Recap Guide 2: Recording is here

Timestamps

00:00 Learning Basics: Scripts & Networks

14:52 "OSI, IP, MAC Basics Explained"

33:46 Linux Password Hunt

49:58 TCP vs UDP: Segments & Reliability

56:53 Prioritize Windows for IT Jobs

01:04:46 ARP Cache Poisoning Explained

01:19:43 ARP Cache Spoofing Explained

01:31:54 Network Troubleshooting Basics

01:39:43 Server vs Client Licensing Costs

01:52:40 Router Makes Cross-Network Delivery

01:55:44 Networking and Password Cracking Basics

Recommended Resources

- Read Network Student Guide
- Download Ports and Protocol Cheatsheet
- Practice Labs: OverTheWire
- Practice Labs: Hack The Box Academy
- Learn more about Networking "Professor Messer network types
- Practice creating a network Cisco Packet Tracer
- Learn Linux: TOP 60 Linux Commands

Other Recordings

1. 🔗 WATCH THE RECORDING HERE - Intro to Networking (Sunday 5th October 2025)
2. Lab 1 - Wednesday 8th October)

Recap of The OSI Model

- Open Systems Interconnection (OSI) model. It is a framework describing how data travels through a network in 7 layers.
- Understanding it makes troubleshooting, network design and security analysis easier.
- The Layers: (from top to bottom)
    - Application (L7)
    - Presentation (L6)
    - Session (L5)
    - Transport (L4)
    - Network (L3)
    - Data Link (L2)
    - Physical (L1)
- TCP/IP Model: A simplified (4-layer) real-world equivalent.
- Focus especially on Layers 2 (Data Link), 3 (Network), 4 (Transport), and 7 (Application).

Network Types and Topologies

- Types tell you how big and wide the network is.
- Topologies tell you how things are connected inside it.

Network Types (These describe how wide a network's reach is:)

- **PAN (Personal Area Network):** The smallest range connects personal devices (like Bluetooth between your phone and earbuds).

- **LAN (Local Area Network):**Covers a small area like a house, school, or office. Uses cables or Wi-Fi to connect computers and printers.

- **MAN (Metropolitan Area Network):** Larger than a LAN. It connects multiple LANs across a city (like a university with several campuses).

- **WAN (Wide Area Network):** Connects LANs across countries or continents. The internet is the biggest example.

- **SAN (Storage Area Network):** Used in data centers to connect storage devices (e.g., servers and databases).

Network Topologies (These describe **how devices are arranged and connected** in a network:)

- **Bus Topology:** All devices share a single communication line. Cheap but one fault can stop the whole network.

- **Star Topology:**Every device connects to a central hub or switch. Common in offices; if the hub fails, the network goes down.

- **Ring Topology:** Devices connect in a circle. Data travels one direction. Fast but breaks if one connection fails.

- **Mesh Topology:** Every device connects to many others. Very reliable, but expensive and complex to set up.

- **Tree (Hybrid) Topology:** Combines star and bus. Good for large organizations.

Terminology to remember

- **IP Address:** A device's unique identifier on a network (e.g., 192.168.1.5).
- **Subnet Mask:** Divides IP address into network & host parts (e.g., 255.255.255.0).
- **Default Gateway:** Your router's address for outbound traffic.
- **MAC Address:** Hardware identifier ("physical address") unique to each device's network card.
- **ARP Cache:** Local memory mapping IPs to MAC addresses.

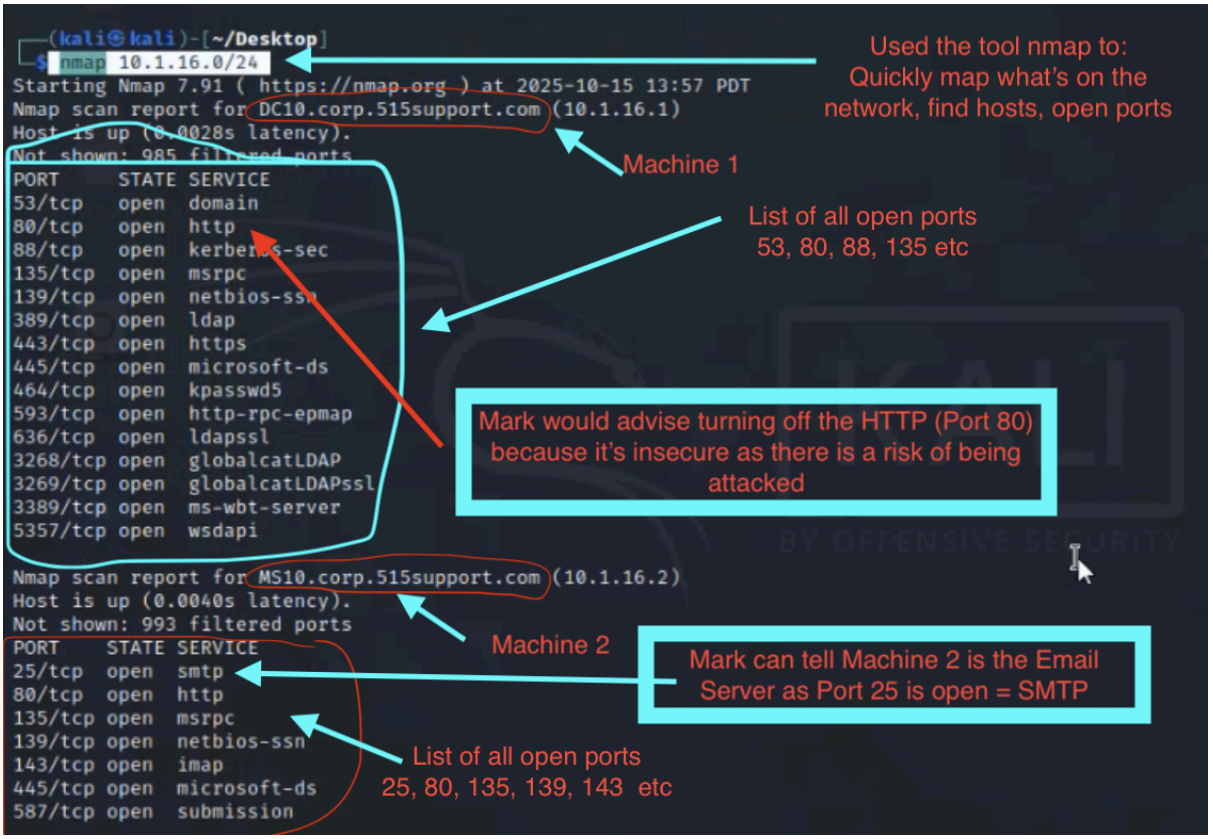
Ports & Protocols

- Port Numbers are logical 'doors' for communication. Each protocol has a default port. Used to distinguish services running on the same device.

- Key Ports to Memorise:
    - HTTP - 80
    - HTTPS - 443
    - FTP - 21
    - SSH - 22
    - SMTP (email) - 25
    - DNS - 53
    - DHCP - 67/68

- Network Scanner:

- Tool: `nmap`
- Find hosts, open ports/services on a network.
- `nmap 10.1.16.0/24`



Marks' Best Practices for Open Ports

The best practice is to only keep the ones open that you actually use, and lock the rest.

- **Keep open:**
  - Only the ports tied to services you actively need (e.g. 80/443 for a web server, 22 for SSH if you manage it remotely).
  - Make sure those services are updated, monitored, and protected (firewall rules, authentication, etc).

- **Close or block:**
  - Everything else that is unused, legacy or testing ports.
  - Especially ports used by vulnerable or unnecessary services (like old file-sharing or remote-desktop protocols).

- **Restrict where possible:**
  - If you need something open (say SSH on 22), restrict *who* can reach it by allowing specific IPs only.
  - Use firewalls or security groups to enforce that.

- **Regularly review:**
    - Run scans (like with **Nmap**) to see what's actually open.
    - Disable or close ports that aren't meant to be public.

General Q&A from the session

Do we need to take an exam in networking?

No, not for this course. You'll need the knowledge to pass Security+, but not a dedicated networking exam like Network+. Security+ is the main certification, and networking knowledge is assumed as foundational for IT.

What should I study if I want to learn programming?

You don't need deep programming skills. Some shell scripting (Linux bash, PowerShell) for automation is useful. If you want, learn Python, but it's not required for the main cyber security track.

What's the difference between the Security+ exam and Network+?

Security+ is more widely required, especially in the U.S. Government. Network+ is a deeper networking qualification, but less popular for entry-level cyber security jobs. Security+ assumes some baseline networking knowledge.

What do I actually need to know about networks to get by in cyber security?

- OSI model (and TCP/IP model)
- Basic switching and routing concepts
- Setting up Packet Tracer labs
- IP addressing (with subnet masks)
- MAC addresses
- Network types
- Core network services (DNS, DHCP, HTTP, FTP, SMTP) and their port numbers, and whether they use TCP or UDP
- Network attacks (DDoS, SQL injection, man-in-the-middle, etc.)
- Basics of wireless networking

Which book should we use for networking?

Use the Network+ Student Guide. Seven to eight chapters are enough for a solid baseline.

Can you explain ARP cache and why it matters? ARP cache is the memory area in each device mapping IP addresses to MAC addresses, used by switches to send frames internally. You can view and flush your ARP cache on your computer, and it's vital for internal network communications and understanding attacks like ARP poisoning.

What happens if a device has never communicated with another device before? It sends an ARP request (broadcast) to all devices on the network asking "who has this IP address?" Only the correct device responds with its MAC address (ARP reply), which then gets cached for future communication.

What is a broadcast, unicast, and multicast?

- **Unicast:** One-to-one communication
- **Broadcast:** One-to-all (every device receives the message)
- **Multicast:** One-to-many (to a group, not all devices)

How can I check my own IP address and MAC address?

- On Windows: Use `ipconfig` and `ipconfig /all` commands in the command prompt.On Linux: Use `ifconfig` or `ip a` in the terminal.

What is the structure of encapsulation in networking? Each layer adds a header to the data. This is called encapsulation. When the data is received, it's stripped layer by layer (de-encapsulation).

What practical labs should we focus on? Focus on using Packet Tracer for switching and routing, understanding IP addressing, using nmap for port scanning, DHCP/DNS labs after week two, network scanner labs, remote access (internal & external), analysing on-path/man-in-the-middle attacks, and Syslog/log management.

Why do cyber security professionals need to understand networks? Because you need to know what you're securing including how networks work, where attacks may occur, and how systems communicate. Most security certifications and jobs assume solid networking knowledge.

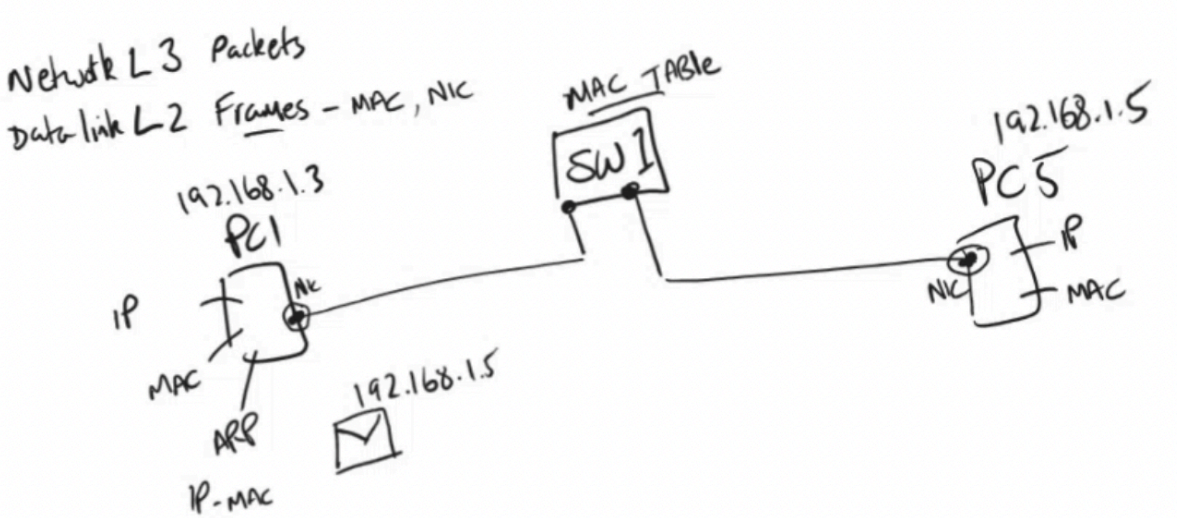
What about wireless networking, how much should we know? Just the basics: access points, wireless controllers, how 802.11 works, differences from wired networking, and awareness of wireless attacks (evil twin, rogue AP).

Which operating systems should we learn for cyber security? Both Windows (most common in business) and Linux (often used for attacks, less vulnerable by default).

How is Microsoft Windows different from Linux in a security/networking context? Windows is easier to break into, so knowing Windows operating systems for security is crucial. Linux is the attacker platform in many hands-on cyber labs.

Analogy from today's session - House Party

To make networking and cybersecurity a bit easier to get your head around, this is an analogy. I know this stuff helps me understand better.



Every guest (your devices) has two types of "ID": a *name tag* (**IP address**) and a *unique badge* (**MAC address**). The OSI model is the different stages each message goes through to get from one guest to another, kind of like passing notes across the room.

When a guest (**Computer 1**) wants to send a secret message (**data**) to another guest (**Computer 5**), they need to know both the name and badge of the receiver. If they don't know the badge, they yell out: "Who's got the badge for Computer 5?" (That's the **ARP request,** a broadcast across the party.) Computer 5 shouts back: "That's me!" and now Computer 1 can send their note straight over.

The **switches** are like the party helpers, making sure notes go directly to the right guest inside the house, and not lost on the way. **Routers** are the doormen, sending messages outside the house, maybe to guests in a neighbouring building.

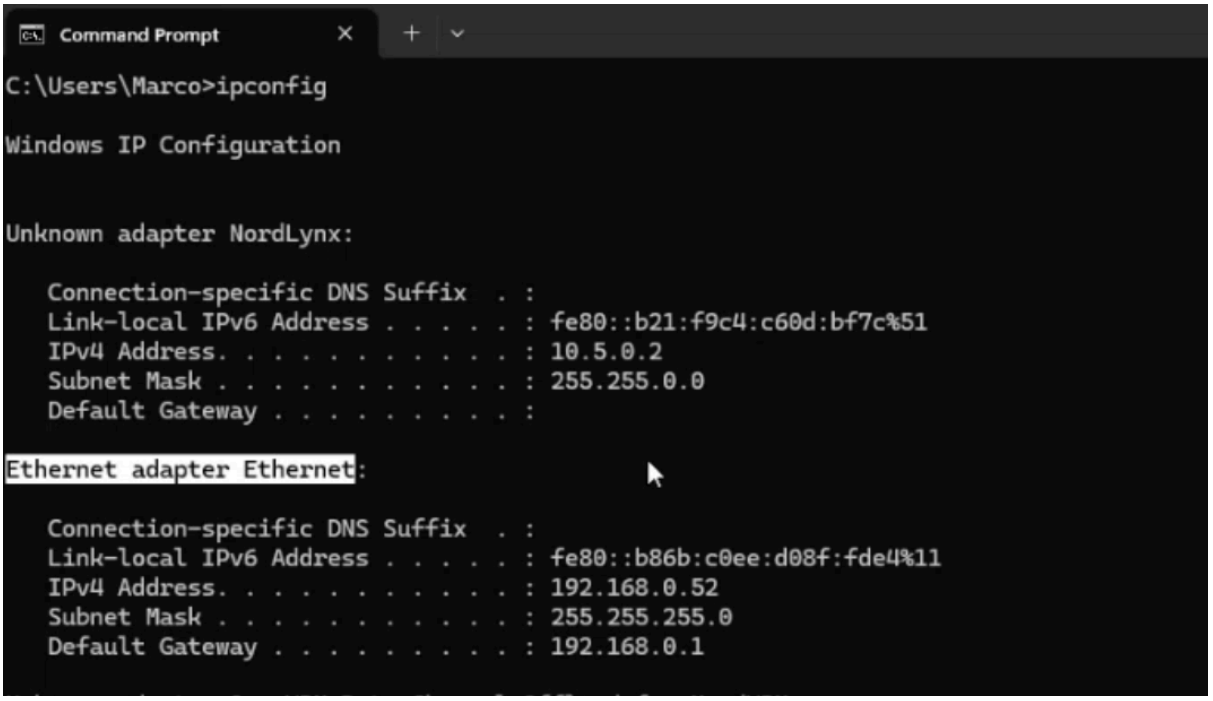
Services (like the snack table or the DJ) are hosted at special spots (**Ports**) in the house. So if you want snacks (**HTTP**), you go to the Snack Room (**Port 80**). For music (**FTP**), you hit the DJ booth (**Port 21**). Some services require a ticket for guaranteed entry (**TCP**), while others are casual drop-ins where it's okay if you miss a beat (**UDP**).

Security is about keeping party crashers at bay. Sometimes people try sneaky moves like pretending to have someone else's badge. That's why knowing **ARP tables** and which badge belongs to who is key to stopping troublemakers!

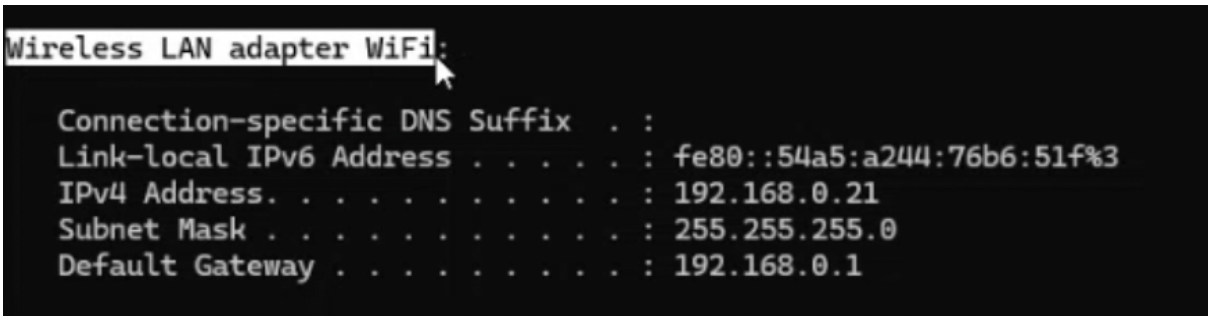
To get good at party planning (**networking)**, you don't need to memorise every rule or every possible snack. Focus on knowing how messages travel, who's who, and the main party spots and you'll keep the network dancing and secure.

Demos of an ARP Cache - ARP Request & Reply, Finding IP Address, MAC addresses

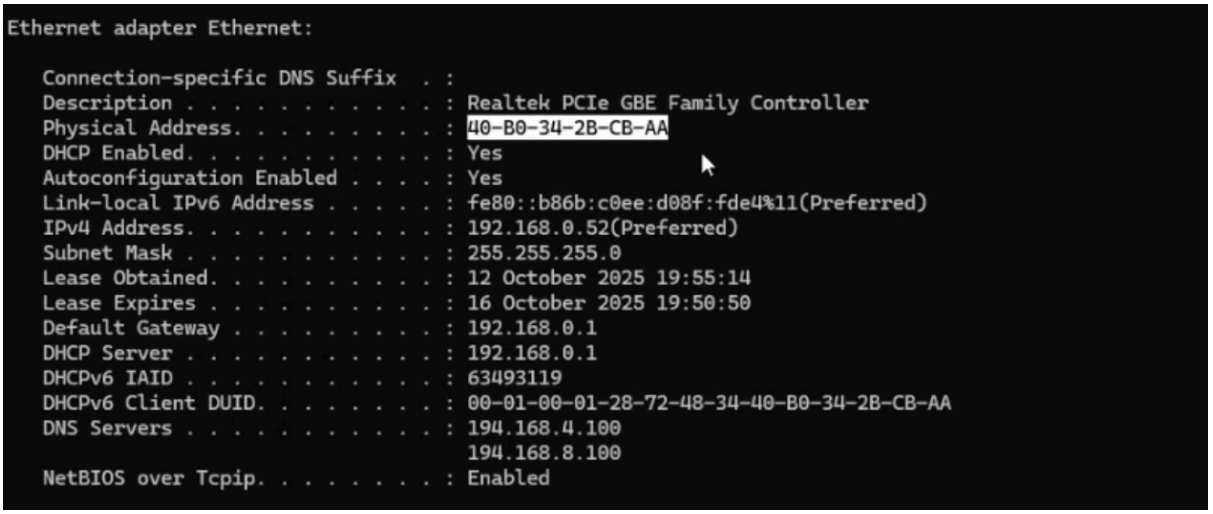
Mark used **ipconfig** to find his Windows IP address labelled "Ethernet adapter"



He scrolled down to find his Wireless Local A Network adapter IPv4 address



He then used another command (ipconfig /all) to find his MAC address also identified as a Physical Address



Now he’s uses the command arp -a, to do an ARP request. The -a stands for all.

An ARP request (Address Resolution Protocol request) is a shout out on your computer to figure out who owns a certain IP address.

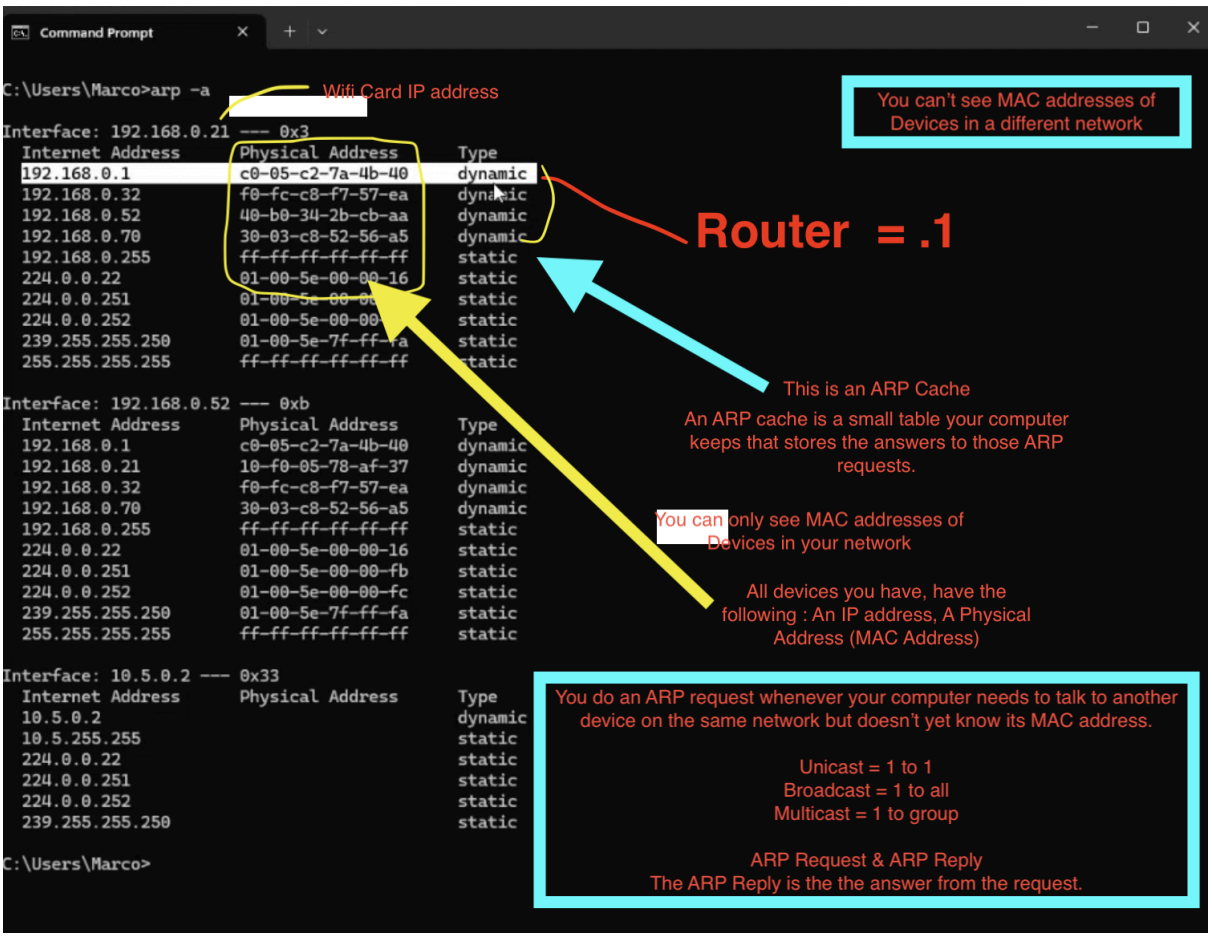
Here’s how it works:

- Every device on a network has an IP address and a MAC address
- When your computer wants to send data to another device but only knows its IP, it broadcasts an ARP request:
  → “Who has IP 192.168.1.10? Tell 192.168.1.5.”

- The device with that IP replies with its MAC address, so now your computer knows exactly where to send packets next time.

In short:
ARP request = asking “Who’s at this IP?”
ARP reply = answering “It’s me, here’s my MAC.”

Then he goes onto explain what everything does and means about the ARP Cache, ARP Request, ARP Reply



Later, we will learn what **ARP spoofing** (aka ARP poisoning is) It is when a device on a local network lies about who owns an IP address so traffic meant for that IP gets sent to the attacker instead. It’s a common local-network attack used for man-in-the-middle (MITM) eavesdropping or denial-of-service.

Reminder To Self

1. **You Don’t Need to Know Everything (Yet!).**
   You’re not expected to remember everything overnight. You're aiming for a solid baseline in networking to prep you for cybersecurity

2. **The OSI Model is Your Friend.**
   The seven-layer OSI model is the industry standard for understanding how networks tick.

3. **Hands-on Beats Theory Every Time.**
   Tools like **Packet Tracer**, **Kali Linux**, and **nmap** are essential. Labs, labs, labs! The more you play, the more you’ll get it. Don’t hesitate to check out Hack the Box and Over the Wire for fun extra practice.

4. **Know Your Basics: Switches, Routers, IP & MAC Addresses.**
   Switch = Layer 2 (frames, MAC addresses, internal connections).
   Router = Layer 3 (packets, IP addresses, connecting networks).

5. **Security+ is King.**
   Security+ is your golden ticket. Nearly everyone in the field needs it, so getting comfortable with networking is a must for passing the exam and getting that first job.

6. **Fun Fact from the Session:**
   Did you know that the **bad guy hacker** is often an expert in not just networks, but operating systems and even a bit of programming? That’s why your learning path is “baseline network knowledge + some operating systems + a dash of scripting = security hero!”

Keep up the great work team! You’ve got this!

All the best, Chris Cownden (fellow JAGUAR cohort team player)