FASE1: Costruzione dell’ambiente di base: rete NAT, Domain Controller e servizi di rete

La prima parte del progetto è servita per costruire da zero la base del laboratorio, cioè la rete virtuale e il primo server: il Domain Controller.  
L’idea era creare una piccola infrastruttura aziendale dove poter simulare servizi reali come Active Directory, DNS e DHCP, ma dentro un ambiente controllato e isolato, senza rischiare di toccare la rete fisica del PC.

Per cominciare ho configurato una rete privata NAT in VMware, che ho chiamato LabNAT.  
L’indirizzo scelto per questa rete è 192.168.149.0/24, con gateway 192.168.149.2.  
Il NAT permette alle macchine virtuali di uscire su Internet attraverso il computer principale, ma allo stesso tempo le tiene isolate: in pratica si comportano come se fossero in un piccolo ufficio collegato a un router locale.  
Il DHCP di VMware è stato disattivato perché voglio che a gestire gli indirizzi IP sia il Domain Controller, proprio come accade in una rete aziendale vera.

Una volta pronta la rete, ho creato la macchina virtuale DC01, basata su Windows Server 2019 Standard (Desktop Experience).  
Ho assegnato alla macchina 2 CPU, 4 GB di RAM e un disco virtuale da 60 GB, impostato su controller LSI Logic SAS, che è quello più stabile con i sistemi Microsoft.  
La scheda di rete è stata collegata alla LabNAT e ho scelto il firmware BIOS tradizionale, che in questo tipo di ambienti è più semplice da gestire.

Dopo l’installazione del sistema operativo, ho impostato un indirizzo IP statico, perché un Domain Controller non deve mai cambiare indirizzo.  
I parametri di rete sono:

* IP: 192.168.149.10
* Subnet mask: 255.255.255.0
* Gateway: 192.168.149.2
* DNS primario: 127.0.0.1
* DNS secondario: 8.8.8.8

Il DNS primario punta a sé stesso, perché il server dovrà gestire internamente le risoluzioni del dominio.  
Il secondario (8.8.8.8) serve solo come backup per le richieste esterne.

Volendo, la configurazione si può anche fare da PowerShell, così da tenerne traccia:

***New-NetIPAddress -InterfaceAlias "Ethernet0" `***

***-IPAddress 192.168.149.10 -PrefixLength 24 -DefaultGateway 192.168.149.2***

***Set-DnsClientServerAddress -InterfaceAlias "Ethernet0" `***

***-ServerAddresses 127.0.0.1,8.8.8.8***

A questo punto ho installato i ruoli principali che serviranno al laboratorio:  
Active Directory Domain Services, DNS Server e DHCP Server.  
Si possono aggiungere dal Server Manager, ma anche da PowerShell, che in questo caso è più veloce:

***Install-WindowsFeature AD-Domain-Services, DNS, DHCP -IncludeManagementTools***

Dopo l’installazione, il passo successivo è promuovere il server a Domain Controller e creare il dominio.  
Il dominio che ho scelto si chiama lab.local, un nome fittizio usato solo per reti interne.  
La procedura guidata chiede anche di impostare una password DSRM per il ripristino del database AD, che va conservata in modo sicuro.

Con il riavvio, il server entra ufficialmente nel dominio lab.local.  
A questo punto il DNS integrato in Active Directory crea automaticamente le zone e i record necessari.  
Per permettere anche la navigazione su Internet, ho configurato i forwarder DNS verso i server pubblici di Google e Cloudflare:

***Add-DnsServerForwarder -IPAddress 8.8.8.8,1.1.1.1***

Il DNS ora è autorevole per lab.local, ma inoltra tutte le richieste esterne ai resolver pubblici.  
Per controllare che tutto funzioni ho fatto qualche test:

***ping 8.8.8.8***

***Resolve-DnsName lab.local***

***Resolve-DnsName dc01.lab.local***

Se il ping e la risoluzione rispondono, significa che Internet e DNS sono configurati correttamente.

Con il DNS a posto, mi sono concentrato sul servizio DHCP, che ho voluto gestire direttamente da DC01.  
Ho creato un nuovo scope chiamato “LabNAT”, con intervallo di indirizzi da 192.168.149.100 a 192.168.149.200.  
Il gateway e il DNS forniti ai client sono rispettivamente 192.168.149.2 e 192.168.149.10, con dominio di ricerca lab.local.  
Gli indirizzi al di sotto di 100 li tengo riservati ai server o ai dispositivi statici.  
Ecco il modo più rapido per configurarlo da PowerShell:

***Add-DhcpServerv4Scope -Name "LabNAT" `***

***-StartRange 192.168.149.100 -EndRange 192.168.149.200 `***

***-SubnetMask 255.255.255.0***

***Set-DhcpServerv4OptionValue -Router 192.168.149.2***

***Set-DhcpServerv4OptionValue -DnsServer 192.168.149.10 -DnsDomain "lab.local"***

***Add-DhcpServerInDC -DnsName "dc01.lab.local" -IpAddress 192.168.149.10***

Questo script crea lo scope, imposta le opzioni di rete e autorizza il server DHCP all’interno di Active Directory, senza passare dalla GUI.  
Il server inizia così a distribuire indirizzi ai client della rete LabNAT.

Per chiudere la prima fase ho verificato che tutto fosse operativo.  
Il comando “***Resolve-DnsName lab.local*”** deve rispondere con l’indirizzo 192.168.149.10, mentre ping 8.8.8.8 conferma che Internet è raggiungibile.  
Infine, controllare lo stato del servizio DHCP:

***Get-Service DHCPServer***

Se il servizio è in stato “Running”, significa che il laboratorio ha finalmente la sua base completa:  
una rete interna isolata, un Domain Controller pienamente funzionante, DNS autorevole con inoltro esterno e DHCP pronto a fornire configurazioni ai client.

Questa è la parte più importante del progetto, perché da qui in poi tutto dipenderà da questo punto di partenza.  
Avere una rete ordinata, un DNS interno e un dominio configurato correttamente è quello che permette di aggiungere altri sistemi — come un server Linux o un client Windows — senza dover rimettere mano alla base ogni volta.

FASE 2: Installazione e integrazione di Ubuntu01 nel dominio

Dopo aver configurato il Domain Controller, la seconda parte del progetto è servita per introdurre un server Linux all’interno della rete, così da poter sperimentare un ambiente misto Windows–Linux e capire come interagiscono tra loro sotto un dominio Active Directory.

La macchina virtuale Ubuntu01 è stata creata in VMware Workstation partendo dall’immagine di Ubuntu Server 22.04 LTS.  
Ho assegnato due CPU, quattro gigabyte di RAM e un disco da 30 GB, collegato a un controller LSI Logic SAS, che Ubuntu riconosce senza problemi.  
La scheda di rete è stata connessa alla rete LabNAT, la stessa del Domain Controller, per mantenere la comunicazione interna diretta.

Durante l’installazione ho impostato la rete manualmente con un indirizzo statico, così da evitare dipendenze dal servizio DHCP e garantire che il server mantenga sempre lo stesso IP.  
I valori scelti sono:  
indirizzo 192.168.149.20, gateway 192.168.149.2, DNS principale 192.168.149.10 (il Domain Controller) e secondario 8.8.8.8.  
Il dominio di ricerca è lab.local, che è lo stesso dominio Active Directory creato in precedenza.  
Questa configurazione permette a Ubuntu di risolvere i nomi interni gestiti dal DC01 e di uscire su Internet quando serve.

Il nome del server è stato impostato su ubuntu01, e l’utente amministrativo locale scelto per la gestione del sistema è ubuntuadmin, con la password definita durante l’installazione.  
Al termine, il sistema si è avviato regolarmente e ho verificato la connettività con alcuni test base:

***ping -c 4 192.168.149.10***

***ping -c 4 8.8.8.8***

***ping -c 4 lab.local***

Tutti i test hanno confermato che la comunicazione interna e l’accesso a Internet funzionavano come previsto.

Prima di unire il server al dominio, ho aggiornato il sistema e installato tutti i pacchetti necessari per permettere l’autenticazione tramite Active Directory:

***sudo apt update && sudo apt upgrade -y***

***sudo apt install -y realmd sssd sssd-tools sssd-ad libnss-sss libpam-sss adcli samba-common-bin packagekit oddjob oddjob-mkhomedir***

Questi strumenti permettono a Ubuntu di riconoscere un dominio e di gestire le credenziali in modo centralizzato.  
realmd si occupa di individuare e collegare il sistema al dominio, mentre sssd mantiene la sincronizzazione delle identità.  
Gli altri pacchetti gestiscono la parte LDAP, Kerberos e la creazione automatica delle home directory per gli utenti del dominio.

Una volta terminata la preparazione, ho verificato che Ubuntu fosse in grado di vedere il dominio lab.local:

***realm discover lab.local***

La risposta è stata corretta: il sistema ha trovato il dominio e il relativo controller.  
A questo punto ho eseguito l’unione vera e propria con:

***sudo realm join -U Administrator lab.local***

Dopo aver inserito la password dell’Administrator del dominio, Ubuntu si è unito senza errori.  
Per conferma ho controllato la configurazione:

***realm list***

Il dominio risultava elencato con tutti i parametri corretti, a indicare che il join era andato a buon fine.  
Per essere sicuro che la comunicazione con Active Directory fosse effettiva, ho eseguito un ultimo controllo con l’utente del dominio:

***id Administrator@lab.local***

Il sistema ha restituito l’elenco dei gruppi e degli identificativi associati, segno che l’autenticazione con Active Directory era attiva e funzionante.

Con questa configurazione, Ubuntu01 è ora membro del dominio lab.local e può utilizzare le credenziali centralizzate gestite dal Domain Controller.  
Il DNS e la rete sono sincronizzati, e i due sistemi — Windows Server e Ubuntu Server — condividono lo stesso spazio di autenticazione.  
Questa fase segna il momento in cui il laboratorio smette di essere solo un ambiente di test isolato e diventa una piccola infrastruttura reale, capace di far convivere più piattaforme nello stesso dominio.

La struttura che abbiamo costruito fino a questo punto non è soltanto un insieme di macchine virtuali collegate tra loro.  
È, in miniatura, la copia del modo in cui le reti aziendali funzionano nella realtà.  
Ogni componente che abbiamo configurato — la rete, il dominio, i DNS, i ruoli server — rappresenta un pezzo preciso di un’infrastruttura IT vera.  
Capire come questi elementi interagiscono è il primo passo per ragionare come un sistemista e non solo come un utente.

L’ambiente LabNAT ci insegna il concetto di segmentazione di rete: in azienda le reti non sono mai “aperte”, ma separate per motivi di sicurezza e gestione.  
Il NAT di VMware riproduce il comportamento di un gateway o di un firewall aziendale, che permette ai dispositivi interni di uscire su Internet, ma non il contrario.  
Quando configuriamo un gateway, un DNS o un server DHCP, stiamo in realtà imparando come vengono gestite le comunicazioni e gli indirizzi in un’intera LAN.

La creazione di DC01 con Active Directory è il passo più importante per capire la logica delle reti Windows.  
In un contesto aziendale reale, quasi tutto ruota attorno al dominio: le autenticazioni, le policy, le risorse condivise, i permessi e la gestione centralizzata degli utenti.  
Con il Domain Controller impariamo che l’identità non appartiene al singolo computer, ma all’intera rete.  
Configurare DNS e DHCP sullo stesso server serve a capire come questi servizi si integrano per fornire connettività e risoluzione dei nomi in modo automatico e coerente per tutti i dispositivi.

L’aggiunta di Ubuntu01 nel dominio introduce la parte più interessante del laboratorio: l’interoperabilità.  
In azienda non esistono solo server Windows, ma anche sistemi Linux che devono comunicare con l’infrastruttura Microsoft.  
Unire Ubuntu al dominio ci fa toccare con mano come Active Directory può gestire anche dispositivi non Windows, e come i protocolli standard (Kerberos, LDAP, DNS) permettono a sistemi diversi di parlare lo stesso linguaggio.  
Quando un utente del dominio può accedere anche su Ubuntu, stiamo vedendo in pratica cosa significa “Single Sign-On” — un concetto fondamentale per chi lavora in reti miste o in ambienti cloud.

Questo laboratorio serve quindi a collegare la teoria alla realtà:  
ogni volta che esegui un comando o imposti un servizio, stai in realtà riproducendo quello che in azienda è già automatizzato o gestito da interfacce più complesse.  
Capire cosa succede sotto il cofano ti permette, un giorno, di risolvere problemi veri e di ragionare in modo sistemico.

Il nostro Mini Enterprise HomeLab, così com’è ora, è già una base completa per imparare:

* come si costruisce una rete aziendale interna con un dominio Active Directory;
* come DNS e DHCP collaborano per gestire i client e garantire continuità di servizio;
* come un sistema Linux può essere integrato e autenticarsi nello stesso dominio;
* come verificare, documentare e mantenere la coerenza di un’infrastruttura.

Tutto ciò che stiamo facendo serve a formare il modo di pensare del sistemista: pianificare prima di installare, capire perché un servizio deve esistere, verificare il flusso logico della rete e avere sempre sotto controllo cosa dipende da cosa.  
Questo approccio è esattamente quello che in un ambiente di lavoro fa la differenza tra chi “usa i comandi” e chi capisce davvero cosa sta costruendo.

In sintesi, questa prima parte del laboratorio non è un esercizio accademico ma un’esperienza diretta:  
stiamo imparando come funziona un’azienda “dall’interno della rete”, un pezzo alla volta, fino a essere in grado di comprenderne l’intera architettura tecnica.