et prérequis

Cette formation s'adresse aux profils techniques cherchant à comprendre et exploiter les vulnérabilités matérielles liées aux accès mémoire physiques.

- Pentesteurs / Red Teamers
- Chercheurs en sécurité et développeurs bas niveau
- Analystes Forensic et Réponse aux Incidents (CSIRT)

Une bonne connaissance de l'architecture des systèmes d'exploitation (Windows/Linux) et de la ligne de commande est requise. Des bases en rétro-ingénierie (utilisation basique d'IDA ou Ghidra) et en analyse de mémoire constituent un avantage significatif la réalisation des travaux pratiques.

Contenu

Jour 1

Architecture de l'ordinateur : périphériques, bus de données, composants clés, connectiques, types de RAM (fréquences, dual/triple channel). Concepts DMA : historique, types d'accès, vecteurs d'attaque modernes, connectiques hotplug. Mécanismes de sécurité : protections au niveau du BIOS/OS, détection de présence de sécurités (IOMMU). Contournement des protections : effacement du mot de passe BIOS (clear CMOS, mots de passe constructeur, bruteforce), vulnérabilités liées à l'IOMMU. Matériel de capture : présentation du PCIScreamer et de l'USB3380, méthodologie de connexion selon la cible, gestion des problèmes matériels (interférences, compatibilité). Labs : réalisation de dumps physiques de la mémoire vive via PCI Express, M.2 et USB3380.

Jour 2

PCILeech : fonctionnement interne et utilisation de l'outil. Exploitation : extraction de la RAM, manipulation de la mémoire, contournement d'authentification à la volée, injection de shellcode kernel. Rétro-ingénierie (Signatures) : présentation des mécanismes d'authentification Windows et Linux, identification des structures à modifier. Outils d'analyse : introduction à IDA/Ghidra pour analyser et patcher ces mécanismes. Labs : écriture d'un fichier de signature (.sig) personnalisé pour contourner avec succès l'authentification d'un système Linux cible.

Jour 3

Post-exploitation classique : dump de disques physiques depuis un accès administrateur (Linux/Windows), limites techniques. Analyse de la RAM : introduction au framework Volatility (v2 vs v3), installation, fonctionnement interne, génération de profils/images pour différents OS. Reconnaissance : méthodes pour déterminer la version et le type de Linux (via le réseau, via un dump mémoire). Labs : réalisation d'un dump de disque depuis des accès Linux et Windows, analyse d'un dump de RAM Windows avec Volatility (traque et recherche de processus malveillants).

Jour 4

Approfondissement Volatility : réveil pédagogique, rappels sur la génération complexe de fichiers images (profils d'OS). Labs d'investigation : fingerprinting complet d'OS (réseau et mémoire), création de A à Z d'un fichier de signature Volatility pour une cible Linux spécifique, analyse approfondie d'un dump de RAM Linux pour identifier un processus malveillant persistant. Conclusion : synthèse des points clés et des vecteurs d'attaque. Évolution des défenses : points à surveiller chez les éditeurs, déploiement de DMA Guard, intégration native et systématisation de l'IOMMU dans Windows.

Windows Forensics

5 jours | Niveau intermédiaire

Description

L'investigation numérique permet de reconstruire et comprendre de manière détaillée la chronologie des activités présentes et passées d'un système. Dans le cas de cette formation, nous nous intéressons au système d'exploitation Windows 10 ou 11. Qu'il s'agisse d'un incident de sécurité ou d'une recherche de malveillance informatique, les premières réponses visent à établir le périmètre de compromission et le mode opératoire de l'attaquant. La démarche technique présentée se veut la plus exhaustive possible et reproductible.

Au cours de ces cinq jours de formation, il sera exposé aux différents participants les fondamentaux à connaître afin de mener une investigation numérique pour Windows et ainsi identifier les traces d'une malveillance. Chaque module sera illustré par des travaux pratiques guidés permettant d'appliquer les notions théoriques enseignées préalablement. Enfin, la formation se conclura par une mise en situation sur plusieurs traces (disque, mémoire, pcap).

Cette formation est focalisée sur le poste de travail et n'intègre pas la dimension entreprise comme Azure/AD (une autre formation abordera prochainement cet aspect).

- 5 jours (35 heures)
- 11 modules de cours couvrant les fondamentaux de l'investigation Windows
- Approche à froid ou à chaud pour couvrir plusieurs cadres d'intervention
- Travaux dirigés sur des artefacts afin d'illustrer au mieux la théorie

Public et prérequis

Cette formation a été conçue pour des personnes ayant une première expérience sur la compréhension des environnements Windows (administration, troubleshooting, utilisation avancée) et désirant aller plus loin dans le domaine de l'investigation numérique. Elle nécessite une maîtrise basique de l'environnement Linux : ce système étant utilisé pour mener certaines investigations.

- Utilisateurs avancés (développeurs)
- Administrateurs système
- Analystes SOC niveau 2 ou d'une équipe de cybersécurité
- Analystes forensique débutants

Des notions de sécurité offensive et de bonnes connaissances Windows & Unix sont un plus.

Contenu

Jour 1

Prises en main : prise en main de l'environnement de formation (machine virtuelle, système Linux). Rappels de l'utilisation de la ligne de commande Linux. Windows : Description du fonctionnement de Windows (historique de Windows, processus, services, drivers, fichiers, modèle de sécurité, pile réseau, principales attaques). Évènements Windows : description du modèle de journalisation Windows et des évènements à connaître par cas d'usage. Mise en situation sur des fichiers d'évènements.

Jour 2

NTFS : étude du système de fichiers privilégié de l'environnement Windows. MFT, Journal des USN et autres fichiers spéciaux. Décodage des dates et fichiers supprimés. Reconstituer la chronologie des événements et pivoter sur un élément (date, IOC).
Base de registre : contenu des bases de registre. Cas d'usage et configuration du système Windows. Mécanismes de persistance : les moyens de persistance privilégiés par un attaquant sont passés en revue et ainsi identifient les programmes malveillants exécutés par un attaquant.

Jour 3

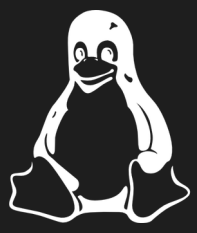
Exécution de commandes : traces liées à l'exécution de commande à distance sur le poste au travers des différents protocoles Windows (WinRM, PsExec, WMI, RPC). Codes et fichiers malveillants : outils et méthodes d'analyse permettant de mener une première étude sur un code malveillant et ainsi extraire les informations d'intérêt (comportement, IOC). Par extension les fichiers pouvant embarquer une charge malveillante sont également étudiés. Protocoles réseau : une attention particulière est proposée afin d'identifier les communications réseau inhabituelles d'un système Windows ainsi que la caractérisation de certaines attaques (tunnel DNS, TOR).

Jours 4 et 5

Artefacts : l'étude des artefacts forensic les plus importants (prefetch, srum, amcache, navigation) afin de compléter la chronologie de la malveillance. Les méthodes de collecte et d'acquisition sont également présentées afin de mettre à disposition les fichiers à étudier par l'analyste (DFIR ORC). Analyse mémoire : les techniques d'acquisition et d'identification d'éléments suspects sont abordées afin de compléter l'analyse des éléments hors ligne. Processus en cours d'exécution, connexions réseau, fichiers en cache, injections mémoire et API hooking. Étude de cas : plusieurs images sont proposées aux participants afin de mettre en pratique l'ensemble des techniques étudiées durant les 5 jours. Ces images regroupent des données variées comme une image disque, une capture mémoire et des captures réseau.

Linux Forensics

5 jours | Niveau intermédiaire



Description

L'investigation numérique permet de reconstruire et comprendre de manière détaillée la chronologie des activités présentes et passées d'un système. Dans le cas présent, nous nous intéressons au noyau Linux et deux types de distribution Linux. Les exemples et illustrations sont retenus pour des distributions sur des bases apt et rpm, néanmoins la plupart des éléments présentés dans la formation peuvent être généralisés.

Lors d'un incident de sécurité ou d'une recherche de malveillance informatique, les premières questions posées concernent à établir le périmètre de compromission et le mode opératoire de l'attaquant. La démarche technique d'une telle investigation se veut la plus exhaustive possible et surtout reproductible.

Au cours de ces cinq jours de formation, il sera exposé aux différents participants les fondamentaux à connaître afin de mener une investigation numérique pour une distribution Linux et ainsi identifier les traces d'une malveillance. Chaque module sera illustré par des travaux pratiques guidés permettant d'appliquer les notions théoriques enseignées préalablement. La formation inclut une mise en situation sur plusieurs artefacts (disque, mémoire, pcap).

- 5 jours (35 heures)
- 12 modules de cours couvrant les fondamentaux de l'investigation Linux
- Approche à froid ou à chaud pour couvrir plusieurs situations
- Travaux dirigés sur des artefacts afin d'illustrer au mieux la théorie

Public et prérequis

Cette formation a été conçue pour des personnes ayant une première expérience sur la compréhension des environnements Linux (administration, troubleshooting, utilisation avancée) et désirant aller plus loin dans le domaine de l'investigation numérique.

- Utilisateurs expérimentés
- Administrateurs système
- Analystes SOC niveau 2 ou équipe de cybersécurité
- Analystes forensique débutants

Des notions de sécurité offensive et de bonnes connaissances Unix sont un plus pour la compréhension de cette formation.

Jour 1

Prises en main et la ligne de commande : prise en main de l'environnement de formation (machine virtuelle, système Linux). Rappel des principales commandes pour Linux. Linux et distribution : description du fonctionnement de Linux dont les processus, file descriptors, modèle de sécurité (user/group, ACL, cgroup), les canaux nommés, signaux, terminal et interpréteur de commande, X11. Système de fichiers : principaux types de système de fichiers rencontrés dans les systèmes Linux (ext4, LVM, XFS). Caractéristiques et particularités pour le forensic : gestion des dates, fichiers effacés, métadonnées, etc. Cas de LUKS et des disques virtuels (qcow, vmdk).

Jour 2

Séquence de démarrage : identifier la séquence de démarrage afin de vérifier l'intégrité de la chaîne de lancement (grub, initramfs, cas UEFI). Recherche de backdoor sur Systemd. Cas du SecureBoot et de la signature des modules noyaux. Gestion des programmes : contrôle des programmes installés sur le système (intégrité, permissions). Format ELF : programme et bibliothèque. Utilisation des gestionnaires de paquets apt et rpm. Journalisation : type de journaux (/var/log) et processus associés (syslog, auditd). Traces de compromission.

Jour 3

Mécanisme de persistance : moyens de persistance système et utilisateur, gestionnaire des périphériques, Systemd. Analyse des processus : outils de diagnostique des processus, principaux processus Linux (ssh, X11), exécution à distance, procfs. Analyse réseau : configuration réseau, outils de diagnostic réseau, socket réseau, protocole généralement rencontré et tunnel, capture réseau.

Jour 4

Codes malveillants : outils et méthodes d'analyse permettant mener une première étude sur un code malveillant et ainsi extraire les informations d'intérêt (comportement, IOC). Artefact : autres artefacts (coredump, viminfo). Analyse mémoire : les techniques d'acquisition et d'identification d'éléments suspects sont abordées afin de compléter l'analyse des éléments hors ligne. Processus en cours d'exécution, connexion réseau, fichiers en cache, injection mémoire et API hooking.

Jour 5

Conteneur : recherche de traces dans la conteneurisation et reconstitution du système. Collecte de données : extraction de fichiers (copie de disque) et sélective (velociraptor). Étude de cas : plusieurs images sont proposées aux participants afin de mettre en pratique l'ensemble des techniques étudiées durant les 5 jours. Ces images regroupent des données variées comme une image disque, une capture mémoire et des captures réseau.

Mobile Forensics

5 jours | Niveau junior

Description

Le téléphone mobile se transforme depuis plusieurs années comme le prolongement du poste de travail et devient une cible privilégiée, car au plus près de la donnée. L'investigation numérique de ce type de dispositif vise à identifier des traces en lien avec des activités criminelles, à détecter des traces d'activités malveillantes et de compromission du téléphone mobile.

Cette formation vise à présenter les principaux artefacts présents sur les environnements Android et iOS, majoritaires sur le marché, et de disposer d'une trousse à outils open source afin de les analyser. Des méthodologies d'analyse adaptées seront présentées afin de pallier l'approche « boîte noire » de certains systèmes et de leurs applications pré-installées qui complexifient l'audit du téléphone.

Cette formation aborde exclusivement le cas où les secrets de déverrouillage du téléphone sont connus.

- 5 jours (35 heures)
- 2 systèmes d'exploitation mobile : Android (≥ 10) & iOS (≥ 14)

Public et prérequis

Cette formation est adaptée pour des personnes ayant des notions en sécurité ou en administration de système Linux. Elle s'adresse principalement aux équipes informatiques souhaitant disposer de méthodes de premier niveau dans l'investigation de téléphones et ne disposant pas d'un logiciel dédié à cette activité. Plus généralement, toute personne souhaitant enrichir son parcours professionnel avec une composante sécurité dans le domaine mobile.

- Équipes informatique
- Administrateurs système
- Équipes sécurité

Des notions de sécurité offensive et de bonnes connaissances Unix sont un plus pour la compréhension de cette formation.

Un iPhone et un téléphone Android sont fournis durant la formation pour les manipulations.

Jour 1

Introduction et fondamentaux : objectifs, prise en main et panorama de l'investigation mobile (sources d'information, principaux formats d'intérêt, gestion de l'horodatage). Fondamentaux iOS : représentation de l'architecture et des principaux services, modèle de sécurité, systèmes de fichiers et formats de données spécifiques. Acquisition iOS : méthodes d'acquisition illustrées par des TP d'acquisition complète (accès root au terminal) et partielle (accès restreint) en exploitant des outils de référence tels que libimobiledevice, mvt et la génération de sysdiagnose.

Jour 2

Artefacts systèmes iOS : revue de l'activité de l'ensemble du téléphone pour lister les traces d'exécution et la présence d'applications (info.plist, base LSD), analyser leurs activités (netusage, powerlogs, KnowledgeC, Biome) et leurs permissions (TCC). Analyse approfondie des journaux issus des archives avec l'outil sysdiagnose.py. Artefacts applicatifs iOS : présentation des applications natives et tierces et étude de leurs cas d'usage courants (comptes, communications, navigation web). Les travaux pratiques s'appuient principalement sur iLEAPP et Apollo pour extraire et analyser ces données spécifiques.

Jour 3

Méthodes d'analyse : capture réseau, sauvegarde chiffrée et sysdiagnose. Fondamentaux Android : architectures, OEM, modèle de sécurité, systèmes de fichiers, avec une présentation détaillée des outils ADB (Android Debug Bridge) et mvt-android. Acquisition Android : méthodes d'acquisition mises en pratique via des TP dédiés au rootage des terminaux, à l'acquisition complète, et à l'acquisition partielle ciblant la collecte d'artefacts spécifiques via ADB.

Jour 4

Artefacts systèmes Android : revue de l'activité du téléphone et étude des principaux formats de données (ABX, bases SQLite, Protobuf). Analyse des artefacts permettant de lister les applications, leurs permissions et leurs activités, suivie d'un TP d'analyse live exploitant les éléments collectés avec MVT et ADB. Artefacts applicatifs Android : présentation de l'arborescence classique des applications natives et tierces, méthodes pour retrouver et parser leurs données internes. Étude des cas d'usage (comptes, communications, web) avec l'utilisation d' ALEAPP comme fil rouge des travaux pratiques.

Jour 5

Applications malveillantes Android : analyse de la menace mobile et présentation des méthodes d'injection courantes. Analyse statique et dynamique : méthodologie pour analyser la structure d'un APK suspect à l'aide d'outils de rétro-ingénierie (apktool, mobsf, jadx) et exécution dynamique au sein d'un émulateur couplée à des captures réseau. Pratique : travaux pratiques dédiés à la manipulation et à l'analyse complète d'un malware Android connu, incluant des phases d'analyse live avec ADB.

Kubernetes Forensics

3 jours | Niveau intermédiaire

Description

Kubernetes s'est imposé comme la solution d'orchestration des infrastructures cloud-native modernes, automatisant le déploiement et l'expansion d'applications à grande échelle. Cette omniprésence en fait désormais un terrain d'investigation incontournable pour les analystes forensiques. L'architecture distribuée de la plateforme, la multiplicité de ses composants et le caractère volatile de ses charges de travail complexifient considérablement la collecte et la préservation des artefacts numériques.

En pratique, Kubernetes est souvent exploité au travers de services managés proposés par les principaux fournisseurs cloud, AWS, Azure et GCP. Cette formation intègre les spécificités de ces environnements, tout en explorant le fonctionnement interne de Kubernetes de manière agnostique, indépendamment de la plateforme.

La formation abordera dans un premier temps les fondamentaux de la conteneurisation et l'architecture type d'un cluster Kubernetes. Plusieurs approches d'analyse forensique seront ensuite présentées : certaines s'appuient sur le socle système des nœuds du cluster, tandis que d'autres exploitent les mécanismes de journalisation et les outils natifs mis à disposition par les fournisseurs cloud.

- 3 jours (21 heures)
- Analyse de conteneurs et de clusters Kubernetes

Public et prérequis

Cette formation est adaptée pour des personnes souhaitant découvrir l'analyse forensique Kubernetes tout en approfondissant leurs connaissances en analyse de conteneurs. Elle s'adresse à toutes les personnes amenées à administrer ou investiguer des clusters Kubernetes.

- Administrateurs système
- Analystes SOC / CERT
- DevOps / DevSecOps

De bonnes connaissances de Linux et de l'environnement shell sont fortement recommandées.

Jour 1

Introduction : prise en main et panorama de l'investigation Kubernetes. Paysage de la menace Kube et conteneurs : volatilité des informations dans Kube, internals des conteneurs, container runtimes. Fondamentaux Kubernetes : architecture d'un cluster, modèle de permissions, faiblesses de Kubernetes, explorations et identifications de comportements suspects.

Jour 2

Collecte et réponse : checklist des artéfacts à collecter, CRIU et checkpoints, journaux d'audit et isolation de conteneurs. Analyse filesystem et mémoire : analyse de l'overlayfs, récupération de fichiers, analyse de la mémoire avec checkpoint, comparer des images.

Jour 3

Analyse réseau et outils : modèle réseau Kubernetes et implications forensiques, capture du trafic, forensiques DNS et NetworkPolicies, capture de syscalls. Exercice : mise en pratique des connaissances acquises dans une réponse sur incident en condition réelles.

Malware Analysis

3 jours | Niveau intermédiaire

Description

Lors d'incidents de sécurité, la découverte de code malveillant est fréquente. Cette formation vise à fournir les clés de compréhension des logiciels malveillants et d'extraction des éléments pertinents.

Au cours de la formation, différents types de code malveillant sont présentés selon le langage utilisé ou la phase de l'attaque (exploitation, persistance). Les différentes méthodes d'analyse statique et dynamique sont expliquées afin de proposer des approches complémentaires. Une part importante de la formation est consacrée à des exercices pratiques, basés sur des incidents de sécurité ou des procédures opérationnelles courantes observées lors de ces incidents. Ce cours traite exclusivement du code malveillant exécuté dans l'espace utilisateur.

- 3 jours (21 heures)
- Analyse de code malveillant dans différents langages
- Étude des fichiers malveillants (PowerShell, LNK, HTA)

Public et prérequis

Cette formation s'adresse aux personnes ayant déjà une certaine expérience en assembleur (x86) ou en analyse de programmes (observation de logiciels malveillants en environnement isolé). Elle est destinée à toutes les personnes impliquées dans la gestion des logiciels malveillants, notamment les équipes de sécurité (SOC, CSIRT) ou souhaitant approfondir leurs compétences dans ce domaine.

- Analystes SOC
- Analystes CSIRT

Une connaissance de base de Windows/Linux est recommandée pour mieux comprendre le fonctionnement des logiciels.

Contenu

Jour 1

Qualification d'un code premier niveau : OSINT, bac à sable automatique. Environnement de travail : installation d'un environnement d'analyse (isolé / ouvert) pour procéder aux traitements de codes malveillants. Structure PE/ELF : comprendre le format et les aspects utilisés par les codes. Analyse statique et dynamique d'un code : concepts et exemples simples.

Jour 2

Assembleur x86(-64) : premiers pas, contrôle du flot d'exécution et des instructions importantes. Windows/Linux : API, bibliothèques à connaître et utiliser par les codes malveillants. Désassembleur 101 : prise en main, cas des décompilateurs. Debugger 101 : prise en main, étude pas-à-pas & point d'arrêt.

Jour 3

Rétro-ingénierie de logiciels malveillants réels : logiciels malveillants PE et ELF, observation des interactions avec le système d'exploitation. Scripts malveillants : désobfuscation PowerShell et émulation de shellcode. Analyse des techniques de dissimulation : LNK, HTA.

Advanced Malware Analysis

3 jours | Niveau avancé

Description

Dans un contexte de menaces en constante évolution, les auteurs de logiciels malveillants emploient des techniques d'obfuscation de plus en plus sophistiquées pour échapper à la détection. Cette formation pratique s'adresse aux professionnels de la sécurité souhaitant perfectionner leurs compétences en analyse de logiciels malveillants.

Les participants utiliseront des outils tels que Ghidra SRE, le débogueur x96dbg et le framework MIASM pour analyser des logiciels malveillants complexes ciblant Windows et UNIX. L'accent est mis sur les menaces émergentes comme les logiciels malveillants Golang, les payloads multi-étapes, les portes dérobées BPF et les loaders fortement obfusqués. La formation aborde la neutralisation des techniques anti-analyse (contournement des environnements sandbox, anti-débogage), la reconnaissance des schémas d'obfuscation/chiffrement pour la désobfuscation automatisée via des scripts Ghidra, la récupération des symboles Golang à partir d'échantillons obfusqués et l'analyse des portes dérobées BPF.

- 3 jours (21 heures)
- Analyse de code malveillant dans différents langages
- Étude des menaces multi-étapes

Public et prérequis

Cette formation s'adresse aux personnes possédant déjà de solides connaissances en analyse de logiciels malveillants et souhaitant approfondir leurs compétences en rétro-ingénierie de logiciels malveillants complexes, ainsi qu'en extraction d'informations utiles pour la réponse aux incidents et la chasse aux menaces. Elle est principalement destinée aux équipes de réponse aux incidents et aux analystes de menaces.

Connaissances pratiques en analyse de logiciels malveillants et en rétro-ingénierie des formats PE et ELF (x86-64). Connaissances de base du fonctionnement interne de GNU/Linux et Windows.

Contenu

Jour 1

Techniques anti-analyse Windows et Linux : contrôles et contournements mis en œuvre par les logiciels malveillants pour protéger dynamiquement leur code et masquer leur comportement. Stratégies visant à réduire la détection du débogage et l'empreinte mémoire des machines virtuelles.

Jour 2

Aperçu des techniques d'obfuscation courantes : encodage et chiffrement des appels d'API, prédicats opaques, aplatissement du flux de contrôle. Outils et scripts Ghidra pour récupérer les symboles et reconstituer le flux d'exécution. Analyse de logiciels malveillants multi-étapes : logiciels malveillants complexes mettant en œuvre diverses stratégies d'évasion, du programme d'installation à la porte dérobée finale.

Jour 3

Rétro-ingénierie des menaces : analyse de logiciels malveillants Golang et Rust, compréhension de leur structure respective et découverte d'outils d'obfuscation dédiés. Stratégies de récupération des symboles et d'extraction des éléments d'intérêt.
Analyse de portes dérobées BPF : introduction à eBPF, étude de cas d'une porte dérobée BPF avancée.

Ransomware Investigation

3 jours | Niveau intermédiaire

Description

Les attaques informatiques de type ransomware sont les menaces les plus redoutées des entreprises. L'urgence provoquée par la destruction, même partielle, du système d'information peut rapidement déborder vos équipes sécurité si elles ne sont pas préparées. Cette formation permet de mieux anticiper ce type d'incident et ainsi réduire la durée de résolution : réduire le stress, anticiper le séquençage des activités à mener et s'accorder rapidement sur les méthodes à appliquer. Cette formation a pour objectif de donner les clés afin d'investiguer un incident de type ransomware qu'il soit localisé à un petit périmètre ou généralisé.

Durant la formation, les bonnes pratiques pour contrer les attaques ransomwares seront dispensées, en particulier afin de contenir et endiguer l'incident. Véritable course contre-la-montre, les participants à la formation seront ainsi familiarisés aux méthodes et outils à mettre en œuvre. Il est à noter que seuls le volet technique d'investigation et les premières étapes de l'investigation sont abordés ici.

- 3 jours (21 heures)
- Identifier le mode opératoire des principaux groupes de ransomware
- Premiers pas vers la remédiation

Public et prérequis

Cette formation est adaptée pour des personnes ayant déjà été confrontées à des incidents de sécurité, et redoutant les évènements d'ampleur de type ransomware. Des compétences techniques sont requises pour manipuler les TD et ainsi investiguer ce type d'incident. Les outils DFIR ORC et Velociraptor sont utilisés pour illustrer la formation : la connaissance préalable de ces outils est un plus.

- Membre d'équipe de cybersécurité interne dans les entreprises ou l'administration
- Administrateur système avec des compétences en cybersécurité
- Responsable sécurité souhaitant appréhender les aspects techniques

De bonnes connaissances Windows sont recommandées pour comprendre le fonctionnement des attaques (Active Directory, RDP, PowerShell, etc.). Les manipulations sont réalisées sur les environnements Linux.

Contenu

Jour 1

Les signes avant-coureurs d'une attaque. Gestes de premiers secours : préservation des systèmes et des sauvegardes, coupure des accès et réduction de l'emprise de l'attaquant. Quelles traces préserver en premier. Cas d'une attaque par un prestataire. Travailler en environnement compromis : mythe, réalité et pragmatisme. Outils à utiliser pour partager l'information et bonnes pratiques pour mener les investigations.

Présentation de DFIR ORC et Velociraptor.

Jour 2

Approche antéchronologique. Trouver rapidement les premières informations structurantes, course contre-la-montre. Identifier les codes malveillants et portes dérobées. Comme remonter le fil de l'attaque. Cas des postes d'administration / postes de crises. Comment établir et partager une situation (IOC / chronologie / périmètre de compromission). Biais de l'analyste. Trouver l'équilibre entre exhaustivité et efficience.

Jour 3

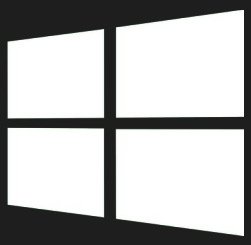
Approche par les modes opératoires. Bénéfices et limites de la CTI. Connaître la surface d'exposition et les opportunités d'exploitation. Mécanismes de persistance. OSINT à la rescousse. La triade : VPN / phishing / service vulnérable. Effets recherchés par l'attaquant. Identifier les fichiers exfiltrés et les mécanismes d'exfiltration. Cas particulier des attaques sous faux drapeau, attaquants multiples sur un réseau.

CyberRemédiation : Bonnes pratiques opérationnelles de gestion de crise et gestion des risques. Reprendre confiance dans le système d'information et nettoyer son parc.

Active Directory: Hardening &

Post-Compromise Recovery

5 jours | Niveau intermédiaire



Description

Pour de nombreuses entreprises, l'Active Directory (AD) constitue le cœur de la gestion des identités et des accès. Son omniprésence au sein des systèmes d'information en fait la cible prioritaire des attaques informatiques complexes et des ransomwares. Il est aujourd'hui indispensable de rendre ces infrastructures plus résilientes, mais aussi de maîtriser les étapes clés post-intrusion pour reprendre confiance en son système après un incident.

Au cours de cette formation de cinq jours, les participants acquerront les compétences nécessaires pour défendre et sécuriser un environnement Active Directory de bout en bout. Le programme détaille la mise en œuvre de mécanismes de sécurité robustes, la protection des chemins de contrôle de l'annuaire et les bonnes pratiques d'administration (modèle de Tiering, LAPS, gMSA) pour réduire drastiquement les opportunités de compromission. Enfin, un module dédié prépare les équipes au pire scénario : la présence d'un attaquant sur le réseau et la méthodologie de reprise de confiance et de nettoyage du cœur du SI.

- 5 jours (35 heures)
- Environnement Active Directory complet simulant une infrastructure d'entreprise
- Approche défensive couvrant l'hygiène continue, le durcissement et la réponse post-incident

Public et prérequis

Cette formation s'adresse aux profils techniques chargés de la conception, du maintien en conditions de sécurité ou de la surveillance des infrastructures Active Directory.

- Administrateurs systèmes et réseaux
- Architectes sécurité
- Membres d'équipes sécurité (Blue Team, SOC, CERT)

Une bonne connaissance de l'administration des environnements Active Directory est nécessaire. Une familiarité avec PowerShell est un plus pour aborder sereinement les exercices pratiques d'audit et de configuration.

Contenu

Jour 1

Architecture de sécurité Windows : modèle de sécurité, gestion des jetons d'accès (Access Tokens), SID. Mécanismes d'authentification : NTLM vs Kerberos en profondeur. Concepts clés : fonctionnement des GPO, délégation. Anatomie d'une compromission : Cyber Kill Chain (scénario ransomware), vol d'identifiants (LSASS dumping, extraction SAM), techniques de déplacements latéraux.

Jour 2

Chemins de contrôle : analyse et sécurisation (BloodHound / SharpHound), nettoyage des permissions (ACL/ACE) sur les objets sensibles. Délégation d'administration : principe du moindre privilège. Services réseaux : sécurisation du DNS, du DHCP et désactivation des protocoles obsolètes (LLMNR / NetBIOS). Modèle de Tiering : concepts (Tier 0, 1, 2), déploiement de silos d'administration, Kerberos Armoring (FAST). Comptes de services : mise en place de gMSA (Group Managed Service Accounts) et protection contre le Kerberoasting.

Jour 3

Postes d'administration : implémentation des PAW (Privileged Access Workstations). Gestion des privilèges : administration JIT (Just-in-Time), JEA (Just-Enough-Administration), déploiement de LAPS (Local Administrator Password Solution). Environnements hybrides : extension vers le cloud (Entra ID), sécurisation du connecteur Microsoft Entra Connect, compréhension des attaques Cloud-to-On-Prem et On-Prem-to-Cloud. Contrôle d'accès : mise en place du Conditional Access et du MFA pour les rôles critiques.

Jour 4

Visibilité et Log Management : configuration avancée de la politique d'audit Windows. Supervision étendue : déploiement et configuration de Sysmon. Centralisation : analyse des logs critiques et identification des Event ID incontournables. Hygiène AD : audit et contrôle de conformité, vérification régulière des droits et privilèges (scripts PowerShell), utilisation d'outils d'analyse de configuration internes.

Jour 5

Scénario catastrophe : méthodologie de Remediation Strategy. Analyse de la persistance : traque des accès maintenus (WMI, tâches planifiées, backdoors). Stratégies de restauration : concept de « Bascule AD » vs « Nettoyage ». Nettoyage AD : procédure de double réinitialisation du mot de passe KRBTGT, éviction de l'attaquant. Durcissement d'urgence : création et application de GPO de survie.

Data Breach: Investigations, Crisis Management & Compliance

1 jour | Niveau junior

Description

Les fuites de données représentent l'un des risques majeurs pour les organisations, avec des conséquences financières, réputationnelles et judiciaires souvent lourdes. Lorsqu'une fuite survient, la rapidité et la justesse de la réponse sont critiques. Il est indispensable de mener de front les actions de remédiation technique et les démarches légales et réglementaires.

Au cours de cette formation d'une journée, les participants apprendront à orchestrer la réponse à une fuite de données. Le programme adopte une approche duale : un volet technique axé sur les "quick wins" et l'investigation rapide selon le type de menace (ransomware, exposition web, malveillance interne), couplé à un volet juridique détaillant les obligations légales, la gestion de la sous-traitance et la communication de crise. L'objectif est de fournir des réflexes opérationnels immédiatement applicables.

- 1 jour (7 heures)
- Approche duale : actions techniques immédiates (quick wins) et conformité juridique
- Étude de scénarios d'investigation concrets (ransomware, vulnérabilités web, mauvaises configurations)
- Focus sur la communication de crise et la gestion des sous-traitants

Public et prérequis

Cette formation s'adresse aux profils techniques et organisationnels impliqués dans la gestion de crise cyber et la protection des données.

- Membres d'équipes de réponse aux incidents (CSIRT / SOC)
- Responsables de la Sécurité des Systèmes d'Information (RSSI)
- Data Protection Officers (DPO) et responsables juridiques
- Directeurs des Systèmes d'Information (DSI)

Une compréhension générale des architectures informatiques et une sensibilisation aux enjeux de protection des données personnelles (ex: RGPD) sont recommandées. Aucun prérequis technique avancé en forensic n'est exigé.

Contenu

Contexte et détection initiale : sources de détection (OSINT, dark web, monitoring interne, signalements), premiers réflexes organisationnels. Qualification de la fuite : évaluation de l'impact, nature des données exposées, volumétrie, aide à l'investigation et au dimensionnement de la crise. Investigations techniques (Quick Wins) : méthodologie de recherche et de confinement par scénarios ; site web (vulnérabilités, injections), attaque par ransomware (double extorsion, exfiltration), malveillance interne (fuite volontaire, compromission de compte), mauvaises configurations (stockage cloud exposé, bases de données ouvertes). Bonnes pratiques : réflexes de préservation des preuves, recommandations de durcissement d'urgence. Communication de crise : stratégie de communication autour de la fuite, éléments de langage, coordination interne vs externe, gestion de la réputation. Obligations légales et notifications : cadre réglementaire, délais légaux de notification aux autorités de contrôle, information des personnes concernées, documentation de l'incident. Écosystème et responsabilités : gestion du cas particulier de la sous-traitance (contrats, chaîne de responsabilité), appréhension du contexte international (transferts transfrontaliers, lois extra-territoriales).

Cloud Forensics in Azure

3 jours | Niveau junior

Description

Microsoft Azure est largement déployé dans de nombreuses entreprises. Les services cloud tels que la messagerie M365, les machines virtuelles ou les bases de données managées sont des cibles privilégiées des attaques. La gestion des autorisations et de l'authentification étant la pierre angulaire de la sécurité de ces infrastructures, elle joue un rôle crucial. La plupart des scénarios rencontrés lors d'incidents exploitent des mécanismes natifs du cloud pour compromettre les locataires.

Face à ces menaces, l'analyse forensique numérique évolue en conséquence. L'analyse des journaux distribués, la corrélation des événements via les API et la compréhension de l'écosystème Azure sont indispensables. Ce cours vise à apporter les connaissances techniques et la méthodologie forensique essentielles pour analyser les incidents de sécurité dans le cloud et y répondre.

- 3 jours (21 heures)
- Concepts, produits et mises en garde concernant Microsoft Azure
- Analyse des attaques les plus courantes et des procédures d'investigation

Public et prérequis

Cette formation s'adresse aux personnes ayant déjà été confrontées à des incidents de sécurité et préoccupées par les risques liés au cloud. Des compétences techniques sont requises pour comprendre les attaques cloud et se connecter à la console CLI (depuis un shell Linux).

Une expérience préalable de PowerShell est un atout.

Contenu

Jour 1

Comprendre l'analyse forensique du cloud : principales différences avec l'analyse forensique traditionnelle. Configurer un environnement d'investigation : outils et scripts essentiels. Qu'est-ce qu'Azure ? Présentation d'Azure et de Microsoft 365. Activation et accès aux journaux. Gestion complexe des licences. EntraID : connexion et rôles. Configuration du locataire. Applications cloud : contournement de l'accès conditionnel. API Microsoft Graph.

Jour 2

M365 : Scénarios de compromission de messagerie professionnelle. Exchange Online : Recherche de comptes compromis. Abonnement Azure : Journaux des ressources. Machine virtuelle, stockage Azure et service géré. Paramètres du locataire. Mappage des journaux.

Jour 3

Exploration KQL : stratégie de centralisation, requêtes et exploration, analyse des journaux. Liste de contrôle des outils d'investigation. Atelier pratique.

Cloud Forensics in AWS

3 jours | Niveau junior

Description

Amazon Web Services (AWS) est la plateforme de prédilection d'innombrables entreprises, ce qui fait de ses services des cibles privilégiées pour les attaquants. Des composants essentiels tels que les instances EC2, les buckets S3 et les bases de données RDS sont constamment menacés. La pierre angulaire de la sécurisation de ces infrastructures est la gestion des identités et des accès (IAM), qui régit l'ensemble des autorisations et de l'authentification. La plupart des incidents survenant sur AWS exploitent des méthodes natives du cloud, fondamentalement différentes de celles utilisées dans les systèmes sur site.

Face à ces menaces, l'analyse forensique numérique doit évoluer. L'analyse des journaux distribués provenant de services tels que CloudTrail et CloudWatch, la corrélation des événements via l'API AWS et une compréhension approfondie de l'écosystème AWS sont indispensables. Ce cours est conçu pour fournir les connaissances techniques essentielles et la méthodologie d'investigation nécessaires pour analyser les incidents de sécurité dans le cloud AWS et y répondre.

- 3 jours (21 heures)
- Concepts, produits et mises en garde d'AWS
- Analyse des attaques les plus courantes et des procédures d'investigation

Public et prérequis

Cette formation s'adresse aux personnes ayant déjà été confrontées à des incidents de sécurité et préoccupées par les risques liés au cloud. Des compétences techniques sont requises pour comprendre les attaques cloud et se connecter à la console CLI (depuis le shell Linux).

Une connaissance préalable des scripts Linux est un atout.

Contenu

Jour 1

Comprendre l'analyse forensique du cloud : principales différences avec l'analyse forensique traditionnelle. Mise en place d'un environnement d'investigation : outils et scripts essentiels. Qu'est-ce qu'AWS ? Présentation des principaux services. Focus sur IAM. Comprendre le système de journalisation CloudTrail/CloudWatch.

Jour 2

Plus d'informations sur les journaux AWS : analyse des instances EC2 compromises, des AMI Amazon, des compartiments S3, des attaques courantes, de GuardDuty, des journaux réseau et des VPC.

Jour 3

Plans d'action IR. Analyse de volumes importants et création de chronologies. Amazon Athena. Optimisation de l'analyse grâce à un outil tiers.

Agentic AI Red Teaming

5 jours | Niveau intermédiaire

Description

Le développement d'architectures agentiques marque une rupture technologique majeure, transformant les LLM passifs en systèmes proactifs capables d'orchestrer des workflows complexes là où l'algorithmique traditionnelle atteint ses limites. Cette évolution permet de concevoir des systèmes autonomes aptes à raisonner, à s'interfacer avec des outils tiers et des bases de données, tout en maintenant un cadre d'exécution maîtrisé et sécurisé.

Cette formation de cinq jours a pour objectif de transmettre les compétences nécessaires à la conception de ces agents de nouvelle génération. L'enseignement est structuré autour d'un fil rouge pratique : le développement complet d'un agent Red Team dédié à la reconnaissance et à l'identification de vulnérabilités. Les participants apprendront à manipuler les modèles à l'état de l'art, qu'ils soient distants (Anthropic, OpenAI) ou locaux (Ollama), et à les intégrer dans des architectures multi-agents robustes.

- 5 jours (35 heures)
- Développement d'un agent de reconnaissance et identification de vulnérabilités
- Utilisation avancée du protocole MCP (Model Context Protocol) et de smolagents
- Environnement de développement Python et accès aux modèles fournis

Public et prérequis

Cette formation s'adresse aux profils techniques souhaitant monter en compétence sur l'ingénierie des systèmes autonomes basés sur l'IA. Elle nécessite une aisance particulière avec les environnements de développement modernes.

- Développeurs logiciels (intermédiaires à expérimentés)
- Pentesteurs et chercheurs en sécurité
- Ingénieurs SecDevOps

Une expérience solide en programmation Python est indispensable. Les participants doivent être familiers avec la ligne de commande Linux et posséder des bases en réseaux (TCP/IP, HTTP). Bien qu'un intérêt pour l'écosystème LLM soit un plus, aucune connaissance préalable en IA n'est requise.

Contenu

Jour 1

Introduction au développement agentique : Concepts clés de l'inférence, état de l'art des modèles (API vs local), dimensionnement hardware et panorama des frameworks. Scripting de requêtes vers des moteurs d'inférence (online/offline) et programmation d'un premier agent simple.

Jour 2

Architecture Agentique et utilisation d'outils : Boucle "Thought-Action-Observation" et conditions d'arrêt. Appels de fonctions et typage. Architecture et spécialisation des agents. Développement d'outils python personnalisés (tool calling). Mise en place d'une architecture multi-agents avec orchestrateur, worker, analyzer. Mise en place de CodeAgent capable d'exécuter du code de façon cloisonnée.

Jour 3

Protocole MCP (Model Context Protocol) : Introduction à MCP. Composants : hôtes, clients et serveurs. Le protocole, ses couches de transport (JSON-RPC), fonctionnalités (outils, rersources, prompts), messages réseaux et sécurité (authentification, isolation). Présentation de python-mcp et fastMCP. Lister programmatiquement les outils, ressources et prompts exposés par un serveur MCP. Création d'un serveur MCP exposant un outil de scan réseau. Création d'un serveur MCP exposant des ressources dynamiques (logs). Intégration d'un client MCP et utilisation de MCP par l'agent.

Jour 4

Mémoire, RAG et injection de contexte : Notion de fenêtre de contexte, limites et optimisations (mémoire courte). RAG: Génération Augmentée par Récupération, théorie et librairies. Mémoire persistante via structures de données classiques. Implémentation d'une base de données (CVE) avec vectorDB. Ajout d'un agent "Sumarizer" pour optimiser la fenêtre de contexte. Implémentation d'un outil "remember" avec stockage persistant.

Jour 5

Qualification, observabilité, mise en production et sécurité : Focus sur les nombreux risques de sécurité liés au déploiement d'agents et bonnes pratiques (sandboxing, proxying, caching, hardening). Présentation des outils de monitoring et d'introspection. Présentation de stratégie de tests pour valider les workflows malgré le non-déterminisme inhérent lié aux LLMs. Implémentation de scénarios de tests fonctionnels et d'évaluation des performances. Isolation des outils et renforcement de sécurité. Finalisation de l'agent.

Network Interception in Rust

3 jours | Niveau intermédiaire

Description

La maîtrise de l'interception et de la manipulation des flux réseau est une compétence critique pour l'analyse de sécurité, le test d'intrusion et le développement d'outils offensifs. Cette formation propose une immersion technique consistant à bâtir, à partir de zéro, un outil de type Man-In-The-Middle (MITM) performant en langage Rust.

L'approche est intégralement basée sur la pratique : les participants conçoivent un intercepteur capable de réaliser des attaques d'empoisonnement ARP et de traiter des flux applicatifs complexes. À travers des ateliers progressifs, la formation explore les couches fondamentales du modèle OSI, le décodage de trames et la mise en œuvre de mécanismes d'interception transparents pour les protocoles HTTP et TLS. L'utilisation de Rust garantit ici une gestion mémoire sûre et des performances optimales pour le traitement de données à bas niveau.

- 3 jours (21 heures)
- Sécurité offensive et développement Rust
- Développement d'un outil de MITM complet et fonctionnel
- Compréhension fine des mécanismes réseaux exploités

Public et prérequis

Cette formation s'adresse aux profils techniques souhaitant allier développement système moderne et expertise réseau offensive.

- Développeurs Rust souhaitant appliquer leurs compétences à la sécurité bas-niveau
- Pentesteurs / auditeurs désirant concevoir leurs propres outils d'interception
- Ingénieurs sécurité spécialisés dans l'analyse de protocoles

Une maîtrise des bases du langage Rust (ownership, structures, gestion des dépendances, des erreurs) est indispensable. Des connaissances fondamentales en réseaux (modèle OSI, encapsulation, TCP/IP) sont nécessaires pour suivre les modules techniques.

Contenu

Jour 1

Couches réseau et LAN : présentation des couches manipulées dans le cadre d'une interception MITM, analyse du fonctionnement d'un réseau local, résolution ARP et comportement des machines dans un LAN. Développement bas niveau : développement en Rust des premières briques d'interception transparente sur les couches Ethernet, IP et TCP

Jour 2

HTTP et HTTPS : présentation du fonctionnement détaillé des échanges (structure, en-têtes et mécanismes de session), implémentation de l'interception HTTP (reconstruction des flux, inspection et modification du contenu). Interception TLS : introduction aux mécanismes TLS (handshake, certificats, contraintes et limites), développement de l'interception TLS. Validation : test en conditions réelles sur un environnement physique simulant un réseau local, avec tests complets de bout en bout.

Jour 3

Protocoles avancés : série de tests approfondis sur les couches IP et TCP (gestion des options, MSS, fragmentation, retransmissions), développement et analyse d'une seconde attaque de réseau local (DHCP Spoofing). Performances et contraintes : utilisation d'outils de benchmarking en Rust (analyse des performances, mesure des temps de traitement, optimisation), réflexion sur les contraintes réelles (stabilité, robustesse, scalabilité, détection par les systèmes de défense). Finalisation : consolidation du code produit et packaging final de l'outil MITM.

Advanced Rust

5 jours | Niveau avancé

Description

Le langage Rust est aujourd'hui incontournable, offrant aux développeurs la productivité d'un langage moderne de haut niveau combinée à des garanties strictes de robustesse et à un puissant système de types. Cette formation a pour but d'aborder les concepts avancés auxquels sont confrontés les développeurs Rust, afin d'acquérir la capacité de concevoir des logiciels complexes, fiables et hautement maintenables.

Les sujets traités, issus de retours d'expérience concrets, mettent un fort accent sur la pratique. Tout au long de la semaine, les participants développeront un outil complet d'analyse et d'extraction de données multi-thread. Ce projet sera itérativement amélioré pour intégrer des critères de recherche dynamiques, des performances concurrentes, une interface réseau client/serveur asynchrone, des tests de bout en bout et un pilotage via une interface web.

- 5 jours (35 heures)
- Projet de scanner de fichiers amélioré itérativement à chaque TP
- Implémentation de concepts complexes : multi-threading, asynchronisme et sérialisation avancée

Public et prérequis

Cette formation s'adresse à des profils techniques souhaitant perfectionner leurs compétences de développement Rust.

- Développeurs logiciels
- Ingénieurs système et embarqué
- Auditeurs de code et chercheurs en sécurité

Une connaissance préalable de Rust est indispensable : les participants doivent être capables de lire du code Rust et être familiers avec sa syntaxe ainsi qu'avec ses concepts fondamentaux (le pattern matching, le borrowing, etc.).

Une expérience de développement système ou embarqué, et/ou des langages C/C++ est un plus.

Contenu

Jour 1

Gestion des erreurs : approches pour gérer les opérations faillibles, via la bibliothèque standard Rust ou via des crates communes, avantages et limitations de chaque approche, éléments de choix et maintenabilité. Panic : spécificités, causes explicites ou implicites, approches idiomatiques pour structurer le code et réduire leurs occurrences. Utilisation optimale des types, API et idiomes : patterns communs (new type, builder), avantages et spécificités du typage Rust pour contraindre l'état ou l'interface d'un programme, API fonctionnelles (combinateurs) des itérateurs / Option / Result, maintenabilité, performance du code généré, antipatterns et erreurs fréquentes.

Jour 2

Performance & multi-thread : primitives et architectures de traitement multi-thread, API associées dans la bibliothèque standard et dans des crates communément utilisées, outils pour évaluer la performance d'un programme (benchmarking et profiling), et identifier des axes d'amélioration. Les concepts sont appliqués au projet via l'optimisation du scanner de fichiers, le passage de l'architecture en multi-thread et l'intégration de compteurs statistiques.

Jour 3

Serde : exemples concrets d'utilisation de Serde, configuration et modélisation pour en tirer le meilleur bénéfice. async / await : présentation du modèle async, ses avantages, inconvénients, spécificités, et pièges communs. Mise en pratique immédiate par l'implémentation d'un protocole commande/réponse réseau s'appuyant sur Serde et le développement d'interfaces client/serveur asynchrones.

Jour 4

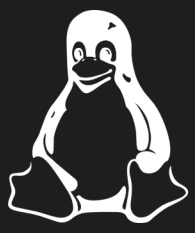
Tests : tests automatiques, structuration dans un projet complexe, niveaux de tests possibles (unitaire, intégration, performance), conseils pour faciliter leur maintenance et efficacité, outillage avancé (mesure de la couverture, snapshot testing). Mise en œuvre approfondie de tests d'intégration complets validant de bout en bout les interactions entre le client et le serveur du scanner.

Jour 5

Développement Web : concepts et intégration du développement web en Rust, appliqués au pilotage des scans via la création d'une interface dédiée. Autres problématiques (selon temps et besoins des participants) : compilation conditionnelle et code build time (macros simples, macros procédurales, build.rs), gestion, maintenance et audit d'un arbre de dépendances, utilisation de FFI pour interagir avec du code C, utilisation avancée du linter Clippy, les parseurs par combinateurs et leurs avantages par rapport aux parseurs par grammaire (mis en pratique via le filtrage par type MIME).

Linux Hardening

4 jours | Niveau intermédiaire



Description

Les systèmes Linux constituent la base de la majorité des infrastructures et des postes de travail. Toutefois, une configuration par défaut ne permet généralement pas de limiter l'impact en cas de compromission d'un service ni d'entraver les mouvements latéraux. La sécurisation de ces environnements repose sur la maîtrise des outils de restriction et d'isolation natifs de l'OS.

Au cours de cette formation de quatre jours, les participants aborderont quatre modules techniques dédiés au durcissement des systèmes Linux : le contrôle d'accès obligatoire (AppArmor), le filtrage réseau (nftables), le confinement des processus (systemd) et la conteneurisation rootless (podman). Ces concepts seront systématiquement mis en pratique à travers des laboratoires dédiés, allant de l'écriture de profils stricts à l'isolation manuelle de processus en s'appuyant sur les primitives du noyau.

- 4 jours (28 heures)
- 4 modules techniques dédiés aux mécanismes de restriction et d'isolation sous Linux
- Machines virtuelles Linux individuelles
- Approche fortement axée sur la pratique (écriture de profils AppArmor, configuration nftables, durcissement systemd, conteneurisation manuelle et via podman)

Public et prérequis

Cette formation s'adresse aux profils techniques souhaitant concevoir, administrer ou auditer des systèmes Linux durcis.

- Administrateurs systèmes Linux
- Développeurs d'applications sécurisées
- Ingénieurs sécurité / DevSecOps

Une maîtrise de l'administration système Linux (ligne de commande, gestion des processus, système de fichiers) ainsi que des bases en réseaux (TCP/IP) et en pare-feu (iptables/nftables) sont nécessaires pour suivre les différents modules.

Contenu

Jour 1 : AppArmor

Mécanismes de sandboxing : panorama des solutions sous Linux, comparaison SELinux et AppArmor. Fondamentaux AppArmor : compréhension d'un profil, cheatsheet des commandes utiles, pièges classiques. Création et modification : modification de profils existants, écriture de profils depuis zéro. Syntaxe avancée : contrôle fin des transitions (clean exec, stacking de profils), contrôle des accès réseau, limites de ressources. Lancements multiples : exécution de deux instances d'un programme avec des profils distincts. Labs : écriture de profils, réalisation d'un jailed shell (invite de commande en lecture seule sans accès réseau).

Jour 2 : Nftables

Netfilter : le système de hooks, comparaison iptables vs nftables. Syntaxe Nftables : structure des règles, gestion conntrack et NAT. Filtrage avancé : filtrage des flux sortants par UID/GID. Exploitation : compteurs, logging et monitoring pour le débogage de règles. Labs : mise en place d'un pare-feu nftables à l'état de l'art sur une VM Linux, implémentation du filtrage des flux sortants.

Jour 3 : Systemd

Présentation : architecture et projet systemd. Gestion de l'init : remplacer initrc, services systemd. Hardening : durcissement des units systemd (restriction système de fichiers, réseau, capabilities). Fonctionnalités avancées : socket activation. Remplacements systèmes : remplacer cron (timers systemd), ifupdown (systemd-networkd), grub (systemd-boot), resolvconf (systemd-resolved), rsyslog (systemd-journald). Labs : audit et durcissement d'units systemd.

Jour 4 : Podman

Concepts d'isolation : présentation des outils de base des conteneurs modernes. Isolation manuelle (labs) : isoler un processus du système de fichiers (chroot), isoler du réseau (unshare), contrôler les accès réseau. Conteneurisation moderne : différences fondamentales entre docker et podman. Utilisation de podman (labs) : compiler du code exotique (cross-compilation) sans polluer le système hôte, lancer une application graphique non approuvée de manière isolée.

SYNACKTIV