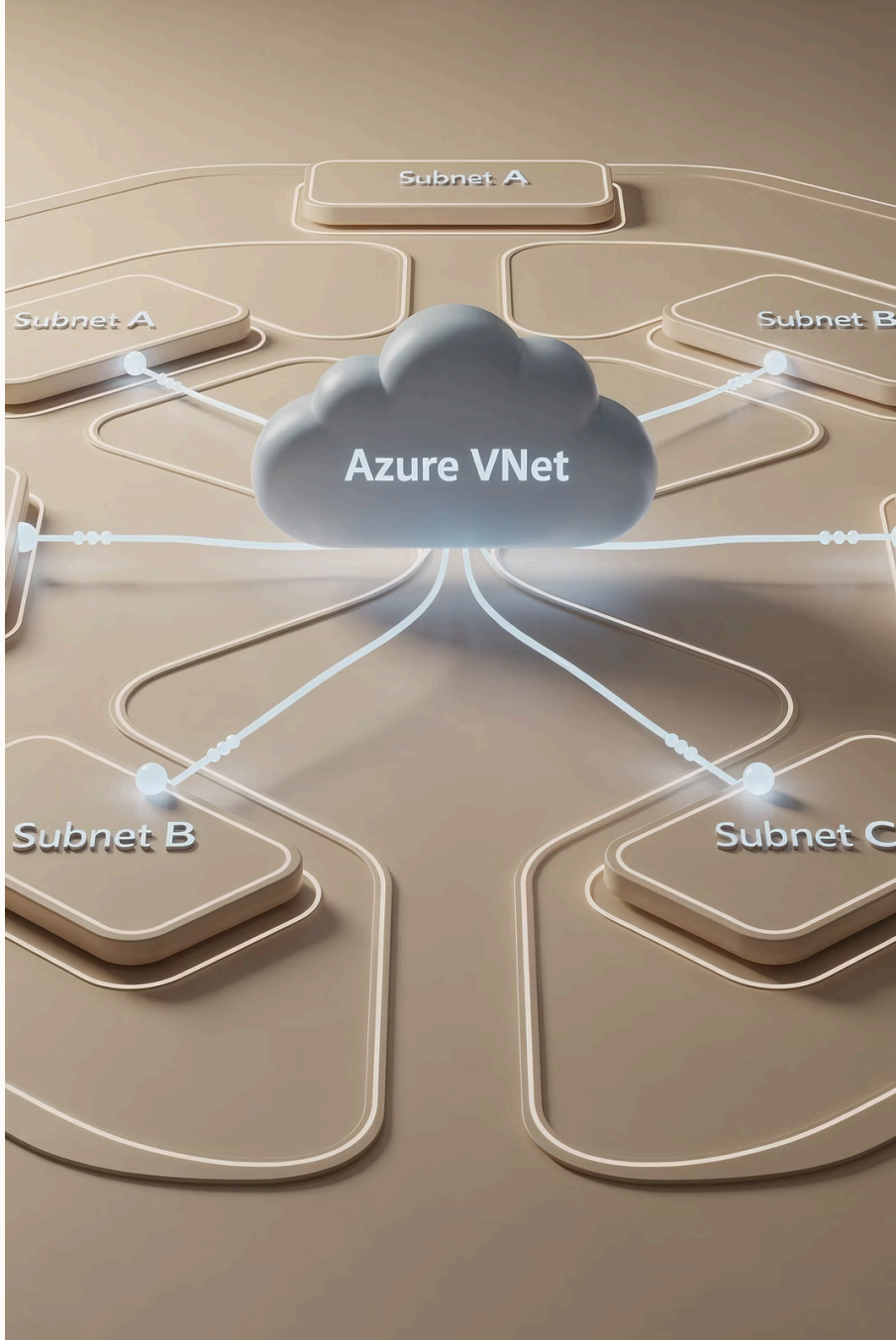
Azure Fundamentals — Sesiunea 4

Rețele Virtuale în Azure

Bun venit la sesiunea dedicată rețelisticii Azure! Astăzi demistificăm VNet, Subnet, IP și NSG — conceptele fundamentale ale oricărei infrastructuri cloud. Fără calcule matematice, fără jargon excesiv. Doar înțelegere practică și intuitivă.

AZURE FUNDAMENTALS | SESIUNEA 4 | NETWORKING



Ce acoperim astăzi

O sesiune structurată în jurul conceptelor esențiale de rețelistică Azure — de la analogia orașului până la construirea primei rețele virtuale.

01

De ce contează rețeaua

Rețeaua este fundația oricărei infrastructuri cloud. Înaintea serverelor, înaintea aplicațiilor.

02

Analogia orașului

Un model mental clar: VNet = oraș, Subnet = cartier, IP = adresă, NSG = portar.

03

IP Public vs. Privat

Diferența dintre adresa vizibilă din internet și adresa internă a rețelei.

04

VNet și Subnets în Azure

Crearea rețelei virtuale și organizarea resurselor în zone cu scopuri clare.

05

Network Security Groups

Regulile de filtru al traficului care protejează fiecare zonă a rețelei.

06

Lab practic + Temă

Construim VNet, Subnets și NSG. Pregătim fundația pentru sesiunea viitoare.

De ce contează rețeaua?

Înainte să construim orice în Azure, hai să răspundem la o întrebare simplă: **Cum ajunge un mesaj WhatsApp de la voi la prietenul vostru?**

Simplificat

Mesajul pleacă de pe telefon prin WiFi sau date mobile → ajunge la serverele WhatsApp în cloud → serverele identifică destinatarul → mesajul este livrat pe telefonul prietenului vostru.

Realitatea din spate

În spatele acestei simplificări există un **întreg sistem de adrese, rute, reguli și infrastructură** care face posibil ca miliarde de mesaje să ajungă la destinatarii corecți în fiecare secundă — fără să se piardă, fără să ajungă la persoanele greșite.

Rețelele sunt **coloana vertebrală** a oricărui sistem digital. Fără rețele, serverele sunt izolate. Fără rețele, aplicațiile nu pot comunica. Fără rețele, datele nu pot călători de la utilizator la server și înapoi.

Rețeaua — Fundația Infrastructurii Cloud

Când construiți infrastructură în Azure, **primul lucru pe care îl proiectați este rețeaua**. Înainte de servere. Înainte de aplicații. Înainte de baze de date.

🔓 Cine poate fi accesat din internet?

Înțelegând rețeaua, înțelegeți de ce un server web poate fi accesat public și altul nu. Aceasta este o decizie arhitecturală deliberată, nu un accident.

🔒 Ce rămâne privat?

O bază de date nu ar trebui să fie niciodată expusă direct pe internet. Rețeaua este mecanismul care implementează această regulă la nivel de infrastructură.

🔗 Comunicare internă sigură

O aplicație web poate vorbi cu un serviciu intern fără ca traficul să iasă vreodată în internet. Rețeaua face posibil acest izolament.

Acestea nu sunt detalii tehnice minore. Sunt **decizii arhitecturale fundamentale** care determină securitatea, performanța și costul sistemelor voastre.

Analogia Orașului — Modelul Nostru Mental

Vom folosi această analogie pe tot parcursul sesiunii. Ea transformă conceptele abstracte de rețelistică în ceva familiar și intuitiv.

🏙️ VNet = Orașul

Spațiul vostru privat în Azure. Granițele lui sunt granițele orașului. Tot ce se întâmplă în interior este izolat de restul lumii, dacă voi decideți asta.

🏘️ Subnet = Cartierul

Subdiviziuni ale orașului cu scopuri clare: cartierul de frontend, cartierul de backend, cartierul de date — fiecare cu regulile lui.

📬 IP = Adresa Poștală

Fiecare resursă din rețea are o adresă IP, exact cum fiecare apartament are o adresă. Fără adresă, nu poți trimite și nu poți primi nimic.

🚨 NSG = Portarul

Verifică fiecare pachet de date care încearcă să intre sau să iasă și decide, pe baza regulilor voastre, dacă îl lasă să treacă sau îl blochează.

IP Public vs. IP Privat

Două tipuri de adrese, două roluri complet diferite. Înțelegerea acestei distincții este esențială pentru orice decizie arhitecturală în Azure.

🌐 IP Public

Adresa vizibilă din afara orașului. Adresa pe care o cunoaște oficiul poștal, pe care o vedeți pe Google Maps, pe care o poate găsi oricine din internet.

- Unică la nivel global — niciun alt dispozitiv nu are același IP public
- Vizibilă și accesibilă din orice punct al internetului
- Dacă serverul vostru are un IP public, oricine știe adresa poate încerca să se conecteze
- Poate fi Static (nu se schimbă) sau Dynamic (se poate schimba la restart)

🏠 IP Privat

Adresa internă, din interiorul casei. Dormitorul este camera unu, baia este camera doi, bucătăria este camera trei. Există și funcționează intern, dar din stradă nimeni nu o știe și nu o poate folosi direct.

- Folosite în rețele interne, nu rutate pe internet
- Intervalele rezervate: 10.x.x.x, 172.16-31.x.x, 192.168.x.x
- Asignate automat de Azure prin DHCP
- Stabile pe durata vieții resursei în Azure

Regula de aur: O bază de date cu IP public este o vulnerabilitate critică. Este una dintre cele mai frecvente greșeli în arhitecturile cloud mai puțin mature.

DNS — Cartea de Telefon a Internetului

Adresele IP sunt funcționale, dar greu de reținut de oameni. **DNS — Domain Name System** — traduce nume ușor de reținut în adresele IP numerice pe care rețeaua le folosește efectiv.

Cum funcționează DNS

Nimeni nu accesează Google tastând `142.250.185.46`. Tastează `google.com` și un server DNS face traducerea automat. DNS este intermediarul invizibil din spatele fiecărei navigări web.

- **Web Apps Azure**: primesc automat un domeniu de forma `numeapp.azurewebsites.net`
- **Mașini virtuale**: pot primi un DNS name de forma `numemașină.westeurope.cloudapp.azure.com`
- **DNS intern VNet**: resursele se găsesc unele pe altele după nume, nu după IP

De ce este important în Azure

Folosiți întotdeauna **DNS names în locul adreselor IP hardcodate** în configurații. Dacă adresa IP se schimbă, DNS name-ul rămâne același și aplicația nu se oprește.

În interiorul unui VNet, Azure oferă un serviciu DNS intern gratuit care permite resurselor să se găsească unele pe altele după nume — serverul web poate apela serviciul de baze de date cu numele lui DNS intern, fără să știe adresa IP exactă.

VNet — Virtual Network în Azure

Un **VNet** este o rețea virtuală izolată în Azure, dedicată exclusiv subscripției voastre. Este spațiul vostru privat de rețea în cloud.

Izolare completă

Resursele din VNet-ul vostru sunt izolate de resursele altor clienți Azure și de internet. Ce se întâmplă în orașul vostru nu se vede în alte orașe.

Spațiu de adrese propriu

La creare, definiți un interval de adrese IP private — **address space**. Toate resursele din VNet primesc adrese din acest interval. Azure Portal oferă valori default valide.

Legat de o regiune

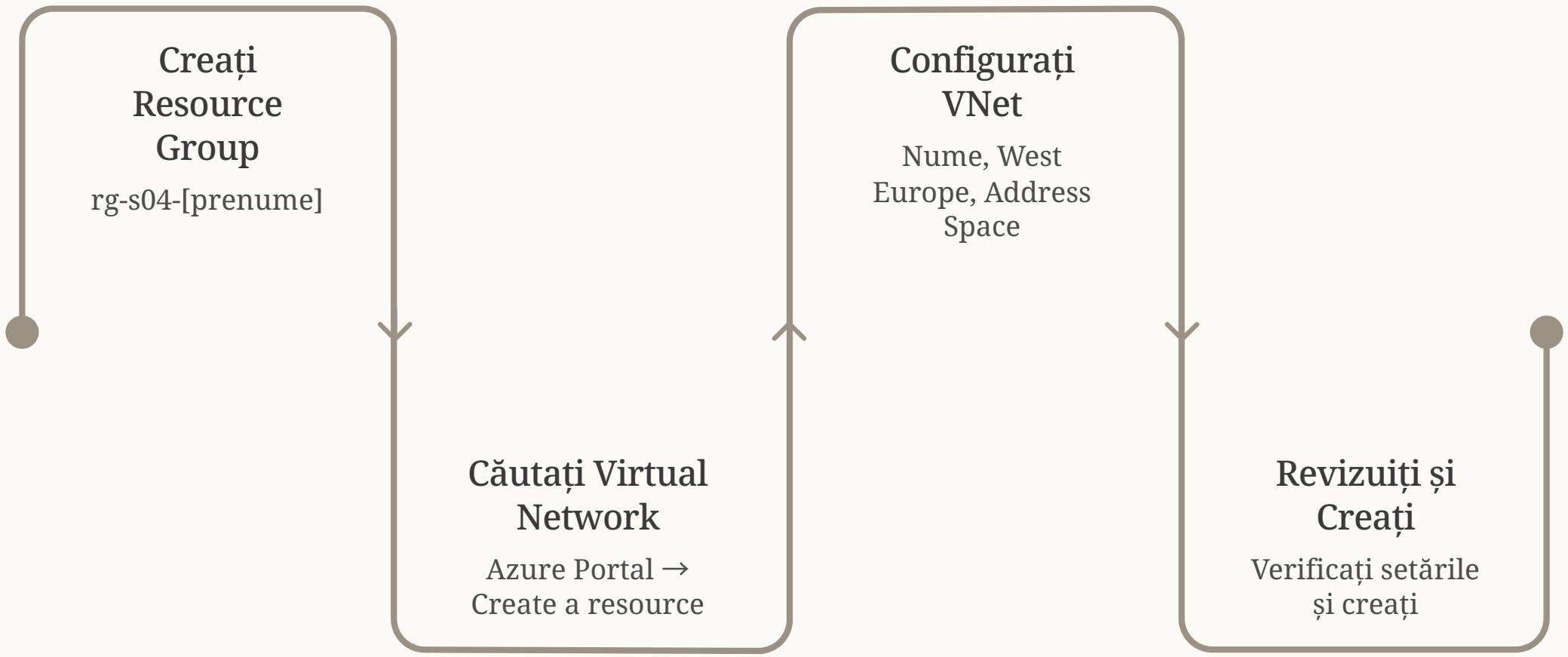
Un VNet există într-o regiune Azure specifică. Când creați un VNet în West Europe, el există fizic în centrele de date din Europa de Vest.

Gratuit

VNet-ul în sine nu costă nimic. Plătiți pentru resursele din el — mașini virtuale, baze de date, IP-uri publice — dar containerul de rețea este gratuit.

Lab 1 — Creăm Primul Nostru VNet

EXERCIȚIU PRACTIC



1

Resource Group

Creați un Resource Group cu numele `rg-s04-[prenumele vostru]`. Acesta va conține toate resursele sesiunii de astăzi.

2

Create Virtual Network

În Azure Portal → **Create a Resource** → căutați **Virtual Network**. La Name: `s04-vnet-lab-[prenume]`. La Region: **West Europe**.

3

Address Space

Lăsați valoarea **default** sugerată de Azure — un interval de adrese private valid, deja completat. Nu modificați nimic altceva.

4

Review + Create

Verificați configurația și dați **Create**. Felicitări — ați creat propriul oraș privat în Azure!

Subnets — Cartierele Orașului

Un VNet mare, cu sute de resurse, fără nicio organizare internă, devine greu de gestionat și greu de securizat. Subnets aduc **ordine, scop și securitate granulară**.

De ce separăm în Subnets?

Gândiți-vă la o aplicație web tipică cu trei componente:

- **Frontend** — serverele web accesibile din internet. Trebuie să primească trafic de la utilizatori.
- **Backend** — serverele de aplicație care procesează logica. Accesibile doar din frontend, nu din internet.
- **Date** — bazele de date. Accesibile exclusiv din backend. Niciodată din internet.

Dacă puneți toate acestea în același subnet, o vulnerabilitate în serverul web permite unui atacator să ajungă direct la baza de date.

Apărarea în Adâncime

Subnets separate cu reguli de trafic diferite implementează **restricțiile la nivel de infrastructură** — cel mai jos nivel posibil, cel mai sigur.

Filozofia: **nu un singur zid înalt, ci mai multe ziduri succesive**, fiecare adăugând rezistență suplimentară.

- Subnet **public** → resurse care servesc utilizatori externi
- Subnet **privat** → logica de aplicație, inaccesibilă din internet
- Subnet de **date** → baze de date, accesibile doar din subnet-uri specifice

Lab 2 — Adăugăm Subnets în VNet

EXERCIȚIU PRACTIC

Acum că avem VNet-ul creat, îl organizăm în cartiere cu scopuri clare.

1

Navigare la VNet

În Azure Portal, găsiți VNet-ul creat anterior. În meniul din stânga, selectați **Subnets**. Dați click pe + **Subnet**.

2

Subnet Frontend

Numiți-l `s04-subnet-frontend`. Lăsați Address Range la valoarea sugerată de Azure. Acesta va găzdui serverele web.

3

Subnet Backend

Adăugați al doilea subnet: `s04-subnet-backend`. Azure gestionează automat împărțirea spațiului de adrese fără suprapuneri.

Notă despre CIDR / Subnetting: Nu intrăm astăzi în calculele matematice ale notațiilor /24 sau /16. Ce contează la nivelul nostru este conceptul: fiecare subnet este o zonă separată cu propriul interval de adrese și cu capacitatea de a fi securizat independent.

NSG — Network Security Group

Un **NSG** este un filtru de trafic. Este portarul la intrarea în cartier sau la ușa clădirii — verifică fiecare pachet de date și decide dacă îl lasă să treacă.

Inbound Rules

Se aplică traficului care **intră** în subnet sau resursă.

- Cineva din internet încearcă să se conecteze → inbound
- O resursă din VNet trimite date → inbound

Default: tot traficul extern este blocat. Traficul intern din VNet este permis.

Outbound Rules

Se aplică traficului care **iese** din subnet sau resursă.

- Serverul accesează un serviciu extern → outbound
- Serverul trimite un răspuns unui client → outbound

Default: tot traficul outbound este permis. Serverele pot iniția conexiuni spre exterior.

Filozofia corectă: Blocați implicit, deschideți explicit. Nu deschideți implicit și încercați să blocați ce nu vreți — este mult mai greu de gestionat și mult mai predispus la greșeli.

Anatomia unei Reguli NSG

Fiecare regulă NSG are **șase câmpuri principale**. Înțelegând aceste câmpuri, puteți construi orice politică de securitate de rețea.

Câmp	Descriere	Exemplu
Source	De unde vine traficul: IP specific, interval de IP-uri, Service Tag Azure, sau Any	`Any` sau `10.0.1.0/24`
Source Port	Portul de pe care pleacă traficul — de obicei Any, ales aleatoriu de sistemul de operare	`*` (Any)
Destination	Unde merge traficul: IP specific, interval, Service Tag, sau Any	`10.0.0.4` sau `Any`
Destination Port	Portul la care bate traficul — câmpul cel mai important în practică	`443` (HTTPS), `22` (SSH), `3389` (RDP)
Protocol	TCP, UDP, sau Any	`TCP`
Action + **Priority**	Allow sau Deny. Priority: 100-4096 — prima regulă care se potrivește câștigă	`Allow`, Priority `1000`

Regulile Default NSG

Azure creează automat reguli default în fiecare NSG nou. Înțelegerea lor este esențială pentru a evita comportamente neașteptate.

Inbound Default Rules

AllowVnetInBound — Priority 65000

Permite traficul din interiorul aceluiași VNet. Orice resursă din VNet poate vorbi cu altă resursă din același VNet.

AllowAzureLoadBalancerInBound — Priority 65001

Azure are nevoie să verifice că resursele sunt sănătoase. Această regulă permite health check-urile interne.

DenyAllInBound — Priority 65500

Blochează tot restul traficului inbound. Dacă nu adăugați reguli explicite de Allow, tot traficul extern este blocat.

Outbound Default Rules

AllowVnetOutBound — Priority 65000

Permite tot traficul outbound între resurse din același VNet.

AllowInternetOutBound — Priority 65001

Serverele din VNet pot iniția conexiuni spre internet — pot descărca actualizări, pot apela API-uri externe.

DenyAllOutBound — Priority 65500

Blochează tot restul traficului outbound care nu se potrivește regulilor de mai sus.

NSG pe Subnet vs. NSG pe NIC

Un NSG poate fi asociat în **două locuri diferite**. Înțelegând diferența, puteți construi politici de securitate precise și eficiente.

NSG pe Subnet

Se aplică **întregului trafic** care intră și iese din acel subnet. Toate resursele din subnet sunt afectate.

Folosiți pentru **reguli generale**: "niciun trafic din internet nu ajunge direct în subnet-ul de backend".

NSG pe NIC

Se aplică specific **interfeței de rețea a unei singure resurse**. Doar traficul acelei resurse este afectat.

Folosiți pentru **reguli specifice**: un server de monitoring are nevoie de acces pe porturi speciale, diferite de restul subnet-ului.

Ambele active simultan

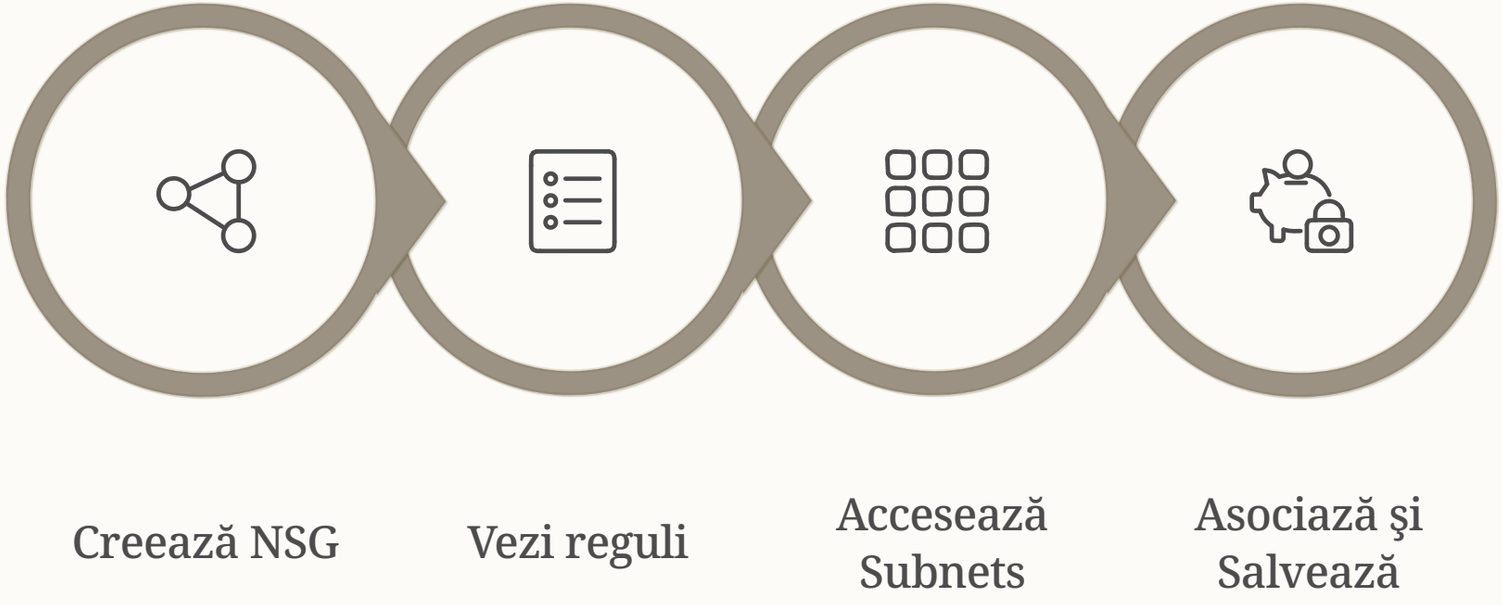
Puteți aplica NSG-uri în ambele locuri simultan. Dacă aveți un NSG pe subnet și un NSG pe NIC-ul unei mașini virtuale din acel subnet, traficul trece prin **ambele filtre**. Trebuie să fie permis de amândouă ca să treacă.

Buna practică

Puneți **regulile generale** pe NSG de subnet. Puneți **regulile specifice** pe NSG de NIC. Mențineți structura simplă, documentată și predictibilă. Evitați regulile contradictorii între cele două niveluri.

Lab 3 — Creăm și Asociem un NSG

EXERCIȚIU PRACTIC



1 Creați NSG-ul

Create a Resource → Network Security Group. Numiți-l `s04-nsg-lab-[prenume]`. Selectați Region: **West Europe** și Resource Group: `rg-s04-[prenume]`. Dați Create.

2 Explorați regulile default

Navigați la NSG-ul creat. Uitați-vă la **Inbound Security Rules** și **Outbound Security Rules**. Observați regulile AllowVnetInBound, DenyAllInBound și prioritățile lor.

3 Asociați NSG-ul cu subnet-ul frontend

VNet-ul vostru → **Subnets** → click pe `s04-subnet-frontend` → la câmpul **Network Security Group**, selectați NSG-ul vostru → **Save**.

Astăzi nu modificăm regulile NSG — le vom configura în sesiunea viitoare, când deployăm primele mașini virtuale. Astăzi important este că NSG-ul există și este asociat subnet-ului.

Călătoria unei Cereri — De la Utilizator la Server

Scenariul: Un utilizator din Spania accesează o aplicație web găzduită pe un server din subnet-ul frontend al VNet-ului vostru din West Europe. Urmăriți fiecare pas.

1. Rezoluția DNS

1

Browser-ul vede un URL. Nu știe IP-ul. Trimite cerere la serverul DNS → primește înapoi adresa IP publică a serverului.

2. Cererea traversează internetul

2

Browser-ul trimite cererea HTTP/HTTPS pe IP-ul public. Cererea ajunge la infrastructure Azure din West Europe.

3. Verificarea NSG

3

NSG-ul verifică cererea: sursă, port, protocol. Dacă există o regulă Allow pentru portul 443 — cererea trece. Altfel — blocată.

4. Procesarea serverului

4

Serverul web din subnet-ul frontend primește cererea pe IP-ul privat. Poate interoga baza de date din backend prin IP privat intern.

5. Răspunsul se întoarce

5

Serverul trimite răspunsul → trece prin NSG outbound (permis implicit) → traversează internetul → ajunge la browser-ul utilizatorului.

Observați: Utilizatorul a interacționat DOAR cu IP-ul public. Nu știe nimic despre subnet-ul backend sau baza de date. Comunicarea internă a folosit IP-uri private și nu a ieșit niciodată în internet.

VNet Peering și Conectivitate Hibridă

VNet-urile sunt izolate implicit — dar există scenarii în care vreți să le conectați. Câteva concepte pe care trebuie să le cunoașteți.

VNet Peering

Permite comunicarea directă între două VNet-uri, cu **latență mică și lățime de bandă mare**. Traficul nu iese niciodată în internet — rămâne pe infrastructure privată Azure. Util pentru: VNet de development ↔️ VNet de producție, sau VNet de servicii partajate ↔️ VNet-uri de aplicații.

VPN Gateway

Creează un **tunel criptat prin internet** între VNet-ul Azure și rețeaua on-premises a companiei (datacenter fizic). Soluție accesibilă pentru conectivitate hibridă, cu performanță moderată.

Azure ExpressRoute

Creează o **conexiune privată dedicată**, fără internet, prin un provider de telecomunicații. Performanță maximă, latență minimă, pentru scenarii enterprise critice. VNet-ul nu este o insulă izolată dacă nu vreți să fie.

Acestea sunt subiecte pentru cursuri avansate de networking Azure. La nivelul sesiunii de astăzi, important este să știți că **aceste opțiuni există** și că ele construiesc pe fundamentul pe care l-ați pus astăzi.

Recap — Întrebări și Răspunsuri

Dacă puteți răspunde clar la aceste cinci întrebări, ați atins obiectivul sesiunii de astăzi.

1

Ce este un VNet?

O rețea virtuală privată și izolată în Azure, dedicată subscripției voastre. Spațiul vostru privat de rețea în cloud, în care plasați resurse care trebuie să comunice între ele, izolat de restul lumii Azure.

2

Ce este un Subnet?

O împărțire logică a unui VNet. Permite organizarea resurselor în zone cu scopuri similare și aplicarea de politici de securitate diferite pentru fiecare zonă. Fiecare subnet primește un interval de adrese IP din spațiul VNet-ului.

3

Ce este un IP privat?

Adresa internă a unei resurse în interiorul unui VNet. Nu este rutată pe internet — nu poate fi accesată din exterior. Folosită exclusiv pentru comunicarea între resurse din interiorul rețelei.

4

Ce este un NSG?

Un set de reguli de securitate care filtrează traficul de rețea. Poate fi asociat unui subnet sau NIC. Conține reguli inbound și outbound. Regulile default blochează tot traficul extern inbound.

5

De ce nu punem totul într-un singur subnet?

Subnets diferite permit politici de securitate diferite. O bază de date și un server web au cerințe de acces complet diferite. Separarea permite securitate granulară și principiul apărării în adâncime.

Tema Sesiunii 4

Două componente practice care vă vor ajuta să verificați dacă ați înțeles cu adevărat conceptele de astăzi — nu doar să le fi auzit.

📐 Componenta 1 — Desen pe hârtie

Desenați o arhitectură de rețea simplă care conține:

- Un singur **VNet**
- Două **Subnets**: frontend și backend
- Un **NSG** asociat fiecărui subnet
- Un singur **IP public**, asociat unui server web fictiv din subnet-ul frontend

Nu trebuie să fie perfect tehnic. Nu trebuie să aibă adrese IP reale. Trebuie să arate că înțelegeți **relațiile dintre componente**. Folosiți cutii și săgeți, scrieți numele fiecărei componente.

✍️ Componenta 2 — Explicați în cuvinte

Scrieți în **cinci propoziții** cum ajunge un utilizator din internet la un server din subnet-ul frontend. Descrieți fiecare pas al călătoriei:

1. Rezoluția DNS
2. IP-ul public
3. Verificarea NSG
4. IP-ul privat al serverului
5. Răspunsul înapoi la utilizator

Nu trebuie să fie perfect tehnic. Trebuie să arate că înțelegeți **fluxul end-to-end**.

De ce aceste teme? Desenatul pe hârtie și explicația în cuvinte proprii sunt cele mai eficiente metode de a verifica dacă ați înțeles cu adevărat un concept. Dacă puteți desena și explica — știți. Dacă nu puteți — aveți un semnal clar unde să reveniți.

Privind Înainte — Ce Urmează

Ce am acoperit astăzi este **fundația fondației**. Pe măsură ce avansați în carieră, veți adăuga straturi deasupra acestui fundament.

Fundația (Astăzi)

VNet, Subnet, IP Public/Privat, NSG. Conceptele de bază fără de care nimic altceva nu are sens în networking Azure.

Sesiunea 5 — Mașini Virtuale

Adăugăm reguli NSG reale și deployăm primele mașini virtuale. Rețeaua de astăzi prinde viață cu resurse reale în interiorul ei.

Nivelul Următor

Application Gateway și Load Balancer pentru distribuirea traficului. Azure Firewall pentru securitate avansată. Private Endpoints pentru acces privat la servicii Azure.

Nivel Avansat

VPN Gateway și ExpressRoute pentru conectivitate hibridă. Azure DNS pentru managementul zonelor DNS. Arhitecturi multi-region cu VNet Peering global.

Întrebările Fundamentale Rămân Aceleași

Indiferent cât de complex devine un sistem, aceste întrebări ghidează orice decizie de arhitectură de rețea cloud. Dacă știți să răspundeți la ele și să implementați răspunsurile în Azure, aveți abilitățile de bază ale unui arhitect de rețea cloud.

> Cine trebuie să vorbească cu cine?

> Ce trafic permitem și ce blocăm?

> Ce este expus public și ce rămâne privat?

> Cum separăm resursele cu cerințe diferite de securitate?

⚠️ **Nu uitați:** Ștergeți Resource Group-ul `rg-s04-[prenume]` la finalul sesiunii practice. Toate resursele create astăzi — VNet-ul, subnet-urile, NSG-ul — vor fi șterse împreună, fără costuri reziduale.