Activité WireShark
Un analyseur de réseau indispensable

L'objectif de cette activité est de comprendre l'importance d'utiliser des logiciels sécurisés.

1 – WireShark, c'est quoi ?

WireShark est un logiciel libre et open source pour l'analyse et le dépannage des réseaux informatiques. Il permet aux utilisateurs de capturer et d'inspecter les données qui circulent sur un réseau informatique en temps réel. Les filtres d'affichage et de capture permettent de se concentrer sur des paquets spécifiques en fonction de critères précis :

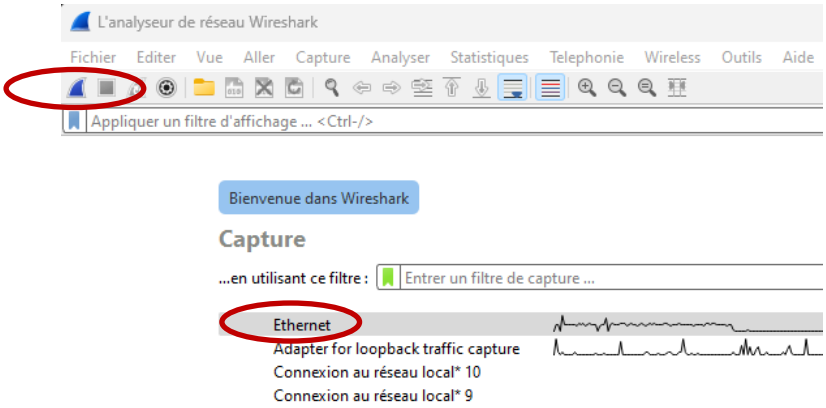
- Adresse IP
- Protocole
- Port
- Etc

Lors de cette activité vous allez voir que l'utilisation d'un logiciel mal sécurisé peut permettre la récupération d'un mot de passe.

2 – Analyse avec WireShark

☞ 1 – Démarrer le logiciel WireShark. Vous devez sélectionner la carte Ethernet que vous allez analyser. Dans mon cas : "Ethernet".

Pour démarrer la capture, cliquez sur l'aileron de requin et sur le carré rouge pour la stoppée.



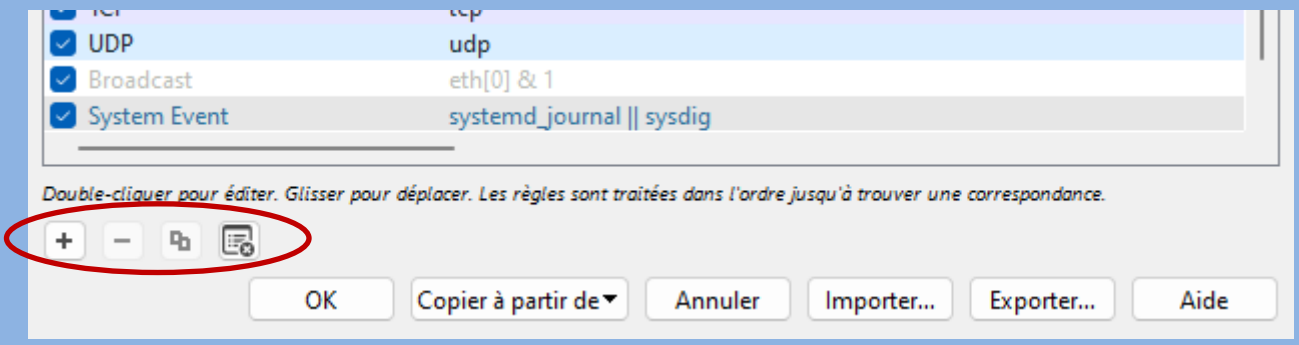
☞ 2 – WireShark utilise un code couleur pour classer les paquets en fonction du protocole, de l'adresse IP, de la fonction, ...

Nom	Filtre
Bad TCP	tcp.analysis.flags && !tcp.analysis.window_update && !tcp.analysis
HSRP State Change	hsrp.state != 8 && hsrp.state != 16
Spanning Tree Topology Change	stp.type == 0x80
OSPF State Change	ospf.msg != 1
ICMP errors	icmp.type in { 3..5, 11 } || icmpv6.type in { 1..4 }
ARP	arp
ICMP	icmp || icmpv6
TCP RST	tcp.flags.reset eq 1
SCTP ABORT	sctp.chunk_type eq ABORT
IPv4 TTL low or unexpected	(ip.dst != 224.0.0.0/4 && ip.ttl < 5 && !(pim || ospf || eigrp || bgp || t
IPv6 hop limit low or unexpected	(ipv6.dst != ff00::/8 && ipv6.hlim < 5 && !(ospf|| bgp || tcp.port==
Checksum Errors	eth.fcs.status=="Bad" || ip.checksum.status=="Bad" || tcp.checksum
SMB	smb || nbss || nbns || netbios
HTTP	http || tcp.port == 80 || http2
DCERPC	dcerpc
Routing	hsrp || eigrp || ospf || bgp || cdp || vrrp || carp || gvrp || igmp || ismp
TCP SYN/FIN	tcp.flags & 0x02 || tcp.flags.fin == 1
TCP	tcp
UDP	udp
Broadcast	eth[0] & 1
System Event	systemd_journal || sysdig

✍ 3 – Effectuez une capture de 2 secondes environ et vérifiez le code couleur. Suivant le résultat de votre capture, relevez trois couleurs et le protocole associé.

Vert clair --> paquet http
Bleu clair --> Paquet UDP
Violet clair --> Paquet TCP

Vous pouvez également créer votre propre règle de couleur en fonction de critères tels que l'adresse IP source ou de destination, le port, le protocole, la taille du paquet, etc.



Dans l'ensemble, les règles de couleur dans Wireshark sont un outil puissant pour aider les utilisateurs à analyser le trafic réseau et à identifier rapidement les paquets importants ou suspects.

Pour la suite du TP vous utiliserez la capture "capture1.pcapng".

✍ 4 – Ouvrir le fichier "capture1.pcapng" puis relevez la durée de la capture et le nombre de trames. Calculez le nombre de trames par seconde.

Durée de la capture : 180.7 secondes
Nombre de trames : 4903
Nombre de trames /seconde = 27.1 trames par seconde

✍ 5 – Utilisez la combinaison suivante : Ctrl + Shift + Alt + C. Qu'obtenez-vous ?

On obtient le détail du fichier de capture : date, nom, système d'exploitation, nom de la carte réseau, durée, nombre de trames, …

☞ 6 – Retrouvez la valeur du nombre de trames par seconde (pps) et vérifiez votre calcul de la question 4.

☞ 7 – Lorsque vous faites l'analyse d'une capture, vous pouvez ajouter des commentaires en lien avec certaines trames. Par exemple, sélectionnez la trame 5, clic droit puis "commentaire sur les paquets" et "ajouter un nouveau commentaire". Ecrivez : "l'adresse IP 192.168.1.25 est celle du poste de DUPONT".

☞ 8 – Utilisez à nouveau la combinaison : Ctrl + Shift + Alt + C. Vérifier la présence du commentaire dans les propriétés du fichier de capture.
Ceci est utile lorsque plusieurs personnes sont amenées à travailler sur le même fichier ou pour mettre en avant une analyse, une méthodologie, …

✍ 9 – Quels sont les protocoles utilisés lors de cette capture ?

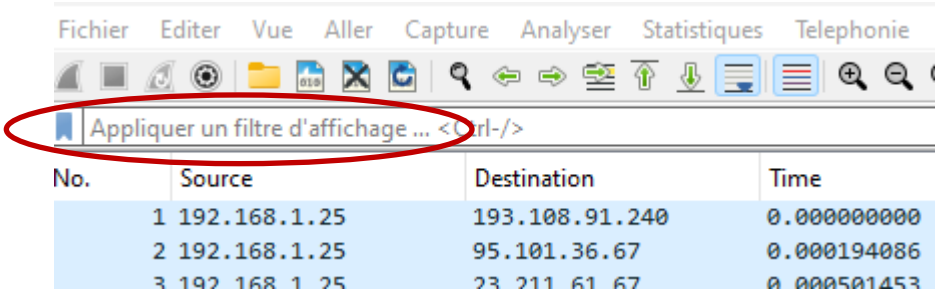
On trouve les protocoles suivants : DNS, TCP, http, TLSv1.2, OCSP, IPv4, TLSv1.3

✍ 10 – Complétez le tableau suivant :

Nom du protocole	La fonction de ce protocole
DNS	DNS (Domain Name System) est un système utilisé pour traduire les noms de domaine en adresses IP. Il permet aux utilisateurs d'accéder à des sites web et à d'autres ressources en ligne en utilisant des noms de domaine faciles à retenir, plutôt que des adresses IP numériques difficiles à mémoriser.
TCP	Le protocole TCP (Transmission Control Protocol) est un protocole de communication utilisé pour transmettre des données sur Internet. Il est responsable de la transmission fiable de données entre deux hôtes connectés à Internet, en garantissant que les données soient transmises dans le bon ordre et sans erreurs.
http	Le protocole HTTP (Hypertext Transfer Protocol) est un protocole de communication utilisé pour transférer des données sur le web. Il est utilisé pour permettre la communication entre un client (généralement un navigateur web) et un serveur web, afin de charger des pages web et d'autres ressources en ligne.
TLSv1.2	Le protocole TLS (Transport Layer Security) version 1.2 est un protocole de sécurité utilisé pour sécuriser les communications entre un client et un serveur. Il est conçu pour protéger la confidentialité et l'intégrité des données échangées entre deux parties, en utilisant des techniques de chiffrement et d'authentification. Lorsqu'un client se connecte à un serveur en utilisant le protocole TLS, le serveur présente un certificat numérique qui permet au client de vérifier son identité. Le client et le serveur négocient ensuite un algorithme de chiffrement symétrique pour protéger les données échangées pendant la session. Les clés de chiffrement sont générées de manière dynamique pour chaque session, ce qui garantit que les données ne peuvent être interceptées ou modifiées par des tiers.
TLSv1.3	Ajout de nouveaux algorithmes plus sécurisés, tels que Cha20 et AES-256-GCM. Suppression de certaines fonctionnalités vulnérables.
IPv4	IPv4 (Internet Protocol version 4) est un protocole de communication utilisé pour transmettre des données sur Internet. Il est responsable de l'adressage et du routage des paquets de données sur le réseau Internet.
OSCP	OCSP (Online Certificate Status Protocol) est un protocole de sécurité utilisé pour vérifier l'état d'un certificat numérique. Il permet de déterminer si un certificat est toujours valide ou s'il a été révoqué, c'est-à-dire s'il a été annulé avant sa date d'expiration normale. Le protocole OCSP fonctionne en permettant à un client de demander l'état d'un certificat à un serveur OCSP. Le serveur OCSP vérifie l'état du certificat auprès de l'autorité de certification (CA) qui a émis le certificat, puis renvoie une réponse au client indiquant si le certificat est valide ou révoqué.

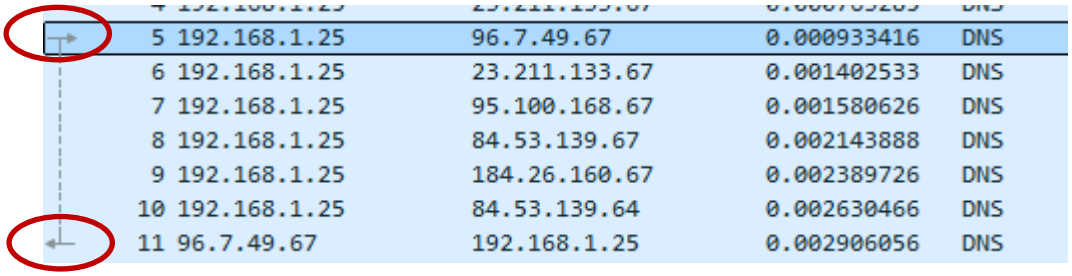
3 – Analyse du protocole DNS

☞ 11 – Vous allez filtrer les trames. Si l'on souhaite filtrer le protocole DNS il suffit d'écrire "dns" dans la zone de filtrage de WireShark. Faites le test.



Vous consterez que seule les trames qui utilisent le protocole DNS sont affichées. Cela fait encore beaucoup.

☞ 12 – Cliquez sur la trame 5. On constate que WireShark vous indique (flèche dans la marge gauche) que la réponse se situe à la trame 11.



✍ 13 – Que pouvez-vous en conclure en analysant ces deux trames ?

C'est la réponse du service DNS à la requête (trame 5) : quelle est l'adresse IP du nom de domaine "a22-67.akam.net"

⇨ Adresse IP : 23.211.61.67

✍ 14 – Quelle est l'adresse IP du serveur DNS ?

La trame 5 est envoyée depuis l'adresse 192.168.1.25 à l'adresse 96.7.49.67. C'est avec cette dernière adresse que la réponse est envoyée vers la source 192.168.1.25.
Donc 96.7.49.67 est le serveur qui héberge le service DNS.

✍ 15 – Effectuez une nouvelle recherche, sélectionnez seulement les trames qui utilisent l'adresses ip du serveur DNS.
Commencez à taper "ip" dans la barre de recherche. WireShark vous propose différentes syntaxes. A vous de jouer !

Commande : ip.addr == 96.7.49.67

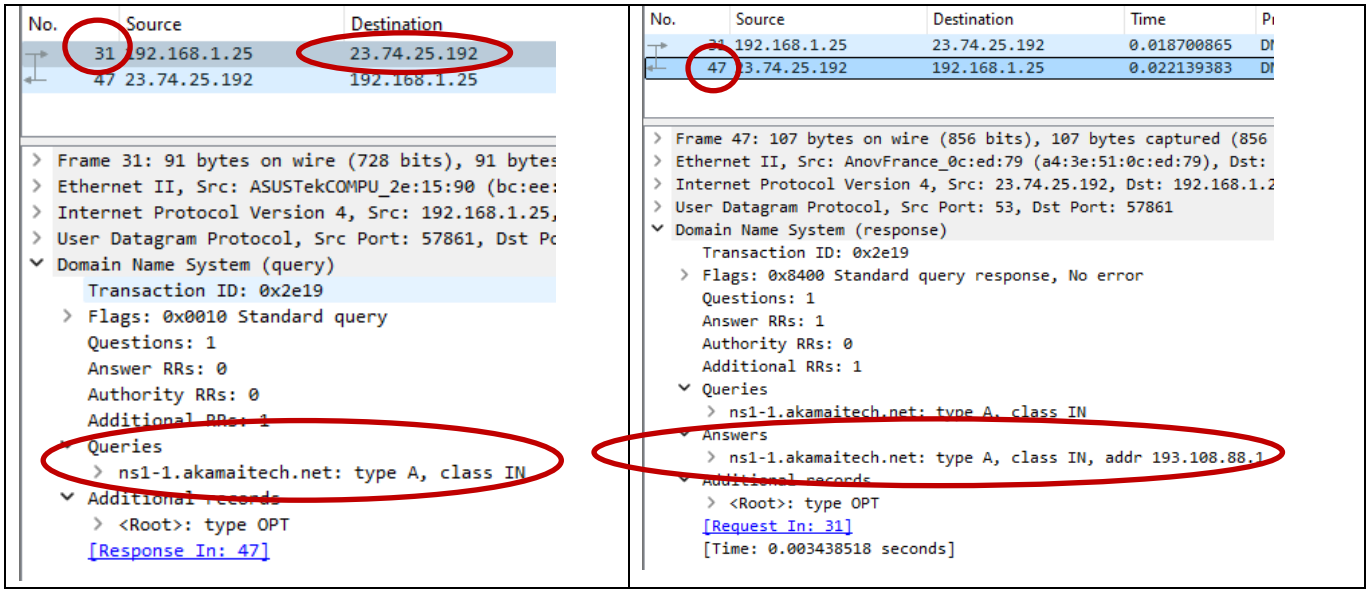
✍ 16 – Quelles sont les trames que vous avez trouvées ?

Trames : 5, 11, 20 et 22.

✍ 17 – Sélectionner la trame 31. Quelle est l'adresse du serveur DNS utilisé et retrouvez l'adresse IP du site demandé ?

Serveur DNS : 23.74.25.192 Site demandé : 193.108.88.1

Ces données sont aussi accessibles dans les données des trames 31 et 47.



✍ 18 – Pouvez vous en déduire le numéro du port utilisé par le service DNS ?

Port : 53

✍ 19 – Trame 113, nouvelle demande DNS, analysez la réponse et donnez l'adresse IP du site recherché.

Demande DNS : URL : a1089.dscd.akamai.net

Adresse IP : 23.200.86.74

✍ 20 – A partir de quelle trame, ce site est-il consulté ?

C'est à partir de la trame 115 que l'hôte 192.168.1.25 initialise une connexion TCP 3 états.

☞ 21 – Pour suivre une requête et consulter toutes les trames en lien avec celle sélectionnée, faire un clic droit puis suivre, puis sélectionné un des protocoles proposés. L'ensemble des trames concernées s'affiche.

✍ 22 – Pourquoi a-t-on soit du TCP, soit du HTTP comme protocole ?

L'hôte 192.168.1.25 va envoyer une requête HTTP pour télécharger le fichier success.txt. Or le protocole HTTP utilise le protocole TCP pour transporter les données. Donc il faut dans un premier temps initialiser une connexion TCP trois états. C'est le rôle des trames 115, 116 et 117.

✍ 23 – Expliquez l'échange au niveau des trames 115, 116 et 117 ?

Les trames 115, 116 et 117 initialisent une connexion TCP trois états. SYN pour demander une synchronisation avec le destinataire, ACK pour valider la requête et un nouveau SYN pour la synchronisation avec la source. Validation de la source (ACK).

On remarquera l'ouverture d'une fenêtre de 30336 octets (à partir de la trame 121)

✍ 24 – Que se passe-t-il à la trame 118 ? Expliquez la suite.

La connexion TCP est ouverte, on envoie le requête http pour télécharger le fichier success.txt.

✍ 25 - Expliquez l'échange entre les trames 121 et 135.

Le téléchargement du fichier se fait en plusieurs trames.

✍ 26 – Que se passe-t-il ensuite ?

On télécharge une nouvelle fois le fichier.

3 – Analyse du protocole OCSP

Présentation du protocole OSCP

Le protocole OCSP (**O**nline **C**ertificate **S**tatus **P**rotocol) est un protocole de sécurité utilisé pour vérifier l'état d'un certificat numérique. Il permet de déterminer si un certificat est toujours valide ou s'il a été révoqué, c'est-à-dire s'il a été annulé avant sa date d'expiration normale.

Le protocole OCSP fonctionne en permettant à un client de demander l'état d'un certificat à un serveur OCSP. Le serveur OCSP vérifie l'état du certificat auprès de l'autorité de certification (CA) qui a émis le certificat, puis renvoie une réponse au client indiquant si le certificat est **valide** ou **révoqué**.

Le protocole OCSP est utilisé dans de nombreuses applications en ligne, telles que la navigation web sécurisée (HTTPS), la messagerie électronique sécurisée (SMTPS, IMAPS, POP3S), le transfert de fichiers sécurisé (FTPS, SFTP) et les connexions VPN sécurisées.

☞ 27 – Isolez les trames OCSP.

Vous pouvez constater qu'il existe une trentaine de lignes, soit des requêtes, soit des réponses. Il faut utiliser une autre méthode pour isoler une requête et sa réponse.

☞ 28 – Sélectionnez la première trame OCSP (trame 221) et sélectionnez l'option suivre.

✍ 29 – On constate que les échanges ont commencé à la trame 218. Pourquoi ?

Trame 218, 219, 220 : On communique en TCP donc il faut au préalable établir la connexion (SYN – ACK).

Trame 221 : Lorsque la connexion est établie, on exécute la requête OCSP.

Trame 224 : retour (ACK) du serveur pour dire qu'il a bien reçu la requête.

Trame 233 : réponse de la requête OCSP.

Trame 223 : retour (ACK) du client pour dire qu'il a bien reçu la réponse.

✍ 30 – Analysez en détail la requête OCSP (trame 221)

URL du site contacté : http://ocsp.pki.goog/GTSGIAG3

Type de hashage : SHA-1

✍ 31 – Analysez en détail la réponse (trame 233)

Status de la réponse : successfull