

eCTHP V2

Intro To Network Hunting

BY: Ahmad Abdelnasser Soliman

abdelnassersoliman0@gmail.com



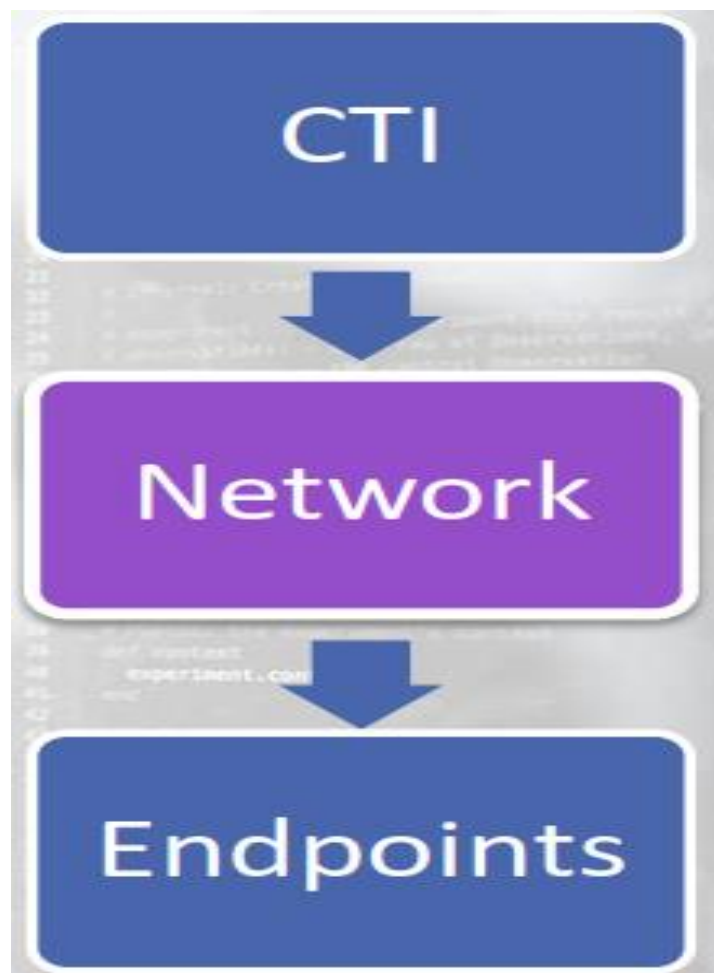
Index Of Content:

2.1	Introduction.....	1-4
2.2	TCP/IP & Networking Primer.....	5-20
2.3	Packet Analysis & Tools.....	20-30

2.1 Introduction:

- فالأول اتكلمنا عن ال **Threat Hunting** بالمصطلحات بتعته وكان
من ضمنها ال **Threat Intelligence** وازاي نستخدمه فال
Hunting .

- طب افرض مجبش معانا نتيجه ساعتها هنروح لل **Network** بتعتنا ندخل نعملها **Hunting** بنفسنا ... طب لو منفعش ساعتها ممكن نروح لل **Endpoint** ننفذ عليها ال **Hunting** بنفسنا ودا لو تلاحظ ترتيب الكورس بتعنا من ال **Threat Intelligence** لل **Network** **Hunting** لل **End point Hunting** ... زي كدا تماما .



- دي بالضبط الطريقه اللى بنشتغل بيها فالكورس بتعنا تدريجي كدا من ال **Cyber Threat Intelligence** لل **Network** لل **End points** ... فأحنا ك **Threat Hunters** بنبدء شغلنا فال **Hunt** أول حاجه بال **CTI** ومبنتمدش عليها بشكل كلي فشغلنا لاء هحتاج ندخل بنفسنا لل **Network** وتبحث عن ال **Threats** اللى انت متوقعها ودا عن طريق **Tools** معينه هنعرفها بعدين ... وبعد ذلك هنروح لل **End point** اللى هي زي ال **PCs** وندخل جواها ونعمل ال **Hunting** .

- احنا ك **Hunters** بناخد ال **Alerts** من ال **Network Team** اللى عندنا فالمؤسسه أو من ال **L1** فال **Soc** اللى هو ال **Incident responder** وبنبدء نعمل ال **Investigation** بتعنا فال **Alerts** اللى بتجلنا دي ... فال **Threat Hunter** مش هيشغل من دماغه لاء دا بيبدء شغل لما يجيله **Request** ...

من ال **Teams** الى شغالين معاه فالمؤسسه وبالذات فحته ال **Network** فحاله حدوث حاله غير اعتياديه زي ان ال **Traffic** بتاع ال **Network** فجاء بقا على عن المعتاد ساعتها ال **Network Team** بيبدء يشوف حاجه مشبوهه أو غير معتاده عندنا فال **Network** فيبعت ال **Alert** لل **Threat Hunter** عشان يبدء شغله بال **Hunting** عال **Network** بأنه عمل **Capture** لل **Packets** وبعد كدا هيقوم عاملها **Analyze** لل **Packet** الى علمها ال **Capture** عشان يتأكد اذا كان فيه **Active Threat** موجوده عندنا فال **Network** وللااء.

- ال **Network Team** الى معانا فالمؤسسه دايمًا بيراقبوا ال **Appliance** الموجوده عندنا فال **Network** الى هي الأجهزة الموجوده عندنا زي ال **IDS** وال **IPS** وال **FW** وال **Router** وال **Switch** وال **VPN** وال **Proxy** وغيرهم ... كل الأجهزة دي هتلاقيها بتبعتك ف **Alerts** وانت تعملها **Analyze** و **Investigation** كمان عشان تشوف هل دا بيشكل علينا **Threat** وللااء ... وكمان انت ممكن تتابع ال **Network** وتشوف الى بيحصل عليها من خلال ال **Statistical Flow Analysis** ودا عن طريقه بتقدر تراقب ال **Traffic** نفسه وانت عارف ال **Normal** بتاع المؤسسه الخاص بال **Traffic** عامل ازاى فانت اى حاجه هتلاحظها غير ال **Normal** زي ال **Traffic** بقا على مره واحده مع ان مفيش حاجه اتغيرت عن الشغل المعتاد لل **Network Team** فهنا تبدء تشك وهتبدء الأجهزة دي تظهر انها مستهلكه أو فيه ضغط عليها فال **Statistical Flow analysis** فانت تبعت علطول ال **Alert** لل **Threat Hunter** عشان يتعامل معاه ... الكلام دا هتلاقي الشركات مطبقاه بالفعل من منتجات شركه زي **Solar Wind** المتخصصه فال **Devices** الخاصه بمراقبه ال **Network** وحمايتها ... وكمان ال **Network Team** من خلال تحديد ال **Network Base line** بيقدروا يحددوا من خلالها ال **Normal Use** وأي حاجه مادون ذلك بتتصنف **Suspicious** .

- فمثلا انتوا ك **Network Team** حاطين ال **Baseline** يعني ثابت
لل **Network Team** عشان يحددوا ال **Normal Use** عندكوا فال
Network ليوم الحد دا اليوم بيبقا فيه ضغط شغل وال **Traffic**
عندكوا بيبقا على وكمال ال **Download** فأحنا راقبنا دا وعارفين انه
بيستهلك **4 GIGA** مثلا دا بالنسبالنا هو ال **Base line** اللي هنشغل
عليه ... جينا فيوم ما لقينا ال **Download** بقا **20 GIGA** وال
Traffic على برضه هنا بقا تبدي تشك فلي بيحصل عندك عال
Network زي ال **Unusual Spikes** اللي هي القفزات اللي بتهر
فال **Statistical Flow Analysis** الخاص بال **Network** فدا
ممك يكون **Exfiltration** لل **Data** يعني تسريب لل **Data**
الموجوده فالمؤسسه عندك من خلال ال **Network** عشان كدا ال
Traffic بقا على مره واحده وتبعت **Alert** لل **Threat Hunter**
عشان يبدء يعمل **Investigation** ويشوف ايه ال **Event** اللي حصل
عندنا عال **Network** خلى ال **Download** وال **Traffic** يبقوا
عالين بالشكل دا... فأحنا بنبص عال **Traffic** اللي عندنا ونقارنه بال
Baseline بتعنا وعلى أساس كدا بنقرر هل دا **Normal** ولا لاء
وعلى أساس كدا بنبتدي ناخذ ال **Action** ... وطبعا احنا ك **Threat**
Hunters ملناش دعوه بشغل ال **Network team** وملناش دعوه
بالتعامل مع ال **Tools** الخاصه بال **Statistical Flow Analysis**
وكمال ال **Network Baselines** وال **Alerting** اللي بتحصل عال
Network دا شغل ال **Network Team** اللي عندنا فالمؤسسه ...
انما احنا ك **Threat Hunters** لازم نعرف ال **Tools** اللي بتخلينا
ناخذ **Packet Capture** لل **Traffic** عشان نعمله **Analyze** زي ال
Wire Shark كدا ونعرف كمال ازاى نعمل عن طريقها
Investigation لل **Packet** دي عشان نشوف هل فيه **traffic**
Suspicious ولالاء ... ومش شرط ال **Alerts** تيجي لل **Threat**
Hunter من ال **Network Team** فقط لاء ممك تجيله عن طريق
ال **Security Team** اللي هما ال **SOC L1** اللي هما ال **IR** لما
يعملوا **Capture** لل **Abnormal Traffic** عن طريق ال **SIEM** .

2.2 TCP/IP & Networking Primer:

- تعالى هنا نراجع على شويه **Basics** فال **Network** هحتاجها قدام واحنا بن **Hunt** فال **Network** ودا هنشوفه فال **Module** القادم بعنوان ال **Suspicious Traffic Hunting** ... فدا بمثابة تمهيد .

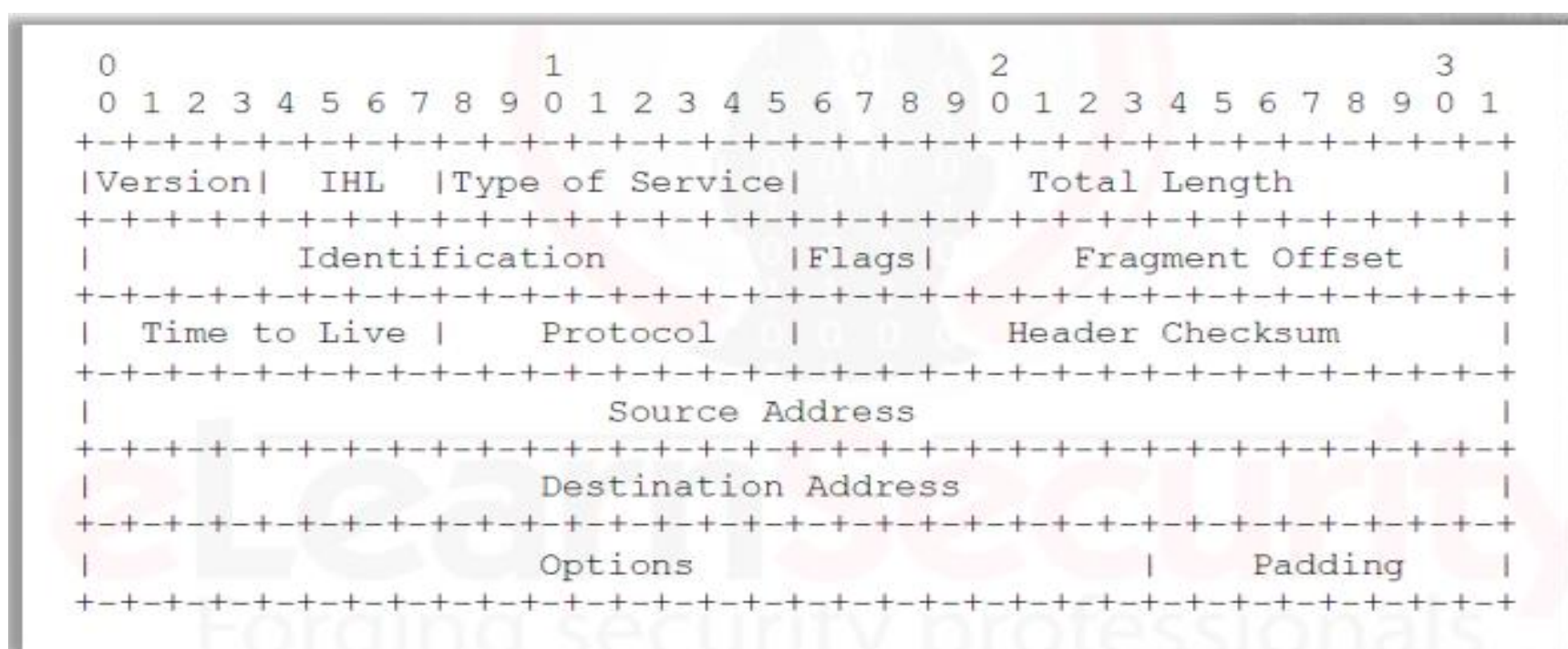
- مهم جدا نعرف ازاي ال **Hosts** بيتم التواصل مابينهم ودا عن طريق فهمنا لل **TCP/IP** ودا هيساعدنا نفهم ازاي ال **Network Communication** بيشتغل وعلى هذا الأساس هنقدر نعمل **Hunt** لأي حاجة مشبوهة أو غير طبيعيه عندنا فال **Network** ... فاحنا هنشوف ال **Normal** من ال **Network Traffic** عامل ازاي بحيث شوفنا أي حاجة مختلفه عن كدا نعرف نصنفها **Suspicious** ونعمل **Hunt** .

- على سبيل المثال انت محدد للأجهزة ال **Protocols** اللى تشتغل بيها وتستخدمها ولقيت عندك **PC** معين شغال ببرتوكول زي ال **FTP** فتيجي هنا تشك فالجهاز دا !! اولاً ال **FTP** لازم عشان يتفعل عندك وتستخدمه لازم تكون واخد **Permission** من ال **Administrator** فمبدعياً دا كدا تصرف **Abnormal** من ال **Machine** دي فتروح تشوف ال **IP** بتعها وتعمل ال **Investigation** هتلاقيه مثلاً مش مسموحه انه يستخدم ال **FTP** أصلاً فهنا الشك أصبح يقين ودا كدا بيعمل تصرف **Suspicious** عندك فال **Network** ودا انت هتعرفه من خلال ال **Monitoring** اللى بتعمله لل **Network** وعن طريق فهمك لل **Network** شغاله ازاي هتقدر تحدد ال **Normal Behavior** من ال **Abnormal** ... فلأزم انت ك **Threat Hunter** تكون فاهم وعارف ال **TCP/IP Model** شغال ازاي وتعرف تتعامل معاه .

- كمان تكون عارف اي هي ال **Port Number** وكل **APP** عندك بيستخدم انهو **Port Number** عشان اي **Port** متساب **Open** بشكل عشوائي أو مش معروف الغرض من ال **Open** دا ايه ساعتها انت تشك فيه وتعمل **Investigation** قد يكون حصل عليك **Attack** مثلا وال **Port** دا اللي فاتحه عندك ال **Malware** عشان يرجع يعمل منه ال **Reverse Connection** وهكذا فمعرفتك لل **Ports** وتعرف كمان شغاله ليه واللي مفتوح عندك دا مستخدم فأيه دي هتساعدك انك تكتشف ال **Ports** اللي بتستخدمها بشكل **Suspicious** ... كمان لازم تكون عارف ال **Normal Behavior** بتاع ال **APP** شغال ازاي وبتاع ال **Protocols** اللي بتستخدمها ال **APP** دا شغالا ازاي وايه الوضع ال **Normal** ليها عشان أي حاجة غير ال **Normal** هنصنفها **Suspicious** ونبدء نعمل ال **Investigation** بتعنا .

- احنا عارفين ان عشان جهازين يتواصلوا مع بعض من خلال ال **Network** فالرساله اللي بيبعتوها لبعض دي اسمها **Packet** وال **Packet** دي بتتقسم لأجزاء عشان تمر من خلال ال **Network Devices** المختلفه لحد متوصل من ال **Source** لل **Destination** والعكس ... وال **Packet** دي بتتكون من **Header** و **Payload** ... وال **Header** دي بتحتوي على ال **Addresses** زي مين ال **Source IP** ومين ال **Destination IP** ومين ال **Source Port** وكذلك ال **Destination Port** فأي **Address** عندك بيبقا موجود فال **Header** بتاعت ال **Packet** أما ال **Content** بتاع الرساله نفسها اللي بيتبعت مابين الجهازين بتلاقيه موجود فال **Payload** ... واللي بيكون ال **Packet** دي بمحتواها هو ال **Source** اللي بيبدء التواصل مع ال **Destination** اللي بيستلم ال **Packet** منه ويفك محتواها ويشوف ال **Source** عاوز منه ايه ويبدء يرد عليه ... وال **Protocols** هي المسؤوله انها تاخد ال **Packet** وتنقلها من ال **Source** لل **Destination** والعكس صحيح .

- تعالى ناخذ مثال ... عشان جهازين يتواصلوا مع بعض لازم قولنا ال **Protocols** اللى هتنقل ال **Packet** تكون موجوده وهنا هنتكلم عن ال **IP Protocol** ودا متخصص فال **Transfer** مابين ال **Devices** فلان لازم نضيف ال **Header** الخاصه بال **Protocol** بتعنا اللى هينقل ال **Packet** جوا ال **Packet** نفسها ... يعني ال **IP Protocol** دا ال **Header** بتعته طولها **20 Bytes** هيتخطوا فجزء ال **Header** فال **Packet** بتعتك ... زي الصوره دي .

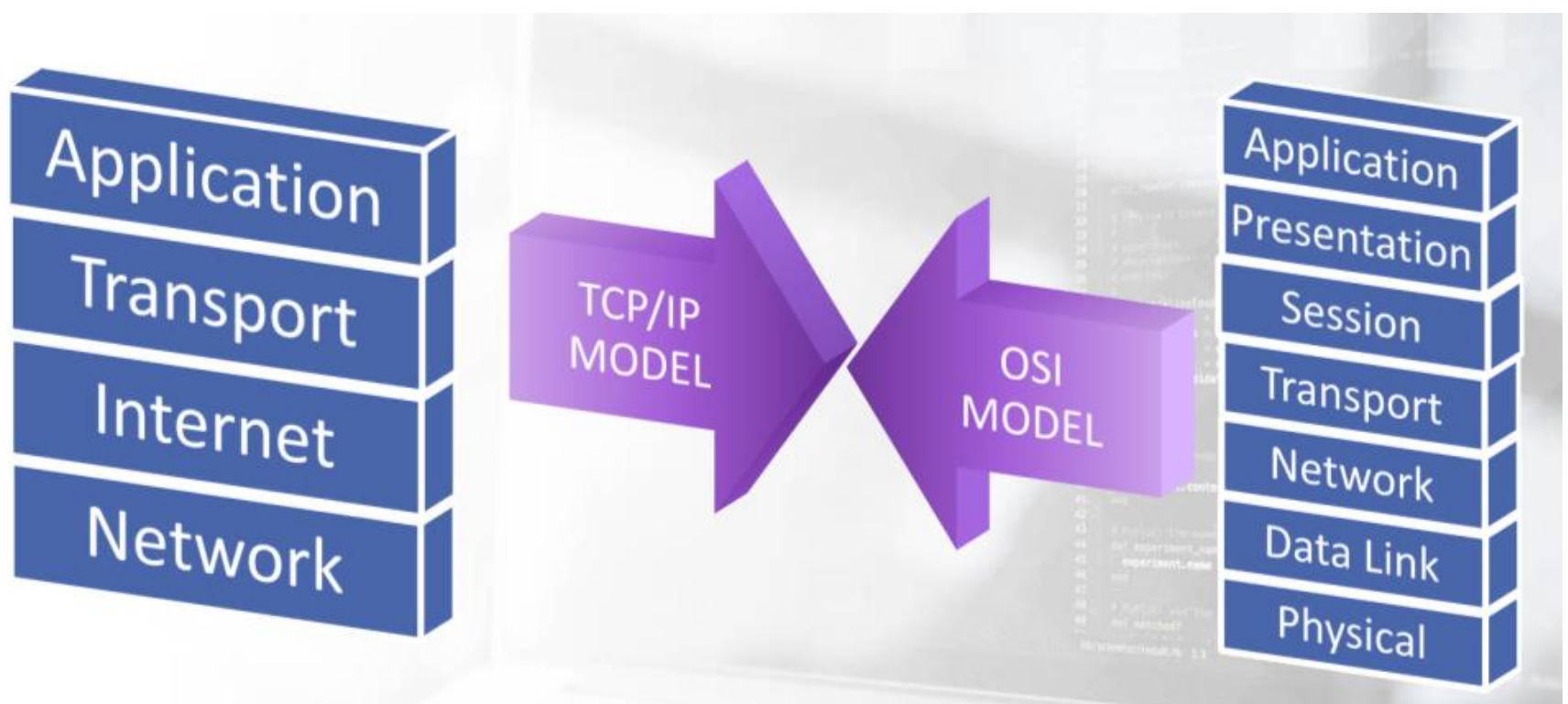


- طبعا دي مكونات ال **Header** بتاعت ال **Packet** واحنا ك **Threat Hunters** مش مطالبين بمعرفه ال **Details** دي خالص دا شغل ال **Network Engineers** الحاصلين على شهادات **Cisco** زي **CCNA** و **CCNP** ... اللى يهمننا واحنا بنعمل **Hunting** عاوزين نبص على حاجه زي ال **Source IP** اللى هنلاقيه موجود فال **Header** زي المثال اللى قدامك بأسم **Source Address** فساعتها احنا شاكين ف **IP** معين ان يكون بيعمل شيء **Suspicious** عندنا فال **Network** فاحنا هنروح لل **Header** نشوف ال **IP** اللى عاوزينه أما باقي ال **Details** متهمناش ... أما ال **Payload** هي ال **Actual** **Information** اللى بتتبعث فال **Packet** عندنا زي ال **Emails** والملفات وغيرهم اللى بنقول عليهم محتوى ال **Payload** ... وفهمك لل **Packet** ومحتواها انت ك **Threat Hunter** هتساعدك قدام وانت بتعمل **Analyze** لل **Large Packets** زي مهنشوف قدام بال **Wire** **Shark** لما نعمل **Capture** ل **Packets** حجمها كبير ...

هتلاقي ان احنا عاوزين حاجات معينه من ال **Packet** دي هنبص على حاجات **Specific** فيها ودا اللي هتوفرهولنا ال **Wire shark** من خلال بعض ال **Features** الموجوده فيها اللي هتساعدنا فكدا .

- تعالى نشوف ال **OSI Model** ونعرف الفرق بينها وبين ال **TCP/IP Model** ؟

- ال **OSI Model** دي طريقه الحوار بين الأجهزة اللي متصله بال **Network (Internet)** فلازم عشان جهازين يتواصلوا مع بعض وكل واحد موجود ف **Network** مختلفه عن الثاني يتواصلوا مع بعض من خلال ال **OSI Model** اللي بتحتوي على **7 layers** وكل **Layer** ليها ال **Protocols** الخاصه بيها ... وبعد ذلك جه بعده ال **TCP/IP Model** وهو نفس قصه ال **OSI Model** ولكنه أسهل وأحدث فال **Communication** مبين الأجهزة فال **Network** وكمال بقا **4 Layers** بدل مكان **7 Layers** فال **OSI Model** ... ودا الفرق بينهم.

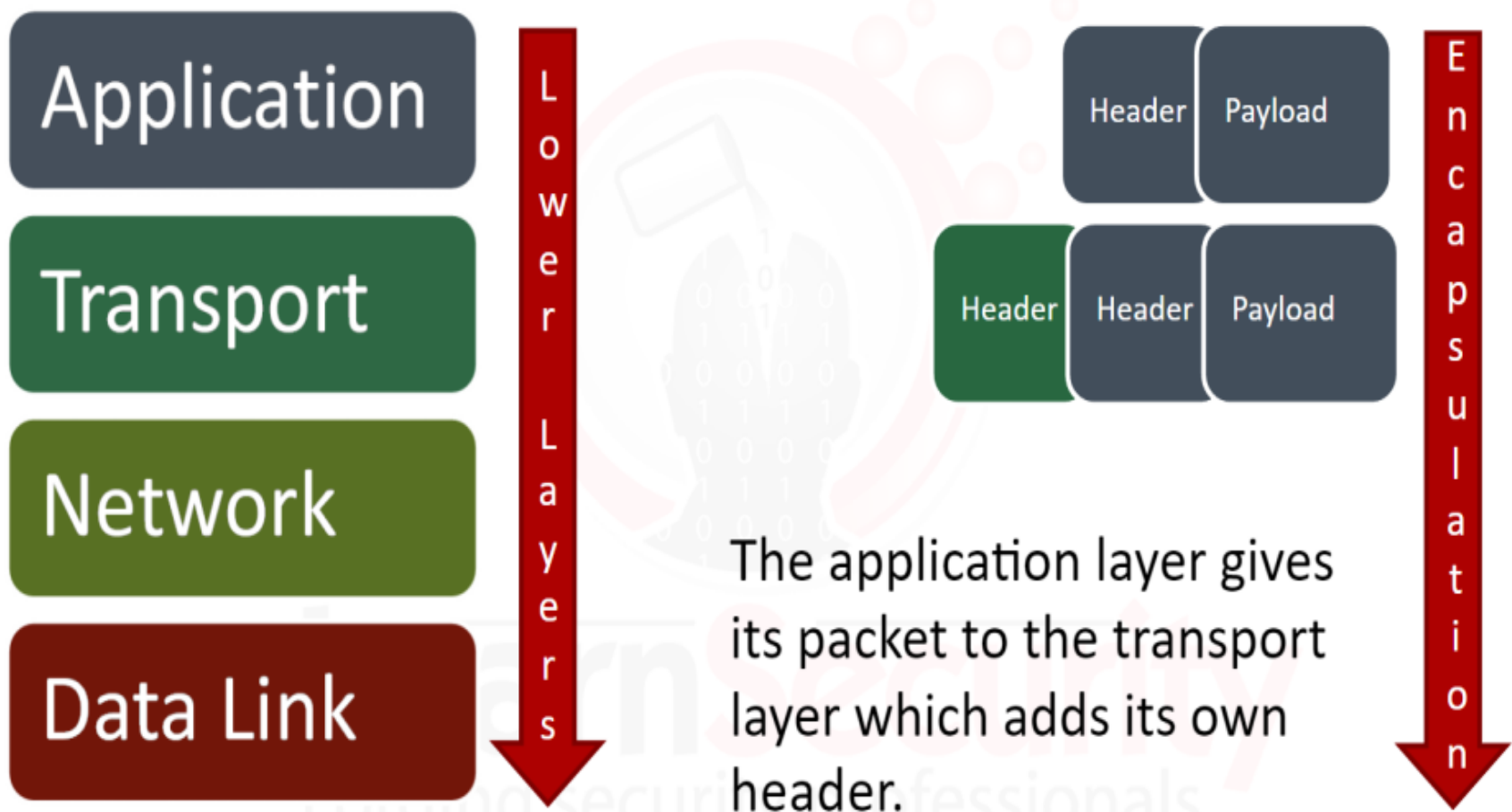


- احنا ك **Threat Hunters** محتاجين نعرف من هنا ال **Different Protocols** عشان كل **Protocol** بيشتغل ف **Different Layer** ... واحنا من هنا ورايح هنعتمد فشغلنا ال **TCP/IP Model** فلازم نعرف كل **Protocol** شغال فكل **Layer** من ال **4 Layer** بتوعها وكمال وظيفته ايه .

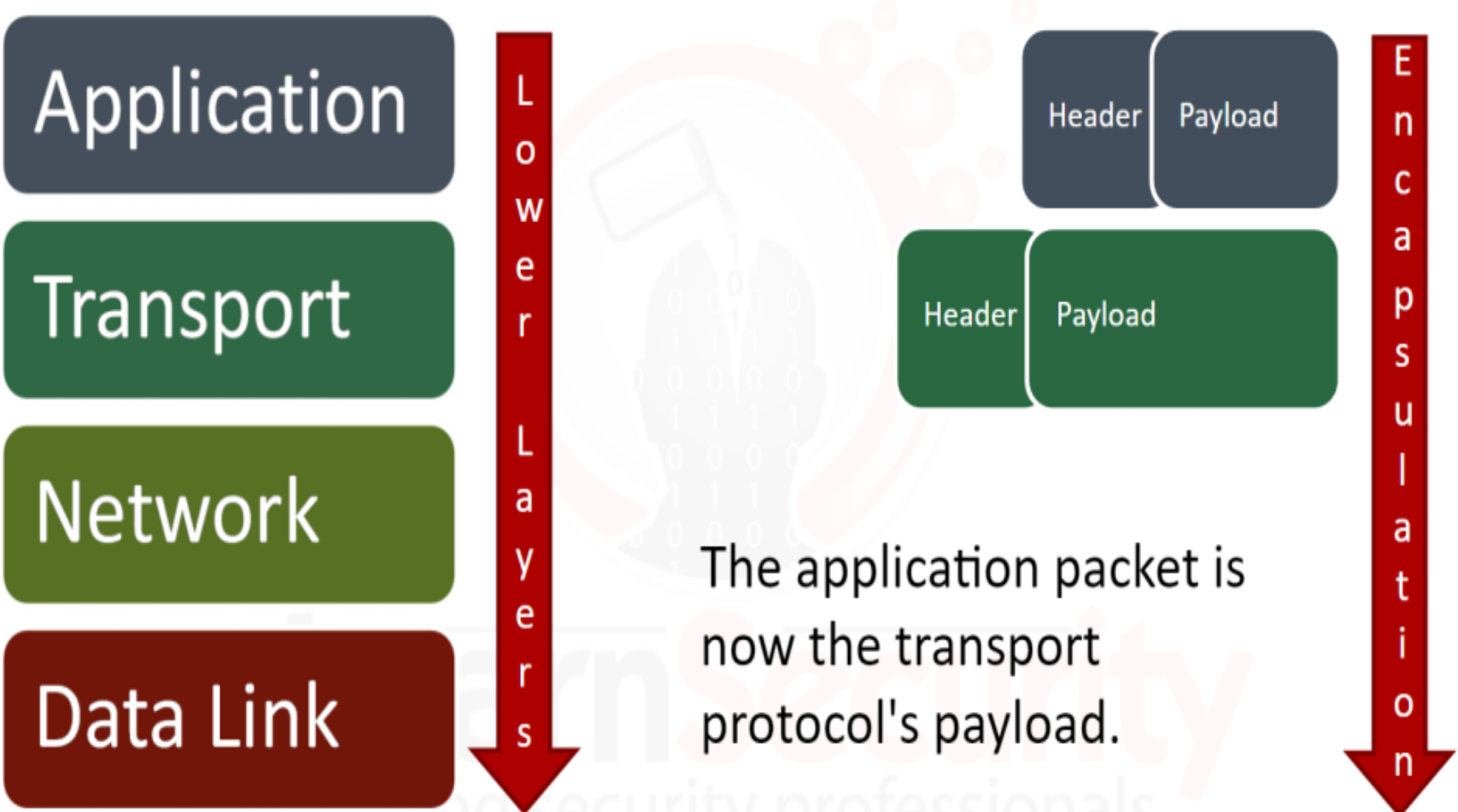
- فعلى سبيل المثال عندك ال **Application Layer** فيها كذا **Protocol** مهمين زي ال **FTP** وال **SMTP** وال **DNS** وال **SNMP** وغيرهم كثير ... ال **Layer** اللى بعدها وهي ال **Transport Layer** ودي برضه فيها **Protocols** زي **TCP** وال **UDP** ... وال **Layer** اللى بعدها وهي ال **Internet Layer** اللى موجود فيها **Protocols** زي ال **IP** وال **ARP** وال **ICMP** ... وال **Layer** الأخيره عندنا هي ال **Network Layer** ودي فيها ال **Ethernet** وال **Token Ring** وال **Frame relay**.

- عشان نكون ال **Packet** بتعتنا وتطلع تروح من ال **Source** لل **Destination** لازم تعدي على كل **Layer** من دول وتستخدم **Specific Protocol** عشان تتم عمليه ال **Encapsulation** اللى هي التغليف لازم ال **Packet** تعدي على ال **Layers** بتاعت ال **TCP/IP** وتستخدم ال **Protocol** المناسب للعمليه اللى هتم وليكن طبقة ال **Transfer** عندنا مندوبين توصيل بيوصلوا ال **Packet** فيه منهم البطيء اللى هو **TCP** ولكن بيضمنك ان ال **Packet** توصل وميحصلهاش **Lose** فالطريق وعندك ال **UDP** دا بيوصلك ال **Packet** بسرعه ولكن ممكن يحصلها **Lose** عادي فبيتم اختيار ال **Protocol** المناسب طبقا لل **Process** اللى هتم ودا بيحصل فكل **Layer** لحد متوصل ال **Packet** لل **Destination** وهناك بيפקها وبيعمل فك للتغليف اللى عمله ال **Source** ويشوف ال **Content** اللى جوا ال **Packet** ... فال **Packet** بتعتنا مش عباره عن **Header** و **Payload** واحده فقط لاء هيحصل لل **Packet** دي ال **Encapsulation** اللى قولنا عليه فكل متعدي على **Layer** من ال **TCP/IP Layer** هتلاقيها بيتعملها تغليف ب **Header** و **Payload** بتاعت الطبقة اللى عدت عليها ... وليكن عدت على ال **Transport Layer** هتلاقيها بتتغلف بال **Header** وال **Payload** بتوع طبقة ال **Transport** وهكذا لحد متوصل لآخر **Layer**.

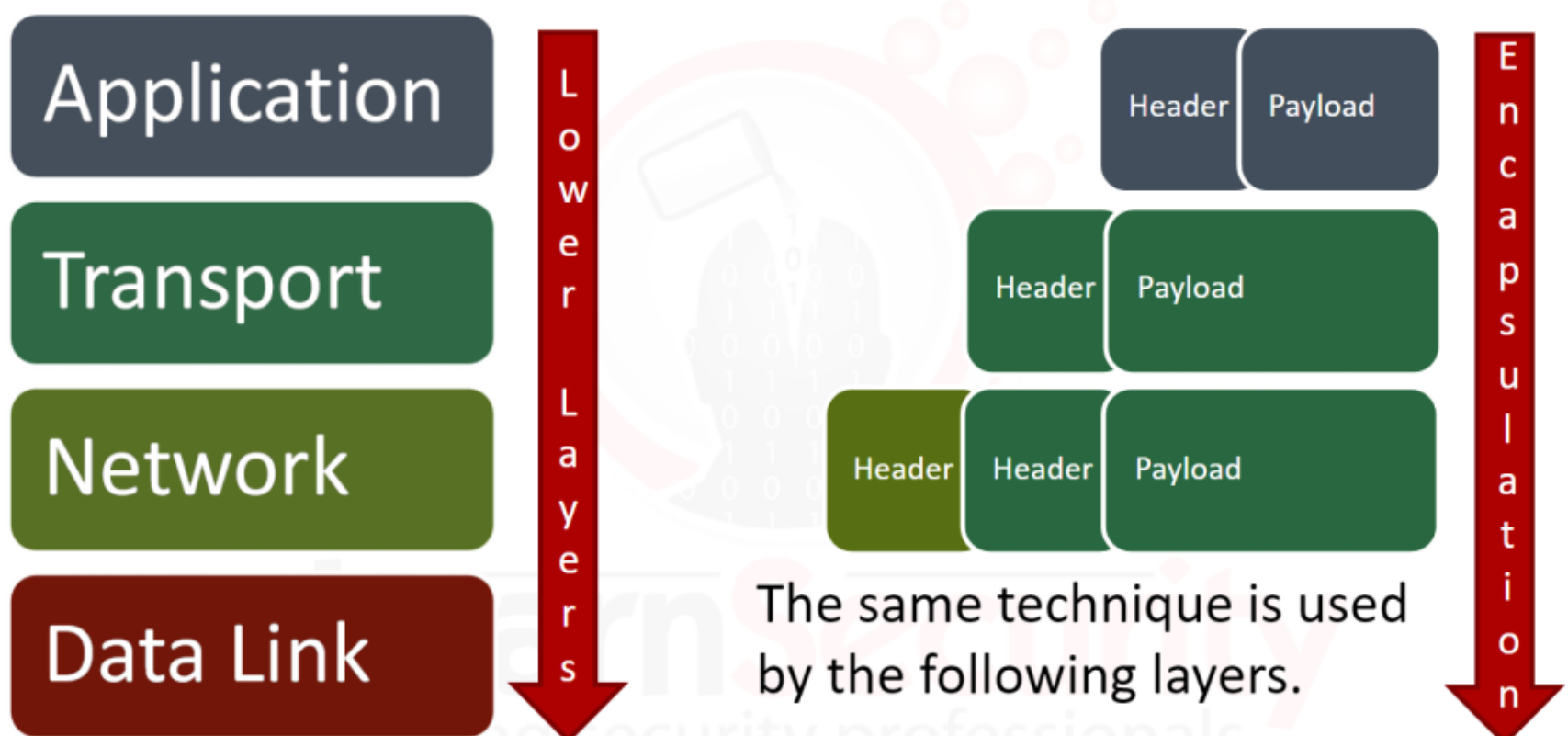
- تعالى نشوف تطبيق الكلام دا عشان الصورة توصلك .



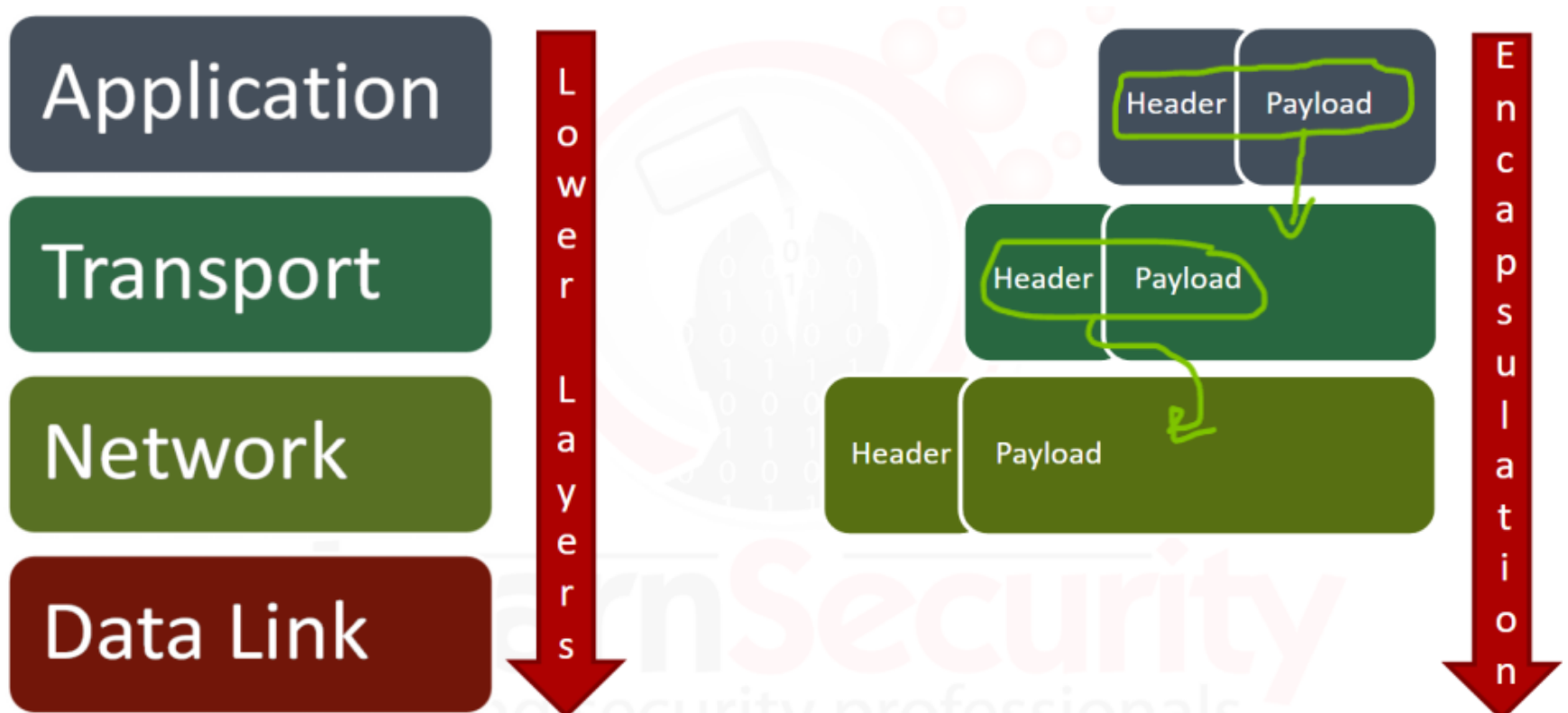
- هنا فال **layer** بتاعت ال **Application** هتلاقينا بنكون ال **Packet** بتعتنا اللى مكونه من **Header** و **Payload** زي مقولنا ... لما نيجي ننزل لل **Layer** اللى بعدها هتلاقي اننا نزلنا بال **Packet** اللى كانت فال **Application Layer** وضمفنا عليهم ال **Header** بتاعت ال **Layer** اللى بعدها وهي ال **Transport** ... وأصبح ال **Header** وال **Payload** بتوع ال **Application Layer** هتخطوا فجزة ال **Payload** بتاع ال **Transport Layer** ... زي كدا .



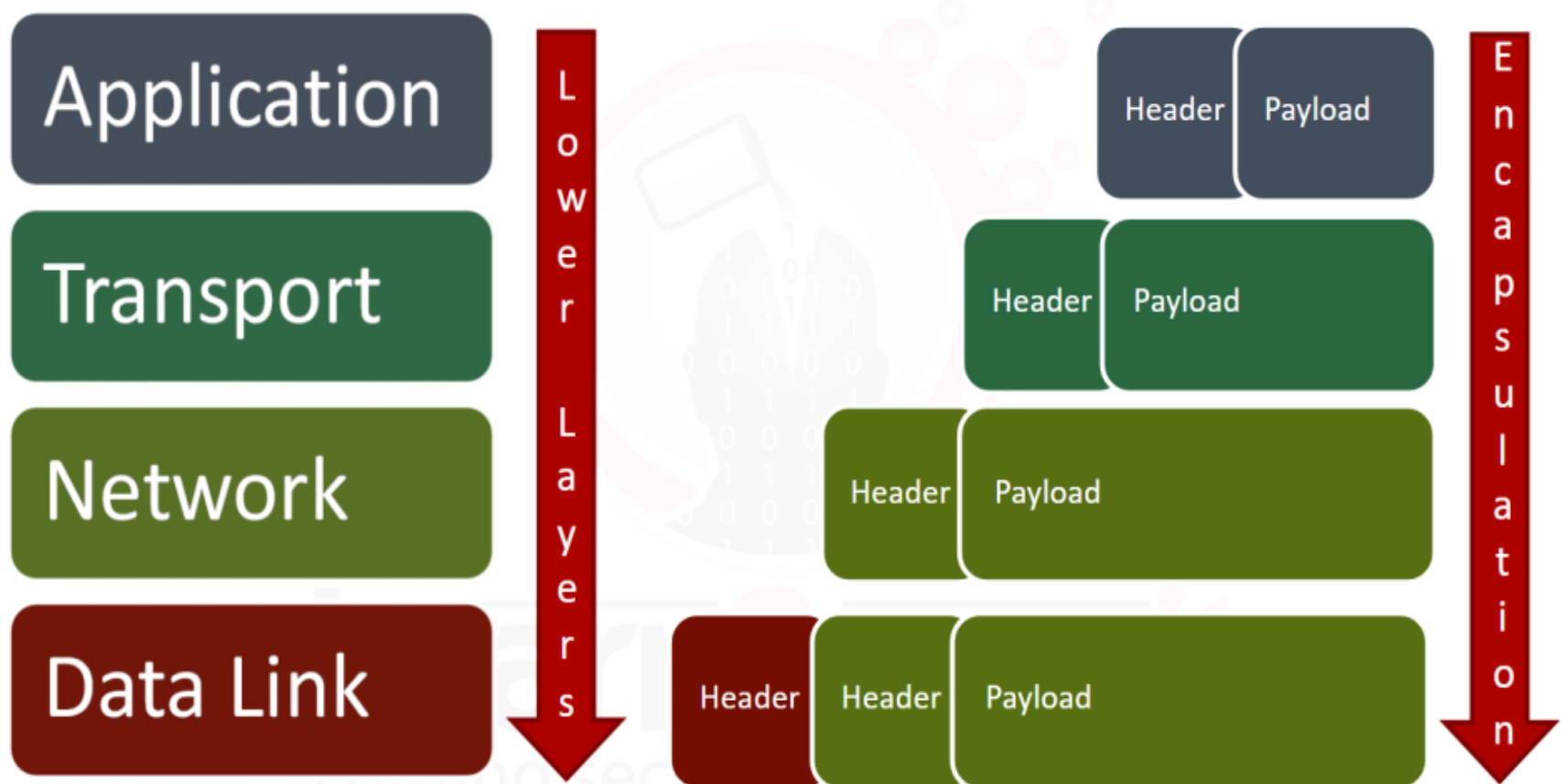
- يبقى الجزء بتاع ال **Header** وال **Payload** بتوع ال **Application Layer** اتخطوا فجزء ال **Payload** بتاع ال **Transport Layer** زي ممتوضح قدامك ... تعالى نكمل .



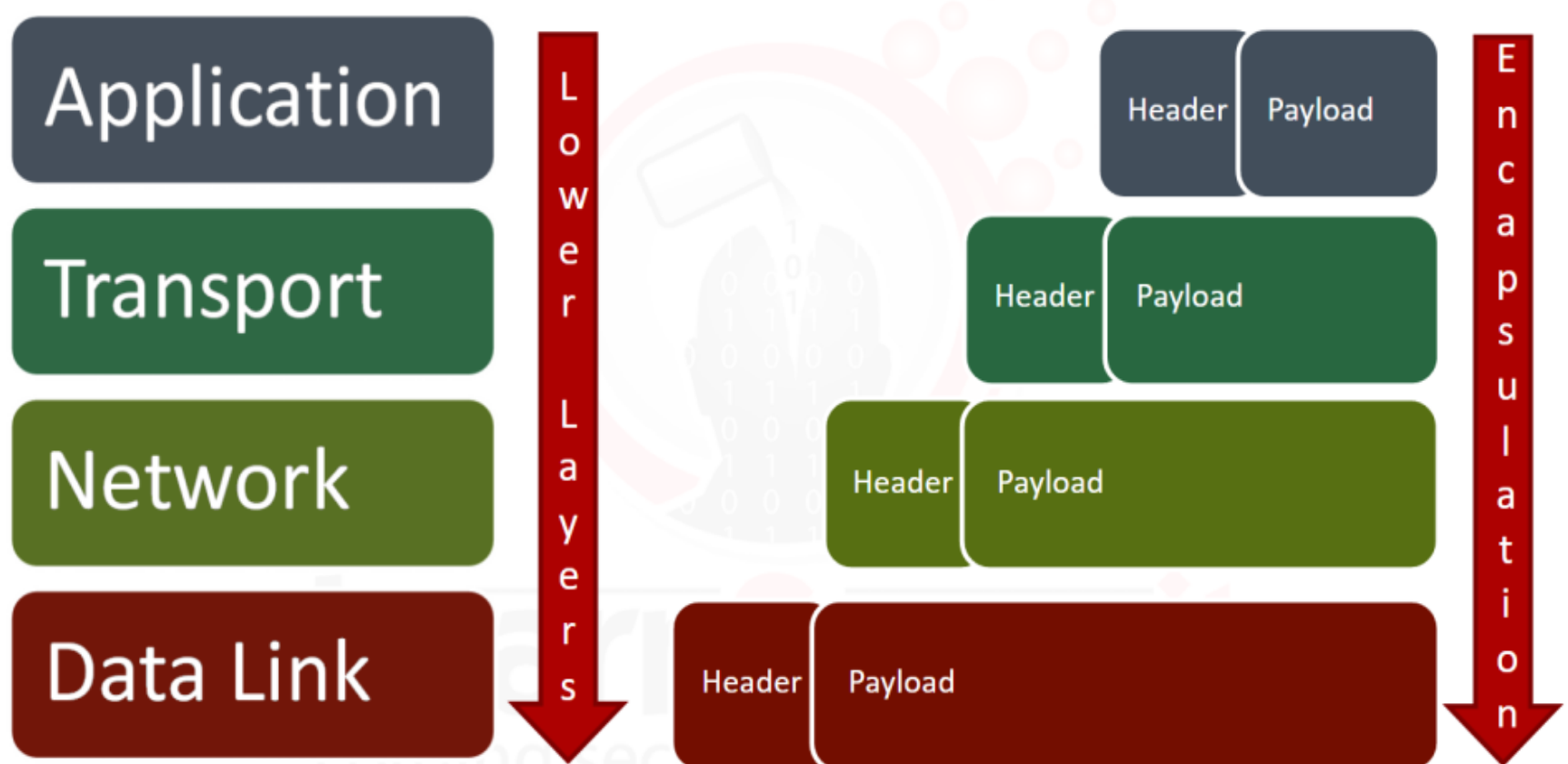
- نزلنا بالجزء بتاع ال **Transport Layer** لل **Network Layer** اللى هو مكون من ال **Header** وال **Payload** ... وال **Payload** دا مكون من ال **Header** وال **Payload** بتوع طبقه ال **Application** ... وبعد كذا ال **Network Layer** كالمعتاد هتضيف ال **Header** بتعها وهتخط ال **Header** وال **Payload** اللى جاين من ال **Transport** فجزء ال **Payload** بتعها ... عامل زي متكون بتغلف طبقه فوق طبقه والطبقه اللى بعدها بتبقا شايله كل اللى فات ... وهتلاقيني مظلها فالرسم التوضيحي عشان توصلك .



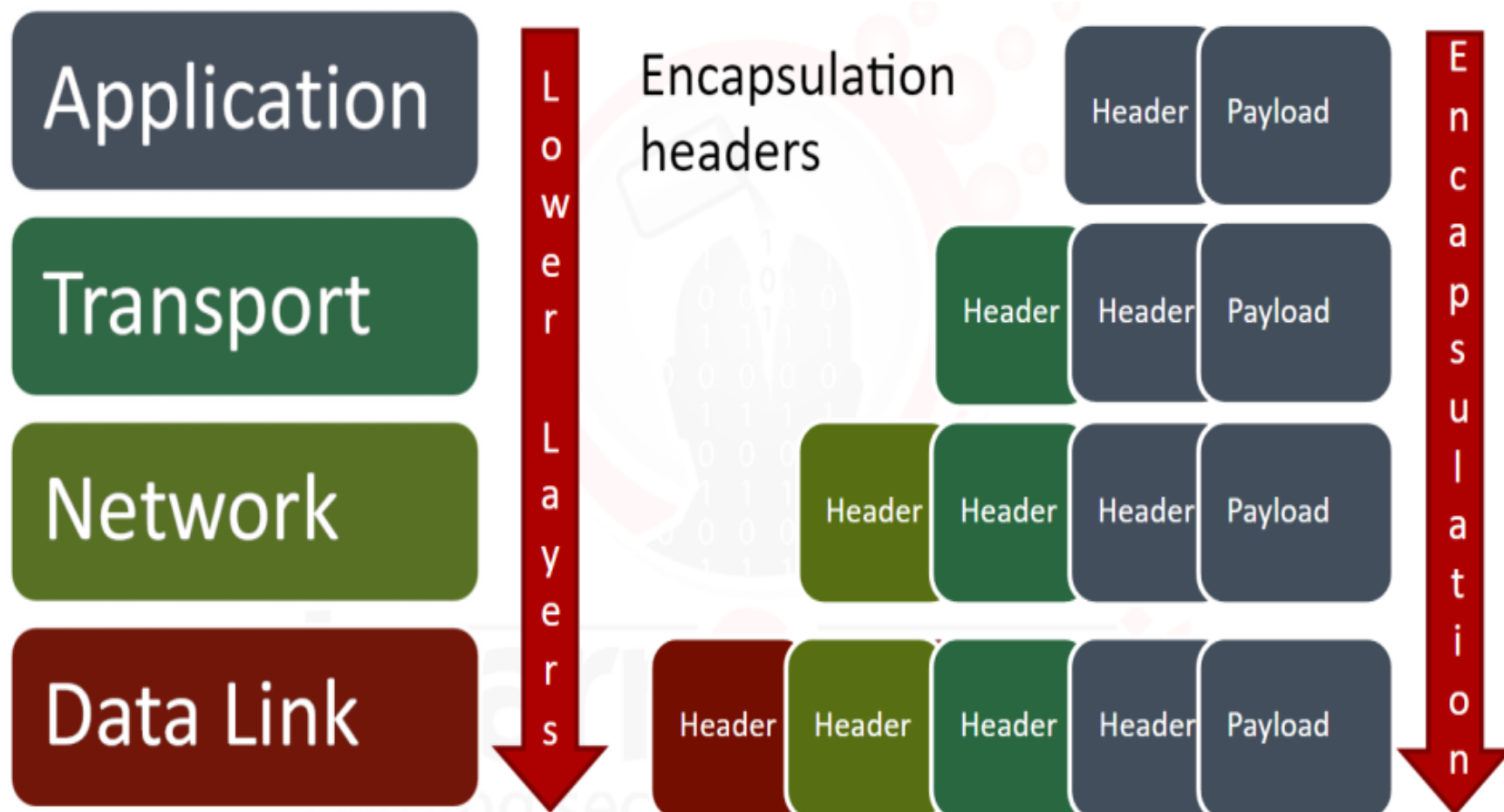
- وبرضه بنفس القصة لأخره هتلاقي ال **Header** وال **Payload** بتوع ال **Network layer** نزلوا لل **Layer** اللى تحتها فجزء ال **payload** واتضاف عليهم ال **Header** بتاعت ال **Layer** اللى بعدها.



- فلما نيجي نبص على آخر **Layer** عندنا وهي ال **Data link Layer** هتلاقي ال **Payload** بتعتها بتحتوي على كل ال **Layers** اللى قبلها .



- العملية اللى شرحناها وشوفناها دي بنسميها ال **Encapsulation** التغليف ولما ال **Packet** بتروح لل **Destination** هتلاقيه بيعمل عكس العملية دي وبيفك الطبقات اللى غلفناها فوق بعض دي عشان يوصل لل **Packet** من أولها من قبل متعدي على ال **4 layers** بتوع ال **TCP/IP Model** ... ودا الشكل النهائي لل **Packet** بتعتنا .



- فكل **Protocol** عندنا بيضيف ال **Header** بتعته زي مشوفنا على المحتوى بتاع ال **Packet** اللى جيله من ال **Layer** اللى قبلها اللى بيتكون من ال **Header** وال **Payload** .

- احنا عارفين ان كل **Host** عندنا فال **Network** بيبقا واخد **IP** عشان يعرف يطلع عال **Internet** وييقا ليه عنوان محدد وزى مشوفنا ال **IP** دا اللى بيديهوله ال **Protocol** الموجود فال **Network Layer** او ال **IP Layer** سميها زي متسميها عادي وهو ال **IP Protocol** الموجود ضمن ال **TCP/IP Model** ... معلومه كدا عالماشي ال **IP Packets** بنسميها ال **Datagrams** عشان لو شفتها فأي حته ... يعني ال **Header** بتاعت ال **IP** وال **Payload** بتعها اللى بيضم ال **Header** وال **Payload** بتوع ال **Transport Layer** اللى تقدر تقول عليها ال **Packet** كامله بتاعت ال **IP Layer** بنسميها ال **Datagrams** .

- تعالى نتكلم بعد كدا عن ال **Devices** الموجود عندنا فال **Network** اللى بتساعد ال **Hosts** فال **Network** انهم يعملوا **Communication** مع بعض ... وأول ال **Devices** اللى هنتكلم عليها دي هو ال **Router** .

- **Traffic** يوصل مبين ال **Source** وال **Destination** فأحنا محتاجين **Path** يمشي من خلاله ... ال **Path** اللى هيمشي فيه ال **Traffic** دا اللى مسؤول عنه هي ال **Routers Devices** ... اللى هي الأجهزة التوجيهيه .

- ال **Router** عبارته عن جهاز وظيفته انه يوصل **Different Network** مع بعض ودا عن طريق ال **Routing Protocols** اللى هي بروتوكولات التوجيه المدمجه فال **Router** المسؤولين انهم يعملوا **Forward** لل **Traffic** اللى طالع من ال **Source** رايح لل **Destination** ... فال **Router** من خلال ال **Routing Table** اللى بيبقا جواها كذا طريق تمشي فيهم ال **Source** توديه لل **Destination** هتقول ال **Source** انهو طريق مفضل ليه عشان يروح لل **Destination** ... وال **Routing Table** هتختار طريق الأسهل والأسرع وال **Secure** لل **Source** عشان يوصل ال **Traffic** لل **Destination** .

- ال **Routing Protocols** بنستخدمها فأيه ؟ ... بنستخدمها عشان نحدد ال **Best Path** اللى اللى هيوصل من خلاله ال **Packet** اللى طالع من ال **Source** رايحه لل **Destination** وبتعملها **Forward** لواحد من ال **Interfaces** الموجوده عند ال **Destination** ... وفالحقيقه بيبقا عندنا أكثر من طريق يوصلنا لل **Destination** ولكن ال **Router** بيختار الطريق الأفضل واللى ال **Cost** بتعته مش مكلفه عليه وأسهل ومختصر ويوصل ال **Packet** بسرعه لل **Destination** ويمكن ال **Network Administrator** هو اللى يحدد الطريق بنفسه اللى هتمشي فيه ال **Packet** طالع من ال **Source** رايحه لل **Destination** .

- تاني جهاز معانا فال **Network** هو ال **Switch** ... بيشتغل زي ال **Router** بس ال **Router** كان بيعتمد على ال **IP Address** انما ال **Switch** بيعتمد على ال **Mac Address** وليس ال **IP** ... وال **Mac Address** هو العنوان ال **Specific** بتاعك انت بشكل **Unique** محدش معاه العنوان دا غيرك فهو **Specific** ليك واللى بيحددك أو بيديك ال **Mac Address** دا هو ال **Switch** ... ال **Switch** عنده **Multiple Interfaces** يعني كذا فتحة يقدر يوصل بيهم كذا جهاز مع بعض لأن ال **Switch** بيوصل كذا جهاز مع بعض انما ال **Router** بيوصل كذا شبكة مع بعض ... ال **Switch** فيه جدول بيعمل ال **Forwarding** للأجهزة اسمه ال **CAM** اللى هو **Content Address Memory** ودا اللى بيبقا متسجل فيه كل **Interface** عليها انهو جهاز بال **Mac Address** بتاعه .

- عاوزين بعد كذا نذكر كام **Protocol** مهمين منهم ال **ARP** وهو ال **Address Resolution Protocol** وال **TCP** اللى هو **Transmission Control Protocol** وعندك ال **UDP** اللى هو **User Datagram Protocol** وكمال ال **DNS** اللى هو **Domain Name Service** كل دول **Protocols** مهمه فال **Analyzing** لل **Network** قدام وفهمك ليهم ولطريقه عملهم بشكل كويس هيخليك تعمل **Hunt** لل **Suspicious Traffic** بشكل احترافي قدام .

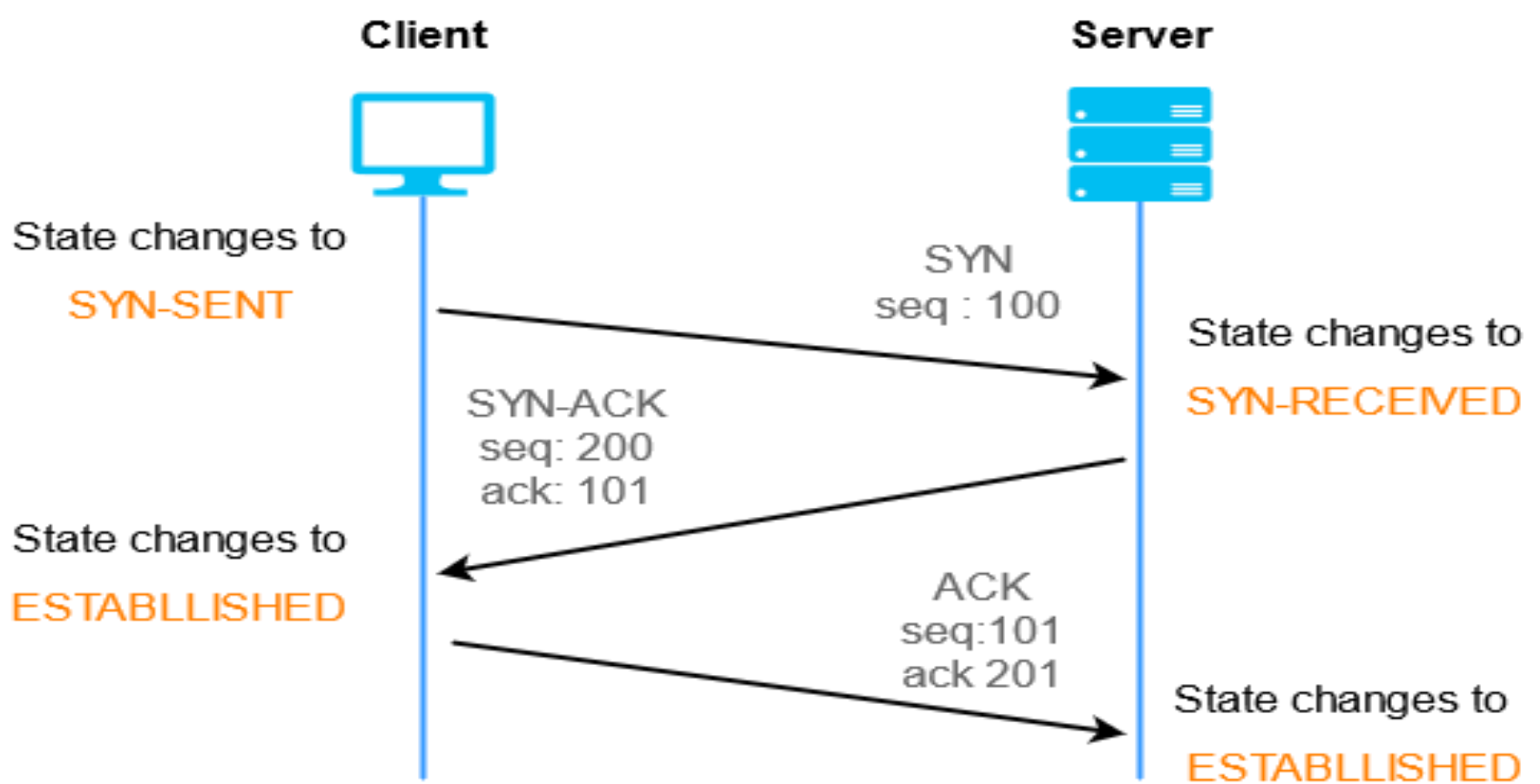
- ال **ARP Protocol** دا شغال ضمن ال **Internet Layer** من ال **TCP/IP Model** ... ودا بنستخدمه لو عاوزين نوصل لجهاز مش عارفين ال **Address** بتاعه ... فلو عندك جهازين عاوزين يتواصلوا مع بعض لازم يكون ال **Source** عارف عن ال **Destination** ال **IP** وال **MAC Address** بتاعه ... ودايما ال **Source** بيكون عارف ال **IP** بتاع ال **Destination** انما معضلته فال **Mac Address** مش عارفه !!.

- فال **Source** هيبعت فال **Network** ال **ARP Request** وهييط
 فال **request** دا ال **IP Address** بتاع ال **Destination** وهييط
 ال **Mac Address** هو ال **Broadcast** اللى هو بيكون
FF:FF:FF:FF:FF:FF ودا بنحطه فحاله اننا ك **Source** مش
 عارفين ال **Mac Address** بتاع ال **Destination** ... فالرساله دي
 هتتبع لل **Switch** وال **Switch** اول ميشوف ال **Packet** اللى
 جباله وفيها ال **Destination** بيكون فيه ال **Broadcast** فيقوم
 اوتوماتيك واخد ال **Packet** دي وعاملها **Forward** لكل الأجهزة أو
 ال **Hosts** اللى موجوده فال **Network** ... وال **Packet** دي كدا
 وصلت لكل ال **Hosts** الموجوده عال **Network** وكدا اللى هيرض
 فقط عال **ARP Request** هو ال **Destination** صاحب ال **IP**
 هيقوم راضض بال **ARP Reply** اللى فيه ال **Mac Address** بتاعه
 وبكدا يكون مع ال **Source** ال **IP** وال **Mac** بتوع ال
Destination فال **Connection** مابينهم يبقا تمام ويقدرُوا يبعثُوا
 ال **Packets** لبعض ويستلمُوا من بعض .

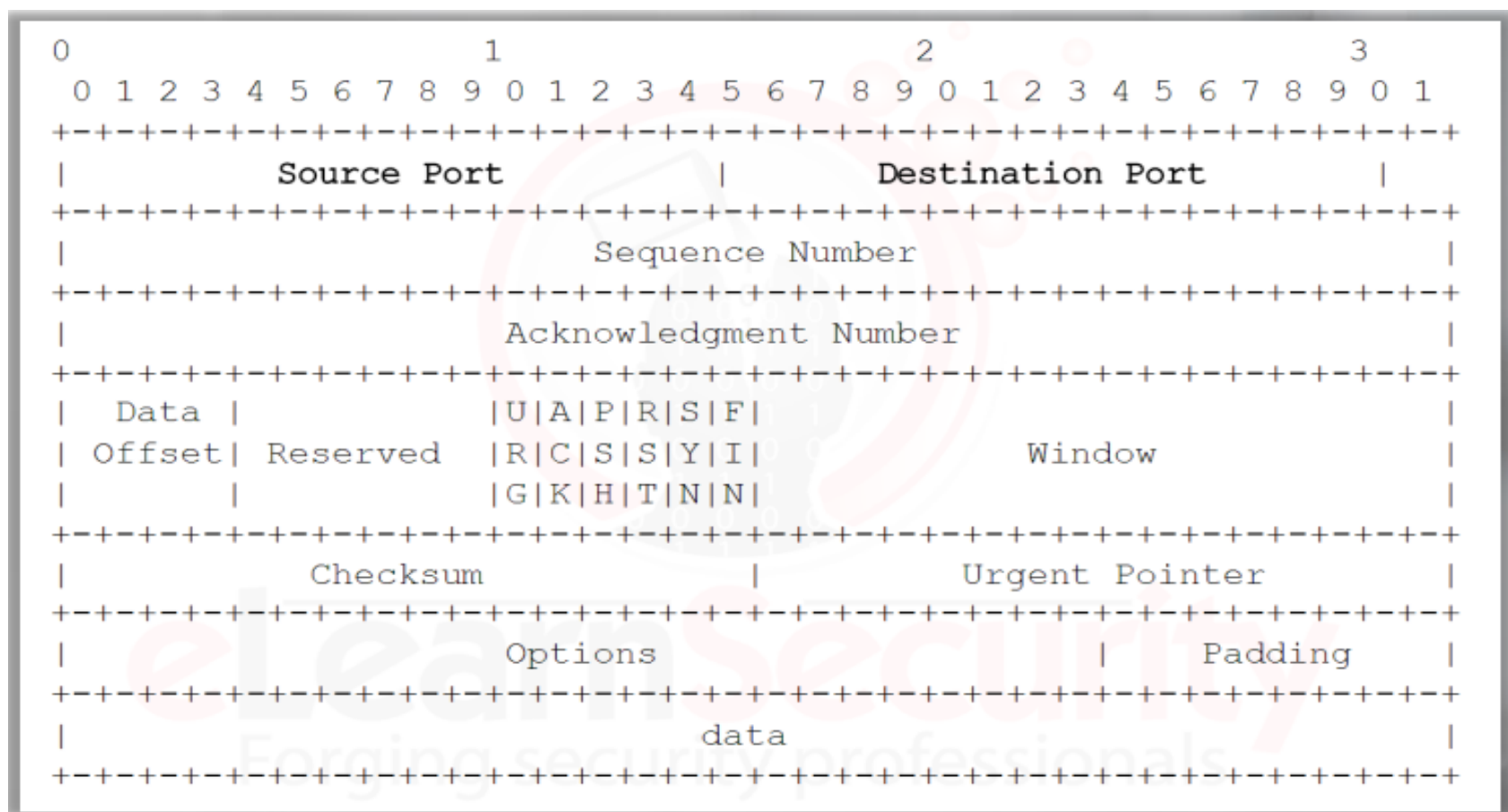
- وطبعا انت ك **User** ملكش دعوه بالكلام دا خالص ال **Operating**
System اللى عندك هو اللى بيقوم بالكلام دا بشكل اتوماتيك بس
 يهمني انك تفهم الدنيا ماشيه ازاي جوا ال **Network** ... ودا شكل ال
ARP Broadcast من **Tool** هتشوفها كتير فال **Network**
Hunting وهي ال **Wireshark** ودي بتعمل **Analyze** لل
Packet اللى بتتبع من ال **Source** لل **Destination** والعكس
 فال **Network** .

1	0.000000000	3com_aa:01:9c	Broadcast	ARP	42	Who has 10.11.12.4? Tell 10.11.12.145
2	0.000311000	CadmusCo_f3:b4:70	3com_aa:01:9c	ARP	60	10.11.12.4 is at 08:00:27:f3:b4:70
41	5.015682000	CadmusCo_f3:b4:70	3com_aa:01:9c	ARP	60	Who has 10.11.12.145? Tell 10.11.12.4
42	5.015691000	3com_aa:01:9c	CadmusCo_f3:b4:70	ARP	42	10.11.12.145 is at 00:01:02:aa:01:9c
84	53.576305000	CadmusCo_f3:b4:70	3com_aa:01:9c	ARP	60	Who has 10.11.12.145? Tell 10.11.12.4
85	53.576320000	3com_aa:01:9c	CadmusCo_f3:b4:70	ARP	42	10.11.12.145 is at 00:01:02:aa:01:9c
210	71.133897000	CadmusCo_f3:b4:70	Broadcast	ARP	60	Who has 10.11.12.3? Tell 10.11.12.4

- ال Protocol اللى بعده وهو ال **TCP** ودا بيستخدم ال **3 Way Handshake** وبتكون بالشكل دا .

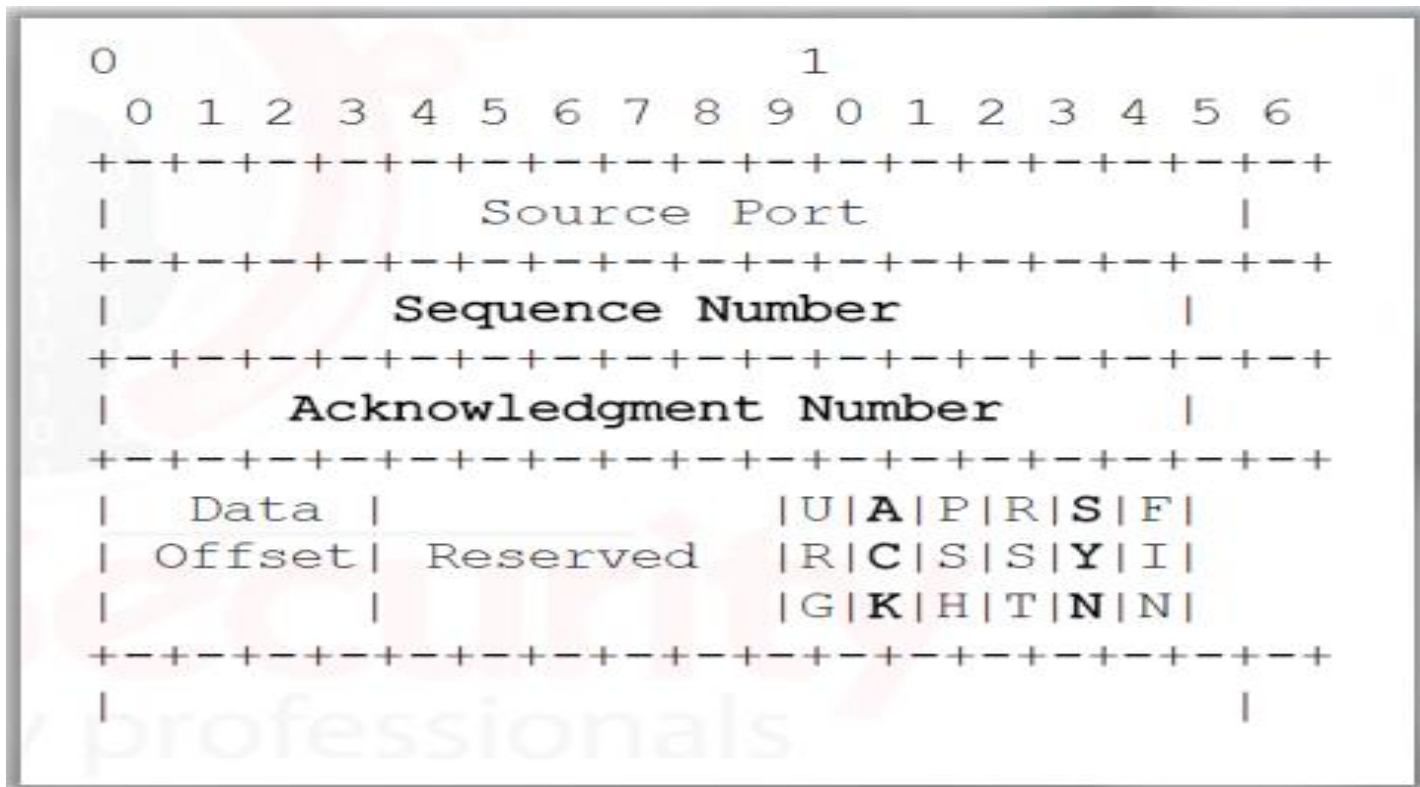


- ال **TCP** قبل ميرسل اي **data** بيقوم بعملية ال **3way handshake** عشان يضمن ان ال **data** وصلت وكله تمام وعشان ي **establish** ال **connection** ما بين ال **2 devices** وهنشوف ال **3Way handshake** بعدين ... تعالى نشوف ال **Header** الخاصه بال **TCP Protocol** .

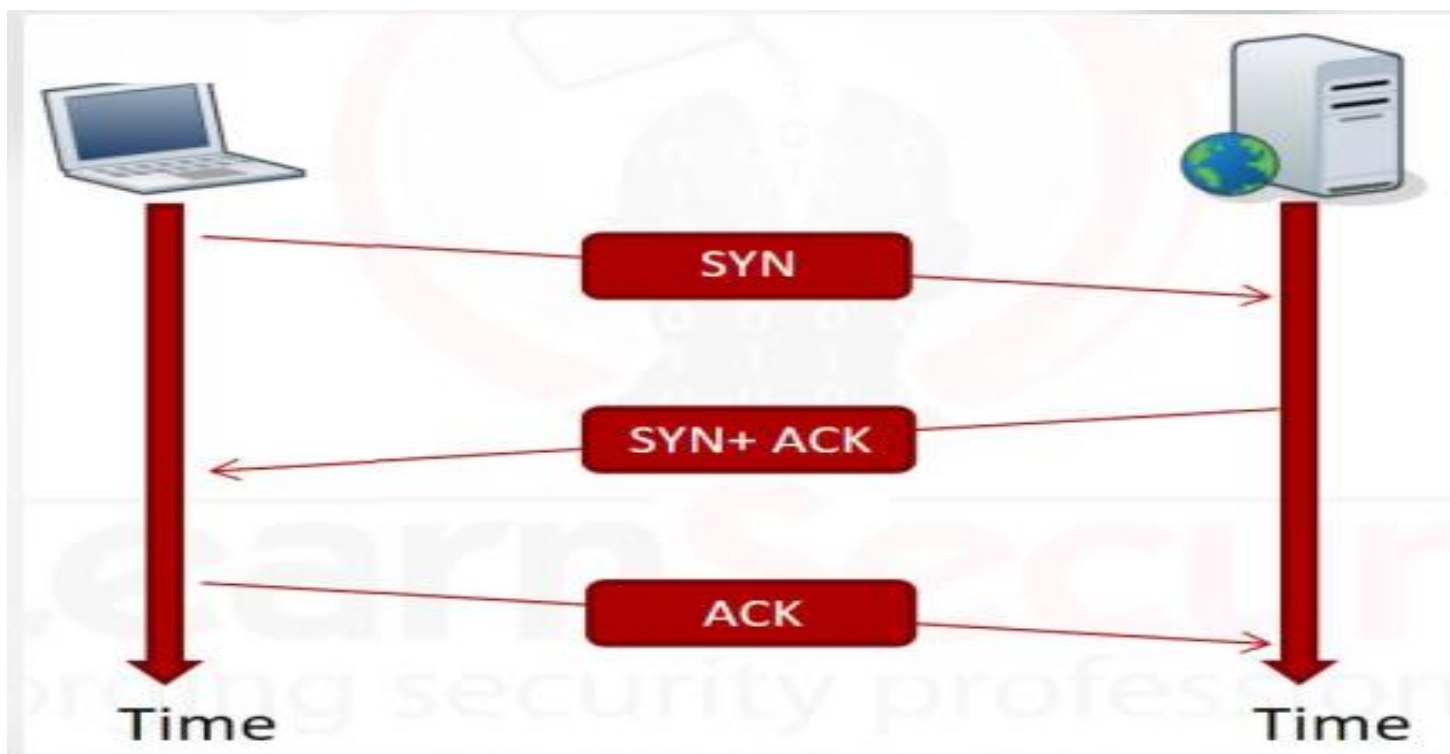


- هتلاقي ال **Header** فيها كذا جزء منهم واهم ال **Source Port** وال **Destination Port** وال **Flags** وال **Sequence Number** وال **Acknowledge Number** .

- أثناء عملية ال 3 Way handshake مابين ال Source وال Destination نحتاج من ال Header بتاعت ال TCP التالي .



- نحتاج ال Sequence Number وال Acknowledge Number ومن ال Flags هناخد ال Syn وال ACK ... وتعالى نشوف ال 3 Way handshake بتم ازاي مابين ال Client وال Server .



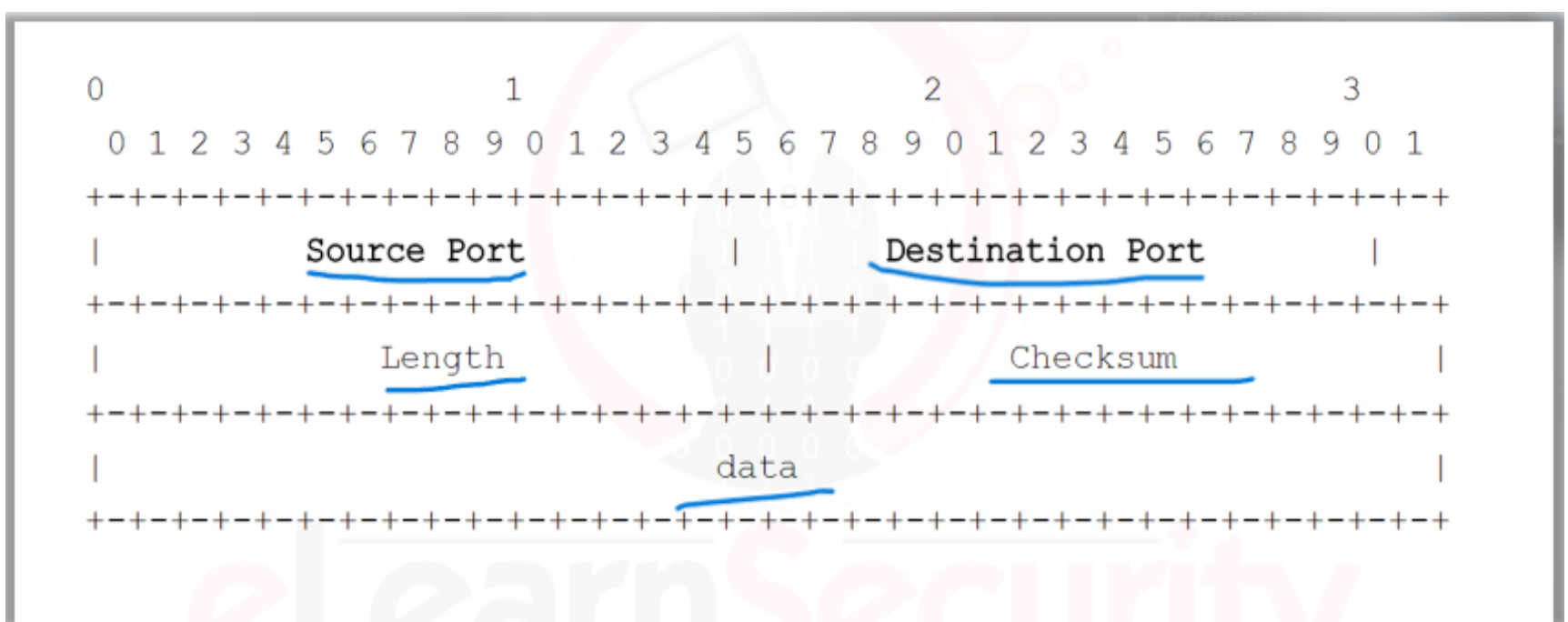
- هتلاقي ال Client بيعت لل Server ال Syn اللى هو عاوز ابدء معاك اتصال وبعد كدا ال Server بيرض عليه بال Syn+Ack بيوافق على طلبه وبعد كدا ال Client بيعت لل Server ال ACK وبكدا بتم عملية ال 3 Way handshake وال Connection بقا Establish مابين ال 2 Devices فيقدروا بيعتوا ويستلموا Data من بعض ... والكلام دا هتلاقيه تفصيلي فكورس ال eCIR ارجعله .

- تعالى نشوف ال **3 Way handshake** عن طريق ال **Wireshark**

Below we can see the 3-way handshake through Wireshark.

56	48.569508000	10.100.13.37	10.11.12.145	DNS	115 Standard query response 0x59ca
57	48.569793000	10.11.12.145	146.128.7.4	TCP	74 34630 > http [SYN] Seq=0 Win=29200 Len=0 MSS=1460 SACK_PERM=1 TSval=213757 TSecr=0 WS=128
58	48.570133000	146.128.7.4	10.11.12.145	TCP	74 http > 34630 [SYN, ACK] Seq=0 Ack=1 Win=14480 Len=0 MSS=1460 SACK_PERM=1 TSval=5748488 TSecr=213757 WS=8
59	48.570186000	10.11.12.145	146.128.7.4	TCP	66 34630 > http [ACK] Seq=1 Ack=1 Win=29312 Len=0 TSval=213757 TSecr=5748488
60	48.591479000	10.11.12.145	146.128.7.4	HTTP	363 GET / HTTP/1.1

- تعالى بعد كدا نشوف ال **UDP** اللى بنستخدمه بشكل **Specific** عشان ننقل ال **Packet** من ال **Source** لل **Destination** زي ال **Streams** بتاعت ال **Gaming** وغيره اللى هتلاقي ال **UDP** بنستخدمه فالحالات دي عشان أسرع من ال **TCP** فتعالى نشوف ال **UDP Header** عامله ازاي ومش هتلاقي فيها **Details** كتير زي ال **TCP**.



- وطبعا أنصحك تشوف باقي ال **Protocols** اللى ذكرناها وطريقه شغلها ودي هتلاقيها فكورس زي ال **eCIR** بالتفصيل بس هي دا مش موضع نقاشها ... تعالى بعد كدا ندخل على ال **Common Ports** ونتعرف عليهم ... ال **Port** عباره عن فتحه وهميه فال **APP** اللى هتتواصل معاه وال **APP** عشان يشتغل محتاج يكون عنده **Protocol** يشتغل بيه ... زي **Firefox** كدا دا **APP** عشان يشتغل محتاج يكون ال **HTTP Protocol** شغال ودا عشان يشتغل محتاج يخرج من ال **PC** بتاعي من **Port** أو فتحه معينه وهميه ملهاش وجود **Physical** على جهازى وانت تقدر فأى وقت تعمل **Close** للفتح دي وتقف ال **Session** مابينك وبين ال **Server**.

- ال **HTTP Protocol** بيشتغل على **Port 80** وهكذا مع باقي ال **Ports** وكل **Port** ليه **Protocol** معين شغال عليه ب **Service** معينه شغاله عليه وانت ك **Threat Hunter** مطالب انك تعرف ال **Most Common Ports** عشان نعرف شغلهم ال **Normal** عامل ازاي ونعرف ن **Hunt** ال **Suspicious** منه بعد كدا وهتلاقيهم موجودين فالصوره دي .

Some common ports are:

- SMTP (25)
- SSH (22)
- POP3 (110)
- IMAP (143)
- HTTP (80)
- HTTPS (443)
- NETBIOS (137, 138, 139)
- SFTP (115)
- Telnet (23)
- FTP (21)
- RDP (3389)
- MySQL (3306)
- MS SQL Server (1433)

2.3 Packet Analysis & Tools:

- عشان نعمل **Analysis** لل **Packet** بديهي اننا ناخذ من ال **Packet** دي **Capture** اللى بنقول عليه ال **PCAP** ... يعني عاوزين ناخذ لقطه من ال **Traffic** دا بيتم بكذا طريقه منها ال **Live** اننا ناخذ **Capture** لل **Traffic** دا بطريقه **Live** من ال **Wireshark** وممكن يكون عندنا ال **Traffic** دا متخزن عندنا من فتره وهنجيبه نعمله **Analysis** واللى هيبتلك ال **Packet Capture** هما ال **Network Team** من خلال ال **Alerts** اللى هيبتوهاك انت ك **Threat Hunter** عشان تعملها **Investigation** ... وفبعض الحالات هتلاقي ال **Network Team** بيبتلك ال **Alert** دا وبيقولك خد انت ال **Packet Capture** واعمل **Analysis** لل **Incident** .

- ال **Packet** اللى بنعملها **Capture** دي بتكون **RAW** يعني **Data** خام لما بنأخذها بال **Wireshark** عشان نعملها **Analysis** ... ولما بنأخذ بال **Wireshark** ال **Capture** بتعنا بيتحفظ عندنا عالجهاز بامتداد **PCAP** ... وكل ال **Tools** الخاصه بال **Analysis** لل **Network Traffic** هتلاقيها بتعتمد ال **PCAP** فال **Files** اللى بتحتوى على **Traffic** عاوز تعمله **Analysis** سواء كان ال **File** دا هتعمله **Export** أو **Import** فلازم هتلاقي امتداد ال **PCAP** موجود ... فتقدر تقول ال **PCAP** هو ال **Standard Format** بتاع ال **Packet Capture** .

- فلو انت **Threat Hunter** أطلب منك تعمل **Network Analysis** وتطلع ال **Suspicious Traffic** منه هتعمل ايه؟؟ ركز فالجي .

- ال **Network Team** عندنا فالمؤسسه جالهم **Alert** ب **Unusual Traffic** عندهم فال **Network** وممكن يكون فال **Network** كلها أو فجزء أو **Segment** من ال **Network** ... انت ك **Threat Hunter** مش هتروح تعمل **Hunt** فال **Network** كلها !! انت بتشوف ال **Alert** جالك من انهو قسم فال **Network** زي ال **HR** أو ال **Engineer's** وتدخل جوا الفرع اللى جوا القسم دا وتشوف ال **Alert** جايلك من انهو جهاز وتبدء تشتغل مش هنعمل **Hunt** ف **Network** بتحتوى على ألاف ال **Endpoints** لاء احنا بنفضل نتعقب ال **Threat** لحد منوصل للمنطقه اللى حصل فيها ال **Threat** بالتحديد ... بنضيق ال **Scope** اللى هنشتغل عليه ... وهتلاقي بعد أما ال **Network Team** جالهم **Alert** بال **Unusual Traffic** فال **Network** هيبعتوا **Alert** لل **IT Security Team** وانت ك **Threat Hunter** جوا ال **IT Security Team** ودا طبعا فالمؤسسات الكبيره وفيه فبعض المؤسسات هتلاقيهم بيبعتوا لل **Threat Hunter** بشكل **Direct** .

- بعد كذا ال **Threat Hunter** يبتدي يعمل **Hunting** ... وبعد كذا ال **Network Team** بي **Provide** ال **Threat Hunter** بال **PCAP** **File** عشان يعمل **Analyze** وفي بعض ال **Cases** ال **Threat Hunter** بيكون مطلوب منه يعمل **Live Packet Capture** .

- وطبعا ال **Threat Hunter** زي مقولنا مش هيعملك **Hunt** فال **Network** كلها دا صعب جدا ... لاء احنا بنحدد **Segment** معين أو **Department** جوا ال **Network** أو **Protocol** معين أو **IP** معين أو **Port** معين هنشتغل ونركز عليهم ... فمينفعش نشتغل على **Network Traffic** بشكل عشوائي هنلاقي عندنا **Terabytes** من ال **Network Traffic** مطلوب منك تعملهم **Analysis** هل هتقدر !؟

- المؤسسات دلوقتي بتستفيد من **Concept** بتطبقه وهو ال **Defense-in-Depth** ... ودا بيساعدنا ن **Create** ال **Security Monitoring Program** فالمؤسسه عندنا ... ودا معناه انها بتحط كذا **Defense Device** زي ال **IPS** وال **IDS** وال **Firewall** وال **Proxy** وغيرهم وأي **Suspicious Traffic** هيمر من خلال ال **Devices** دي هيجلنا بيه **Alert** هنروح نعمل **Investigation** و **Hunt** فالمكان اللى جالنا منه ال **Alert** عشان نكسب وقت ونضيق ال **Scope** عال **Threat** ... وطبعا هفكر كذا بال **CTI** اللى هي ال **Cyber Threat Intelligence** اللى اتكلمنا عنه بشكل تفصيلي ف **Module** كامل فال **Section** الأول من الكورس ... فهمك لل **CTI** وازاي تستخدمه فال **Hunting** لل **Threat** هيساعدك فأنك تعمل **Hunt** لأي **Suspicious Traffic** مثلا جايلك من **IP** معين انت من خلال ال **Information** اللى جمعتها من خلال ال **Search** عرفت ان ال **IP** دا **Malicious** ... فساعتها لو شفت أي **Traffic** جاي منه لل **Network** عندك هتعرف انه **Suspicious** و هتعمله **Detect** علطول .

- فانت بتشوف ال **Alert** وتقرئه وتشوف هل فيه حاجه **Malicious** بت **Match** ال **Information** اللى جمعتها من ال **CTI** كل ده قبل نمعمل ال **Analysis** لل **Packet** اللى عملناها **Capture** فدا هيووفر علينا وقت كتير ولو معرفناش ن **Match** ال **Alert** اللى جيلنا بال **Information** اللى جمعناها بال **CTI** ساعتها هنروح بال **Tools** بتعتنا زي ال **Wireshark** نمعمل **Analysis** لل **Traffic** اللى عملنا له **Capture**.

- فبعض الحالات عندنا هتلاقى واحنا بنعمل **Hunt** ال **Suspicious Traffic** ظاهر فهنعمله **Detect** ... وفي بعض الحالات هتلاقى فيه **Suspicious Traffic** عندنا فال **Network** و ال **Devices** زي ال **IPS** وال **IDS** توقفت عن العمل للحظات معينه فوقت معين ودا طبعا هتلاقىه الوقت اللى كان ال **Suspicious Traffic** فيه معدي من خلال ال **Network** ... ساعتها احنا بنعمل **Deep Investigation** بنفسنا وبندخل جوا ال **Packet** اللى عدت فالوقت وقبل الوقت دا اللى كانت فيه **Devices** زي ال **IPS** وال **IDS** معمولها **Disable** عندنا عال **Network** لحد محنا شوفنا ال **Threat** ونعملها **Deep Analysis** ونعمل **Hunt** ال **Suspicious Traffic** ... فال **Cases** اللى شبه دي ال **CTI** مش هتفيدك بشكل كلى انك تعتمد عليها فقط ... لاء هتحتاج تنزل بنفسك عال **Packet** وتعملها **Deep Investigation** ... فلازم انت ك **Threat Hunter** تكون عارف ال **Network Infrastructure** بتعتك عامله ازاي وايه ال **Devices** زي ال **Firewall** وال **Routers** كام ومكانهم فين وال **Switches** وهكذا وباقي ال **Devices** الموجوده فيها وأماكنها فين وكمان ال **Endpoints** بتاعت المؤسسه موجوده فين وال **Network Rules** الخاصه بالتعامل مع ال **Endpoints** فالمؤسسه عندنا ودا هيخليك تتعامل مع ال **Incident** أو ال **Threat** اللى هتحصل عندنا فالمؤسسه بشكل منسق ومرتب فلازم يكون عندنا ال **Full map** لل **Network**.

- معلومه كدا عالماشي ... يفضل ال **Threat Hunter** يكون بيعرف يتعامل مع ال **Windows OS** وكمان ال **Linux OS** عشان دا هيفيده قدام عشان أغلب ال **Tools** بتاعت ال **Offensive** بنتعامل من خلالها بال **Linux OS** ودا اللي أغلب ال **Attackers** بيلجأوا ليه فال **Attacks** بتعتهم نظام زي **Kali** أو **Parrot** ودول **Linux** فانت ك **Threat Hunter** يفضل تكون بتعرف تتعامل مع ال **Linux** عشان تعرف ازاي ال **Attackers** بينفذوا ال **Attacks** بتعتهم وايه ال **Tools** اللي بيستخدموها فال **Attack** ... أما ال **Defensive Side** هتلاقيهم بيستخدموها فالأغلب ال **Windows OS** فعشان تعرف تتعامل معاه برضه ك **Threat Hunter** محتاج تكون دارس حاجه زي **MSCA** عشان تعرف ت **Administrator** ال **Windows OS** وهنا أحب أرشحلك كورس **مهندس محمد زهدي** على **YouTube** لو حابب تخوض جزء ال **MSCA** هيفيدك جدا .

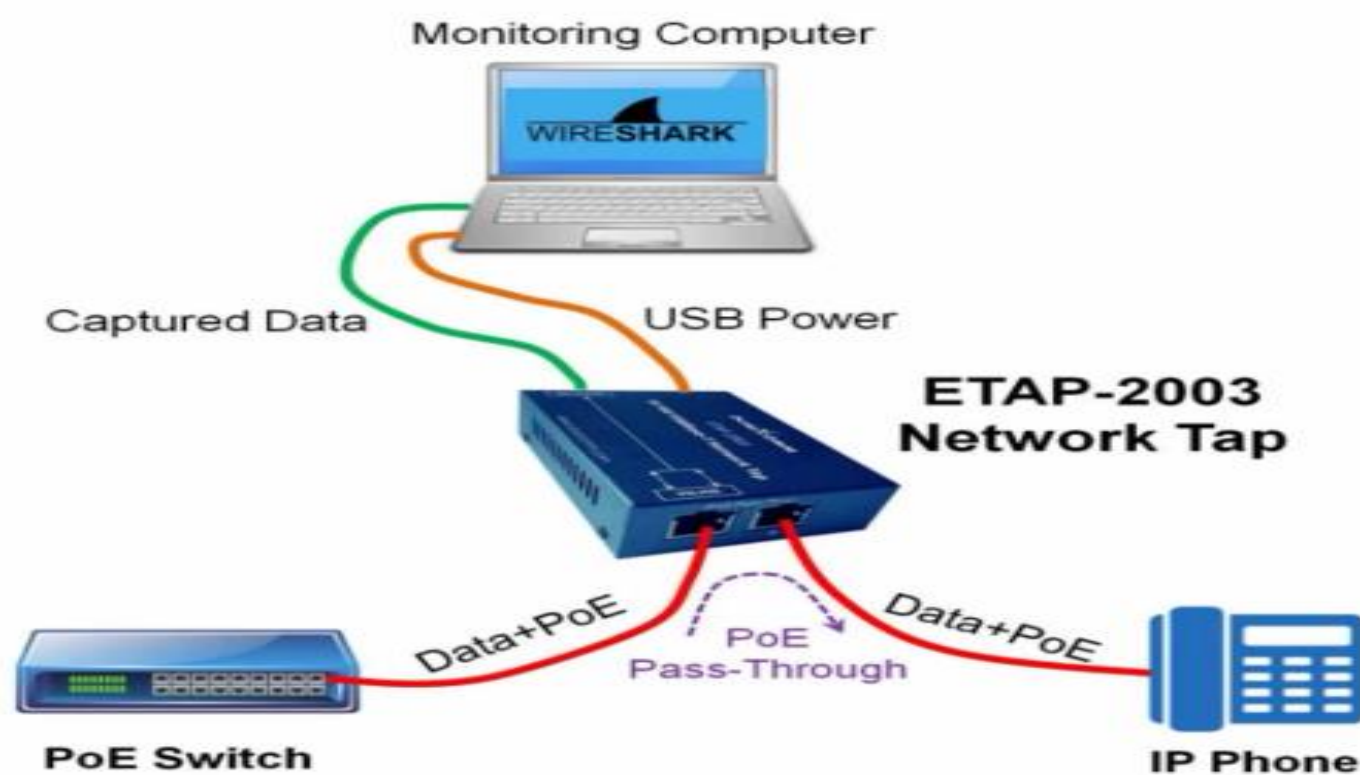
- قبل منشغل عال **Capture** اللي خدناه أو اللي هنعمله **Live** من ال **Network** فيه كام نقطه مهمين لازم تاخد بالك منهم وهم ... محتاجين فالأول نعمل **Test** عال **Capture** اللي خدناه دا عن طريق اننا مثلا نعمل **Capture** لل **Traffic** بتعنا اللي هنشتغل عليه لمدى 5 دقائق مثلا وبعد كدا نعمل عليه ال **Test** ونشوف هو دا اللي احنا عاوزينه ولا لاء ... بعد كدا نتأكد من ال **Machine** اللي بتعمل منها **Capture** فيها **Computing Power** ولالاء ودا ليه ؟ لأننا لو عملنا **Capture** ل **Traffic** متواضع فالأمكانيات زي جهاز مثلا **CPU** بتعته وال **RAM** مش قد كدا فدا هيخلي ال **Traffic** اللي علمناله **Capture** مش كافي بالنسبه لينا عشان نعمله **Analysis** ... فانت فالحاله دي هتفكك انك عملت **Capture** لكل الحاجات اللي احنا عاوزين نشتغل عليها بس فالحقيقه انت عملت **Capture** للحاجات اللي جهازك بس عرف يهندها ... فلانم تتأكد ان ال **CPU** بتاعت جهازك قويه وال **Resources** فالعموم عشان ال **Capture** اللي خدناه يبقا كامل .

- النقطة الثالثة معنا نتأكد ان عندنا **Enough Disk Space** ... فالمساحة دي لابد منها لأننا ك **Threat Hunters** فشغلنا بناخد لقطة من ال **Traffic** اللى هي ال **Packets** اللى معديه فال **Network** وبنحفظها عندنا عال **PC** بتعنا لحد منعملها **Analysis** فلازم يكون عندنا مساحة كافيه لل **Traffic** اللى هنخزنه عندنا عال **PC** ... ولو برضه كنت بتاخذ ال **Capture** دا من ال **Virtual Machine** تتأكد ان فيها **Space** كافي ... وكل متطول مده ال **Capture** بتعتك تلقائي لازم تزود المساحة اللى على جهازك لأن اللى بياخد **Capture** لمده ساعه مش زي اللى بياخد **Capture** لمده يومين أكيد ! فدا يدك **Hint** انك لازم تهتم بال **Space** على جهازك.

- النقطة الأخيره معنا اننا هنعمل **Capture** لكل ال **Traffic** اللى ماشي فال **Network** فلازم هنعدي على جهاز زي ال **Switch** فال **Network** فلازم انت ك **Threat Hunter** تكون مفعل عندك على ال **Switches** خاصيه ال **Mirror Port** ودا اللى بيعمله ال **Network Engineer** ودا معناه ان اي **Traffic** هيعدي من خلال ال **Switch** يتاخذ منه **Copy** ويتبعت للجهاز بتعتنا اللى هو جهاز ال **Threat Hunter** ودا معنى تفعيل ال **Feature** بتاعت ال **Mirror Port** ودي هتخلينا نتأكد اننا بنستقبل كل ال **Traffic** اللى بيمر خلال ال **Network** ويجي عال **Port** بتعنا اللى ال **Network Engineers** مفعلين عليه خاصيه ال **Mirror Port** وكل شركه هتلاقيها مسميه ال **Feature** دي بأسم **Specific** ليها ... يعنى مثلا شركه **Cisco** هتلاقيها مسميه ال **Mirror Port** بأسم ال **Cisco Switched Port Analyzer** اللى هو **SPAN** وكل شركه هتلاقيها بتختلف من حيث تسميه ال **Port Mirroring** زي موضحنا ... فبيقا ال **Port Mirroring** باختصار خالص أي **Traffic** داخل لل **Network** عندنا هتلاقينا بناخد منه **Copy** ونوديه لل **Port** اللى عاملين عليه ال **Feature** دي على جهازنا اللى متوصل فنفس ال **Local Network**

- طب فيه فبعض ال **Cases** للأجهزة اللى فالمؤسسة عندك مش هتلاقي ال **Port Mirroring** فهتلاقيه **unavailable** فعندك حل زي انك تجيب عندك فالمؤسسة جهاز **Physical** زي ال **TAB** اللى بنسميه **Tapping the Network Cable** وهو بيجبك ال **Traffic** من ال **Switch** ويودي له **Threat Hunter** ... وعندك طريقتين تانيين وهما ال **ARP Spoofing** اللى بيسبب ال **Man in the middle Attack** اختصاره ال **MITM** وال **MAC Flooding** اللى بيحول ال **Switch** بتاعك ل **HUB** دول عن طريق ال **Network Administrators** تقدر تفعلهم عندك وبرضه هيجيلك نسخه من ال **Traffic** على جهازك انت ك **Threat Hunter** ... ودا شكل ال **TAB Device** برضه عشان تجمع بيشتغل ازاي .

Top Pick For Wireshark

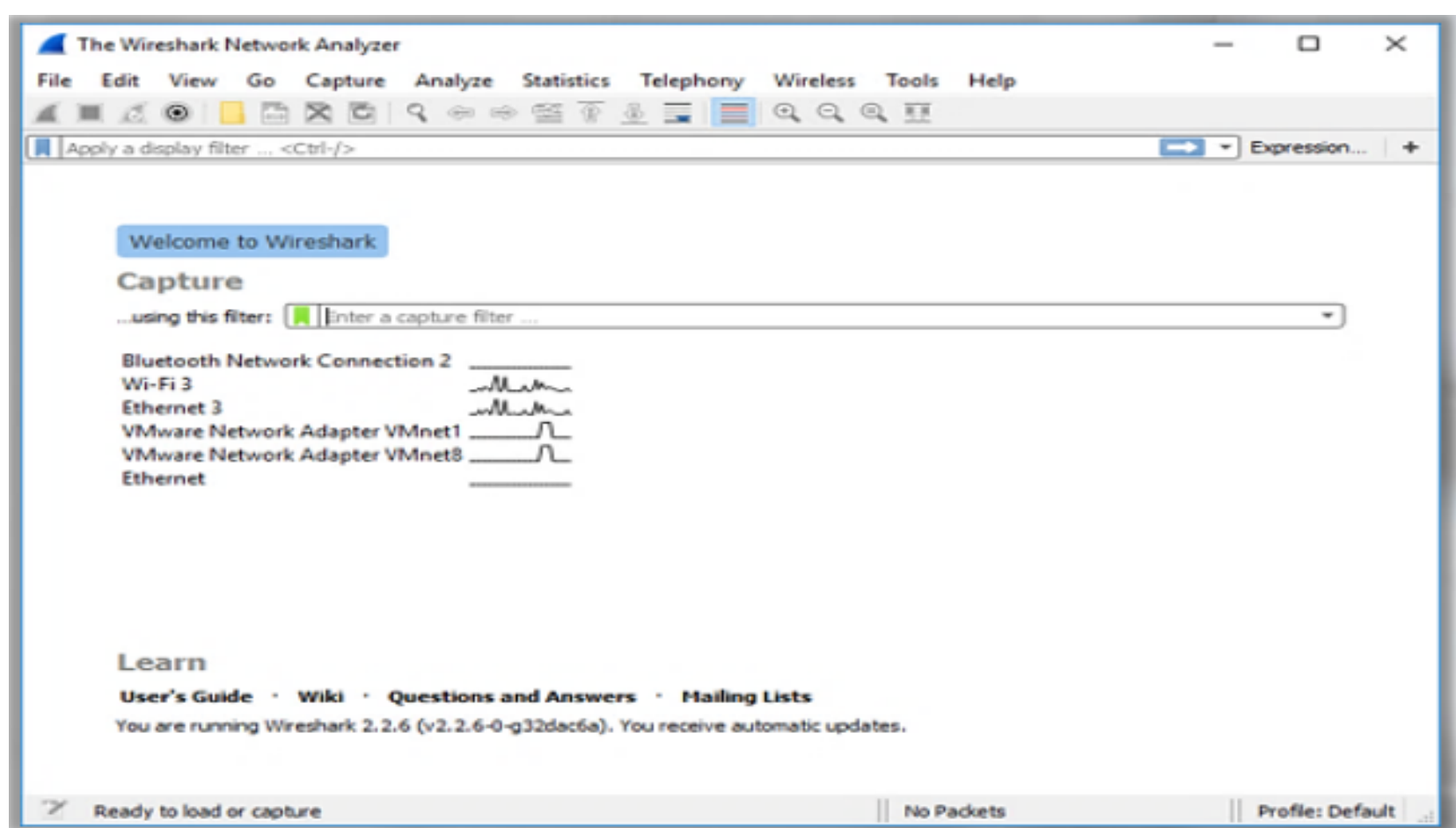


- عشان نعمل ال **Capture** عال **Target** بتعنا محتاجين ال **System** بتعنا اللى هنشتغل عليه لازم يكون عنده بعض ال **Libraries** ... يعنى مثلا فال **Unix** أو ال **Linux System** عندنا **Library** اسمها ال **Libpcap** ودي المسؤوله عن ال **Packet Sniffing** عندنا فال **Network** ... بمعنى **Tools** زي ال **Wireshark** أو ال **TCPDump** عشان يشتغلوا ويعملوا **Capture** لل **Traffic** لازم يتأكدوا الأول ان ال **Machine** دي وخصوصا ال **OS** بتعها عليه ال **Library** اللى هي ال **Libpcap** .

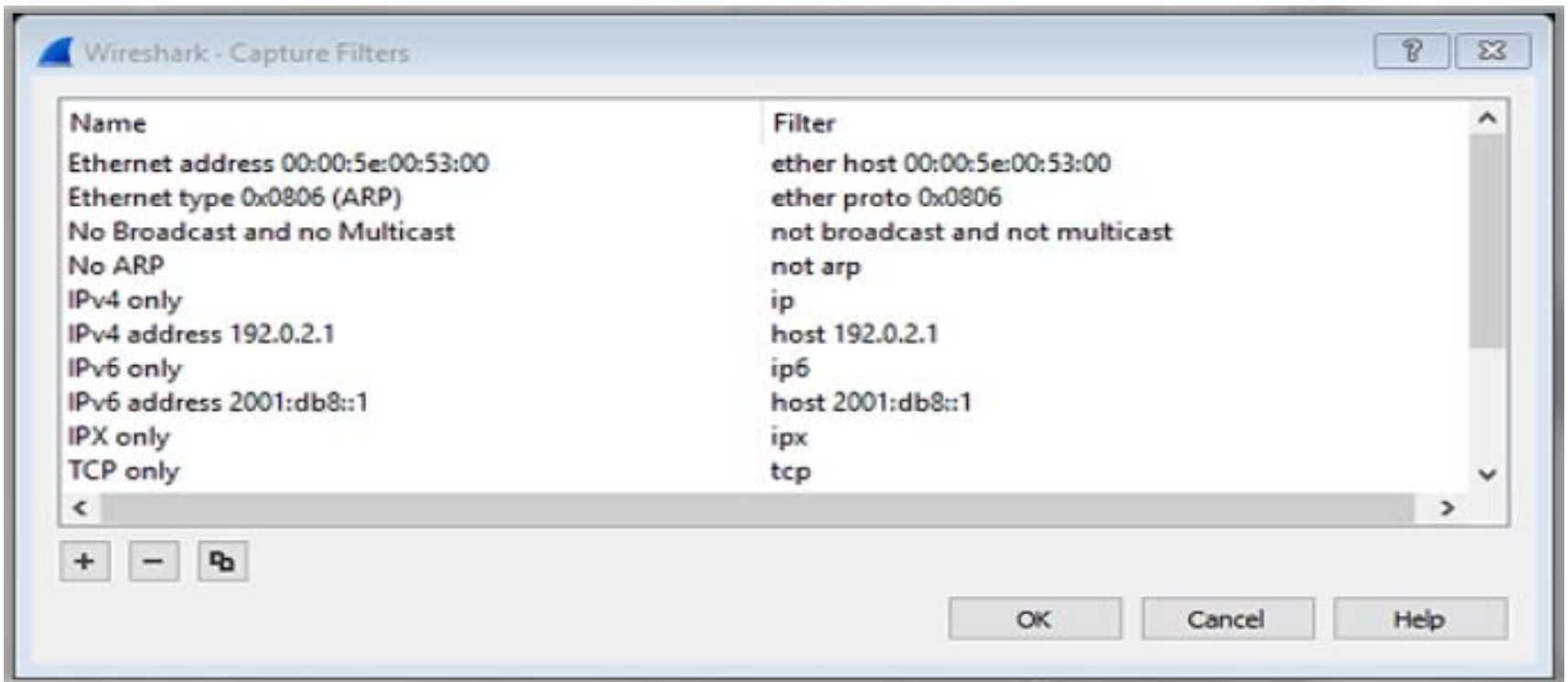
- لو ال **Machine** اللى هنتغل عليها هي ال **Windows** هيقا عندنا **Library** اسمها ال **Winpcap** عشان نعرف نعمل عال **Windows** دا ال **Capture** لل **Traffic** بتعنا.

- وطبعا ال **Tools** اللى هنعمل بيها ال **Analysis** زى ال **Wireshark** وال **TCP Dump** هتلاقيها هي بشكل **Automatic** بتنزل ال **Libraries** دي مجرد متنزل ... فتعالى دلوقتي نشوف ال **Tools** بتعتنا اللى هنستخدمها فال **Hunting** لل **Traffic** وهنعمل بيها ال **Analysis** لل **Traffic** بتعنا.

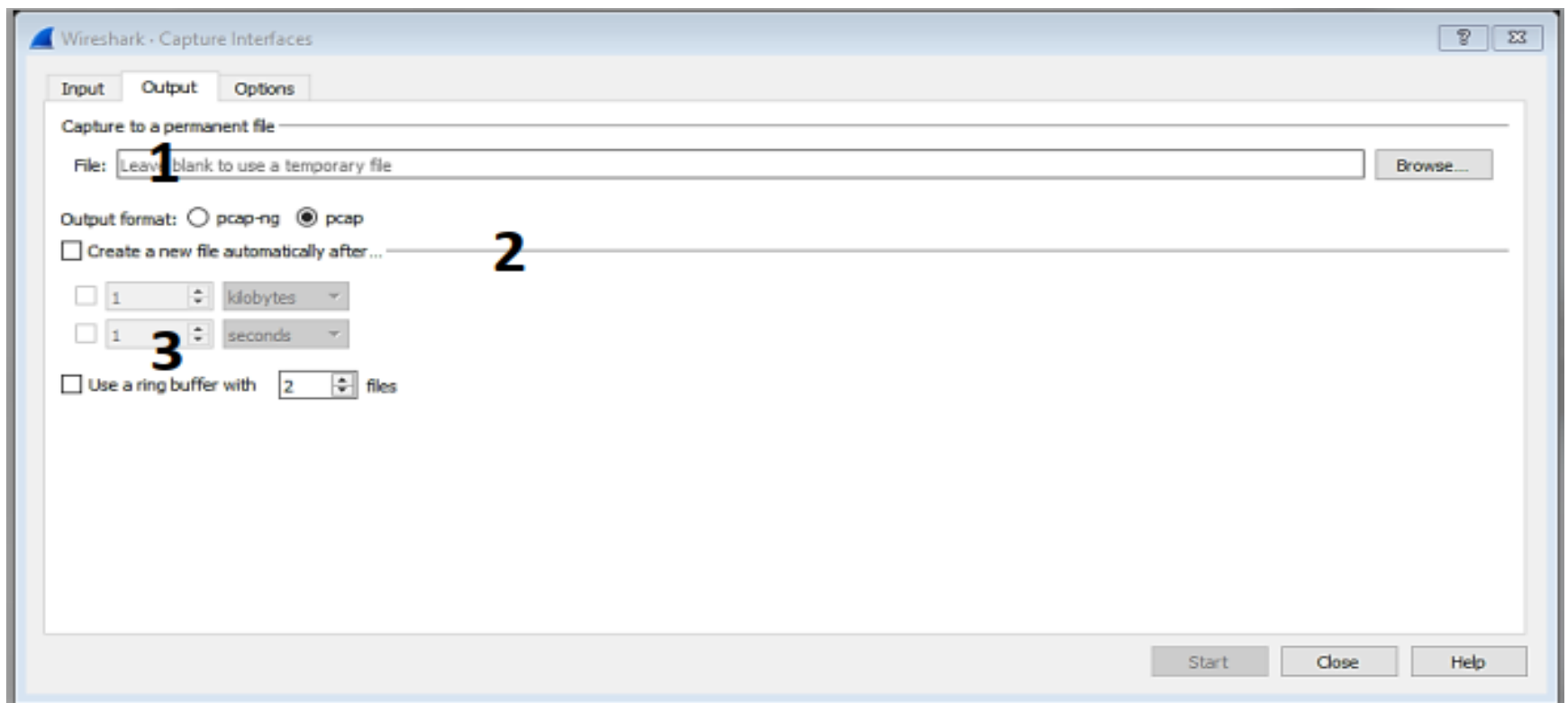
- أول **Tool** معنا هي ال **Wireshark** وهي عبارة عن **Network Sniffer** أو **Protocol Analyzer** ... بتحدد لها ال **Network Interface** اللى هتشتغل عليه عندك عال **PC** وبتعمل **Capture** لل **Traffic** اللى هيعدي من خلاله ... تعالى نشوف شكل ال **Wireshark** بعد أما تنزلها عندك ... ودي ال **Interface** بتعتها وفيها كروت الشبكة اللى عندك .



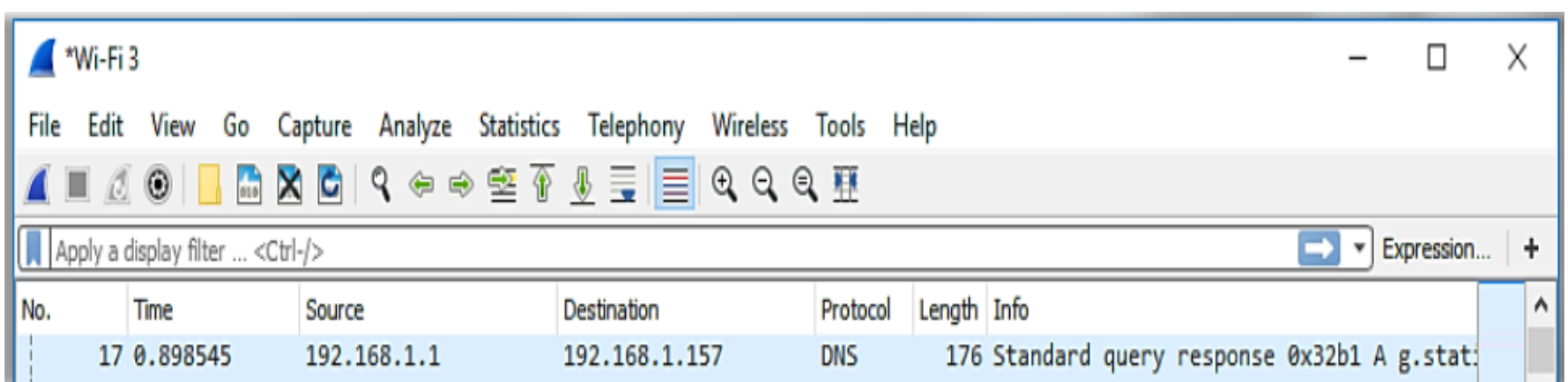
- بعد كدا ممكن ت **Manage** ال **Capture Filter** وتحدد اللى انت عاوزه واللى انت عاوز تعمله **Capture** .



- وتقدر تدخل من ال **Capture** لل **Options** وتختار من ال **Menu** **bar** ال **New Window** وتحدد بالضبط ايه اللي انت عاوز تعمله **Capture** عال **Network Interface** بتاعك زي كدا .



- فهتلاقي تلقائي ال **Wireshark** عملتك ال **Capture** لل **Traffic** زي دا تماما فيه ال **Time** اللي اتعمل فيه ال **Capture** بال **Source** **IP** وال **Destination IP** وال **Protocol** المستخدم وال **Length** الخاص بال **Packet** وكمات معلومات عن ال **Packet** اللي اتعملها **Capture** زي كدا .



- عندنا **Plugin** جوا ال **Wireshark** اسمها ال **Dump cap** ودي بتساعدنا اننا نشغل ال **Wireshark** ك **Command Line Tool** ... وال **Wireshark** نفسه بينزل **Command line Tool** وبعد كدا انت لو عاوز تشغله ك **Command line Tool** زي موضحنا عن طريق ال **Plugin** اللى هي **Dump cap** وعندك **Plugins** تانيه كتير مفيده زي ال **T-Shark** وغيرهم هتفيدك وهتسهل عليك وانت بتستخدم ال **Wireshark** فأبحث عليهم واعرف استخدامهم .

- عندنا ال **Tool** التانيه وهي ال **TCP Dump** اللى هنستخدمها فال **Packet Analysis** اللى هنعملها **Capture** ... وال **Tool** دي موجوده تلقائي فال **Linux System** وهي **Command Line** مش **GUI** زي ال **Wireshark** وهي أسرع أكيد من ال **Wireshark** وبتعمل نفس الشغل لكن الفرق دي **GUI** ودي **Command Line** .

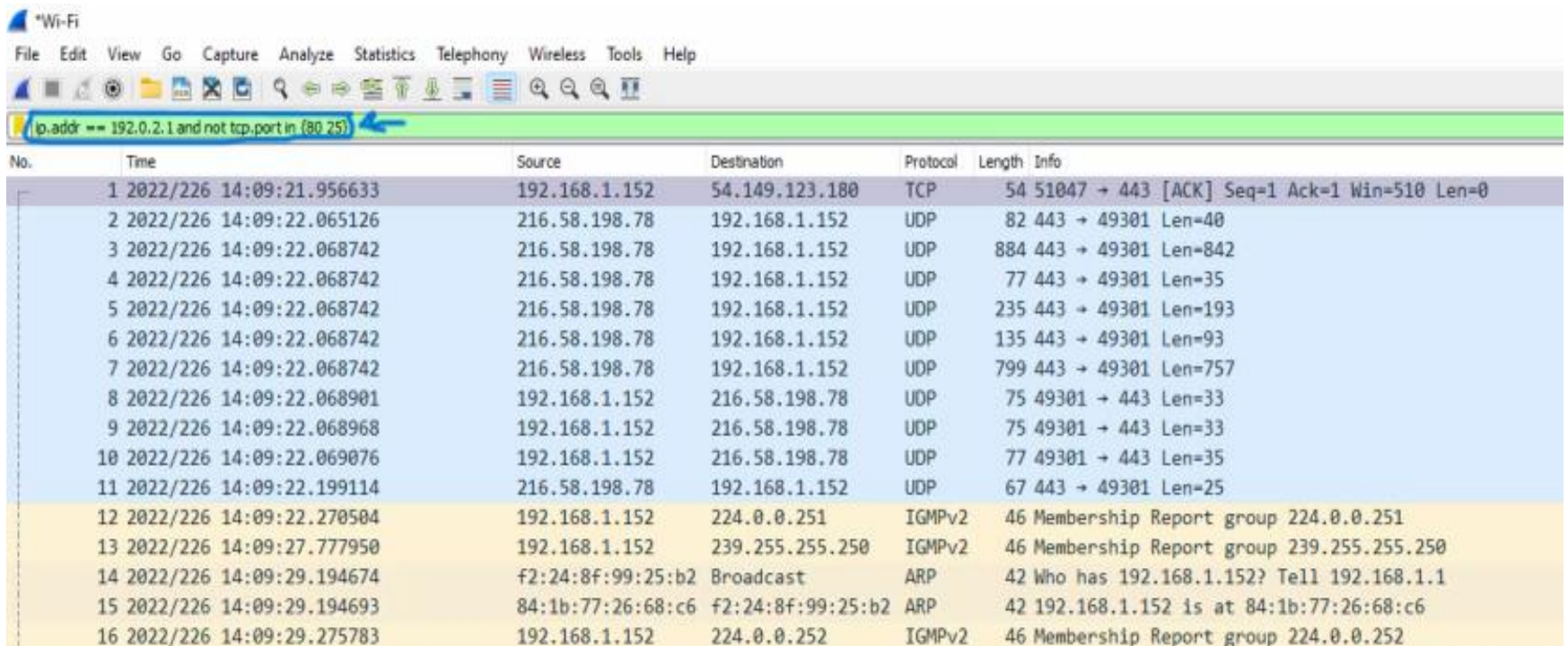
```
tcpdump [options] [filter expression]
```

```
sudo tcpdump -i eth0
```

```
stduser@els:~$ sudo tcpdump -i eth0
[sudo] password for stduser:
tcpdump: verbose output suppressed, use -v or -vv for full protocol decode
listening on eth0, link-type EN10MB (Ethernet), capture size 262144 bytes
09:18:02.133182 IP 192.168.102.1.17500 > 192.168.102.255.17500: UDP, length 200
09:18:02.854919 IP 192.168.102.147.56976 > 192.168.102.2.domain: 53964+ PTR? 255.102.168.192.in-addr.arpa. (46)
09:18:02.864964 IP 192.168.102.2.domain > 192.168.102.147.56976: 53964 NXDomain 0/0/0 (46)
09:18:02.865051 IP 192.168.102.147.59910 > 192.168.102.2.domain: 12279+ PTR? 1.102.168.192.in-addr.arpa. (44)
09:18:02.875296 IP 192.168.102.2.domain > 192.168.102.147.59910: 12279 NXDomain 0/0/0 (44)
09:18:03.848147 IP 192.168.102.147.48873 > 192.168.102.2.domain: 27140+ PTR? 2.102.168.192.in-addr.arpa. (44)
09:18:03.858098 IP 192.168.102.2.domain > 192.168.102.147.48873: 27140 NXDomain 0/0/0 (44)
09:18:03.858183 IP 192.168.102.147.50818 > 192.168.102.2.domain: 50563+ PTR? 147.102.168.192.in-addr.arpa. (46)
09:18:03.867137 IP 192.168.102.2.domain > 192.168.102.147.50818: 50563 NXDomain 0/0/0 (46)
```

- ال **Wireshark** وال **TCP Dump** بيكون جواهم لغه ال **Filtering language** واللى بنسميها ال **Berkley Packet Language** واختصارها ال **BPF** ... ودي اللغه اللى بتساعدنا اننا نعمل **Filter** لل **Output** اللى بيطلعنا من ال **Wire Shark** أو ال **TCP Dump** .

- وطبعا ال **Filters** دي مبتحفظهاش ولكن بتكون موجوده فال **Sheet Sheet** الخاص بال **Wireshark Filters** تقد تنزله من ال **Internet** وتأخذ منه **Copy Paste** وال **BPF Language** بتكتب فال **Wireshark** بالطريقه دي .



The screenshot shows the Wireshark interface with a packet capture filter applied: `ip.addr == 192.0.2.1 and not tcp.port in (80,25)`. The packet list shows 16 packets. Packets 1-11 are TCP and UDP traffic from 192.168.1.152 to 54.149.123.180 and 216.58.198.78. Packets 12-16 are IGMPv2, ARP, and ICMP traffic related to the 224.0.0.251 group and the 192.168.1.152 host.

No.	Time	Source	Destination	Protocol	Length	Info
1	2022/226 14:09:21.956633	192.168.1.152	54.149.123.180	TCP	54	51047 → 443 [ACK] Seq=1 Ack=1 Win=510 Len=0
2	2022/226 14:09:22.065126	216.58.198.78	192.168.1.152	UDP	82	443 → 49301 Len=40
3	2022/226 14:09:22.068742	216.58.198.78	192.168.1.152	UDP	884	443 → 49301 Len=842
4	2022/226 14:09:22.068742	216.58.198.78	192.168.1.152	UDP	77	443 → 49301 Len=35
5	2022/226 14:09:22.068742	216.58.198.78	192.168.1.152	UDP	235	443 → 49301 Len=193
6	2022/226 14:09:22.068742	216.58.198.78	192.168.1.152	UDP	135	443 → 49301 Len=93
7	2022/226 14:09:22.068742	216.58.198.78	192.168.1.152	UDP	799	443 → 49301 Len=757
8	2022/226 14:09:22.068901	192.168.1.152	216.58.198.78	UDP	75	49301 → 443 Len=33
9	2022/226 14:09:22.068968	192.168.1.152	216.58.198.78	UDP	75	49301 → 443 Len=33
10	2022/226 14:09:22.069076	192.168.1.152	216.58.198.78	UDP	77	49301 → 443 Len=35
11	2022/226 14:09:22.199114	216.58.198.78	192.168.1.152	UDP	67	443 → 49301 Len=25
12	2022/226 14:09:22.270504	192.168.1.152	224.0.0.251	IGMPv2	46	Membership Report group 224.0.0.251
13	2022/226 14:09:27.777950	192.168.1.152	239.255.255.250	IGMPv2	46	Membership Report group 239.255.255.250
14	2022/226 14:09:29.194674	f2:24:8f:99:25:b2	Broadcast	ARP	42	Who has 192.168.1.152? Tell 192.168.1.1
15	2022/226 14:09:29.194693	84:1b:77:26:68:c6	f2:24:8f:99:25:b2	ARP	42	192.168.1.152 is at 84:1b:77:26:68:c6
16	2022/226 14:09:29.275783	192.168.1.152	224.0.0.252	IGMPv2	46	Membership Report group 224.0.0.252

- وبكدا نكون أنهينا الحديث عن ال **Topics** اللى محتاجه تتوضح فال **Module** دا وبدايه من ال **Module** القادم هنبء نعمل **Deep Dive** فال **Packet** وهنعملها ال **Analysis** وهنعرف ازاي نعمل **Hunt** لل **Suspicious Traffic** وازاي نميزه عن ال **Normal Traffic** واحنا هنا اتكلمنا عن اساسيات ال **Packet Analysis** وكمان ال **Tools** اللى هنستخدمها فال **Analysis** وال **TCP / IP Model** وعملية ال **Encapsulation** ازاي يتم لل **Packet** وال **Devices** الموجوده عندنا فال **Network** زي ال **Routers** وال **Switches** وغيره من المواضيع الهامه اللى يهمنى تفهمها عشان مبني عليها ال **Section** بتاع ال **Network Hunting** واللى جي كله ... فلو عاوز شرح تفصيلي للكلام دا اللى شرحناه فال **Module** دا هتلاقية فكورس زي ال **eCIR** بالتفصيل أرجعله لو محتاج **Details** .