**Day 15**





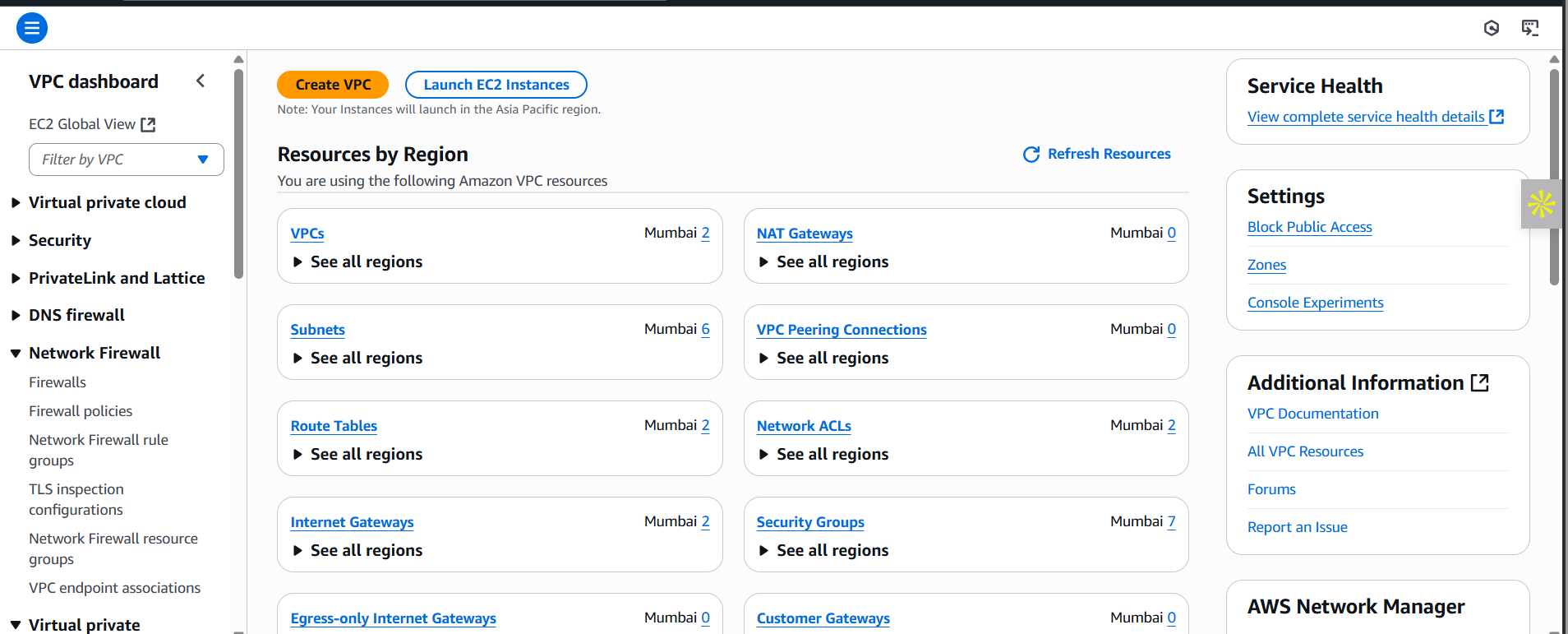
**“CLOUD SECURITY”**

**Securing a VPC using AWS Network Firewall:**

1. Advanced Threat Protection: AWS Network Firewall is a managed, stateful firewall that provides intrusion detection and prevention (IDS/IPS), web filtering, and customizable rule enforcement to block unauthorized domains, IPs, and malicious traffic using Suricata.
2. Scalable and Highly Available: It automatically scales with network traffic and supports centralized logging via CloudWatch, S3, or Kinesis for real-time visibility and analysis of firewall activity.
3. Policy-Driven Management: It uses core components — firewalls, firewall policies, and rule groups — allowing administrators to define fine-grained inspection rules and centrally manage security controls across subnets and AZs.

**Where can we find this Network Firewall option?**

It is under the VPC dashboard, under the subsection named “Network Firewall”.



**Some basic information about this:**

**What is a firewall?**

A firewall is a security system that monitors, filters, and controls incoming and outgoing network traffic based on defined security rules.

**Why do we need a firewall?**

We need firewalls to:

* Block malicious traffic
* Inspect traffic at deeper layers (e.g., application or packet-level)
* Detect intrusions or attacks (e.g., port scanning, malware communication)
* Enforce fine-grained security policies

**Why is it needed if we already have VPC, NACL, and CloudWatch?**

* VPC offers isolated networking but not deep packet inspection.
* NACL provides stateless filtering (IP/port-based) at the subnet level.
* Security Groups are stateful, but limited to allow rules.
* CloudWatch is for monitoring and alerting, not filtering.
* Firewall adds deep inspection, intrusion detection/prevention, domain-based filtering, and scalable threat defense — beyond basic access control.

**What makes AWS Network Firewall different?**

* Uses Suricata for stateful inspection and signature-based threat detection
* Supports domain-based blocking, custom rule sets, and protocol-level filtering
* Integrated with CloudWatch, S3, Kinesis, and Firewall Manager
* Automatically scales with traffic and supports centralized control

**What does AWS Network Firewall inspect?**

* Source/Destination IPs
* Ports
* Protocols (e.g., TCP, UDP, ICMP)
* Domains (e.g., block bad.com)
* Packet content (deep inspection)

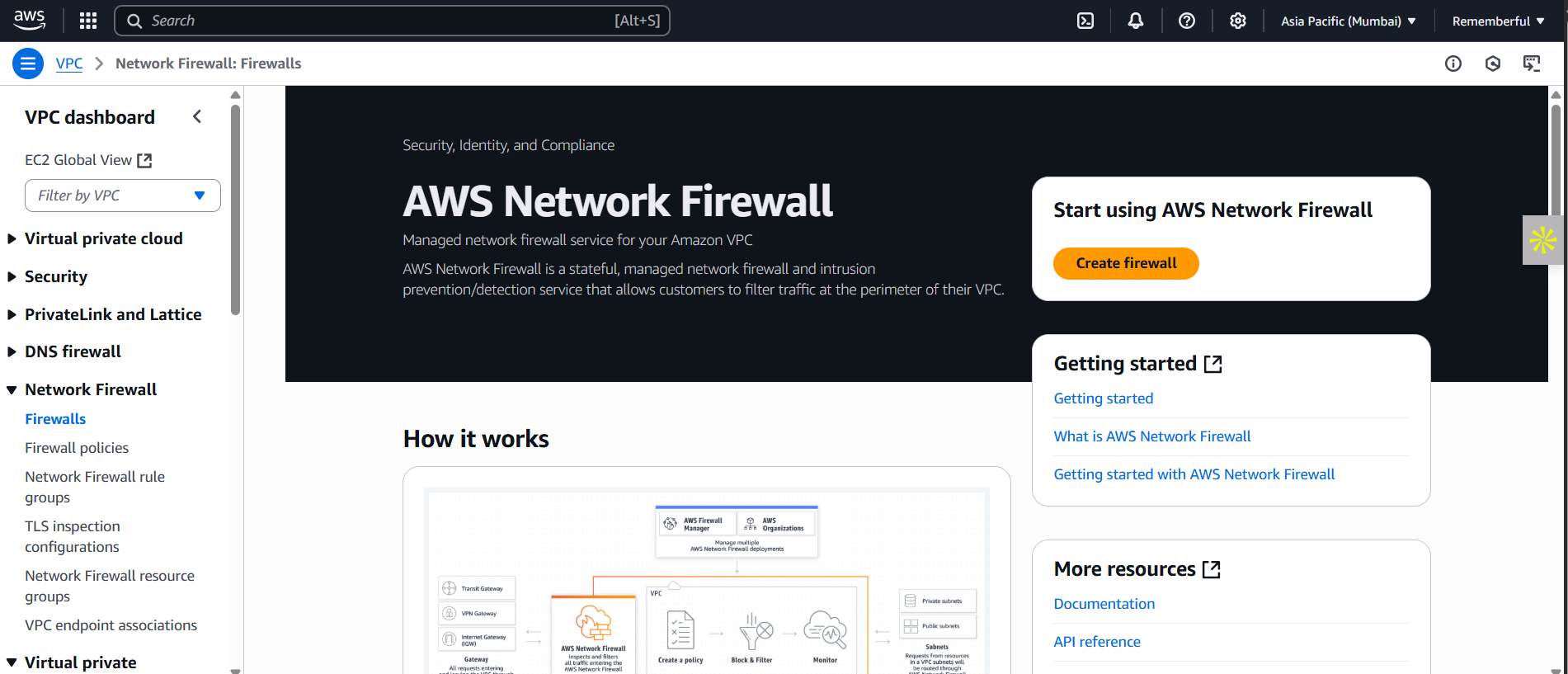
**Who should use AWS Network Firewall?**

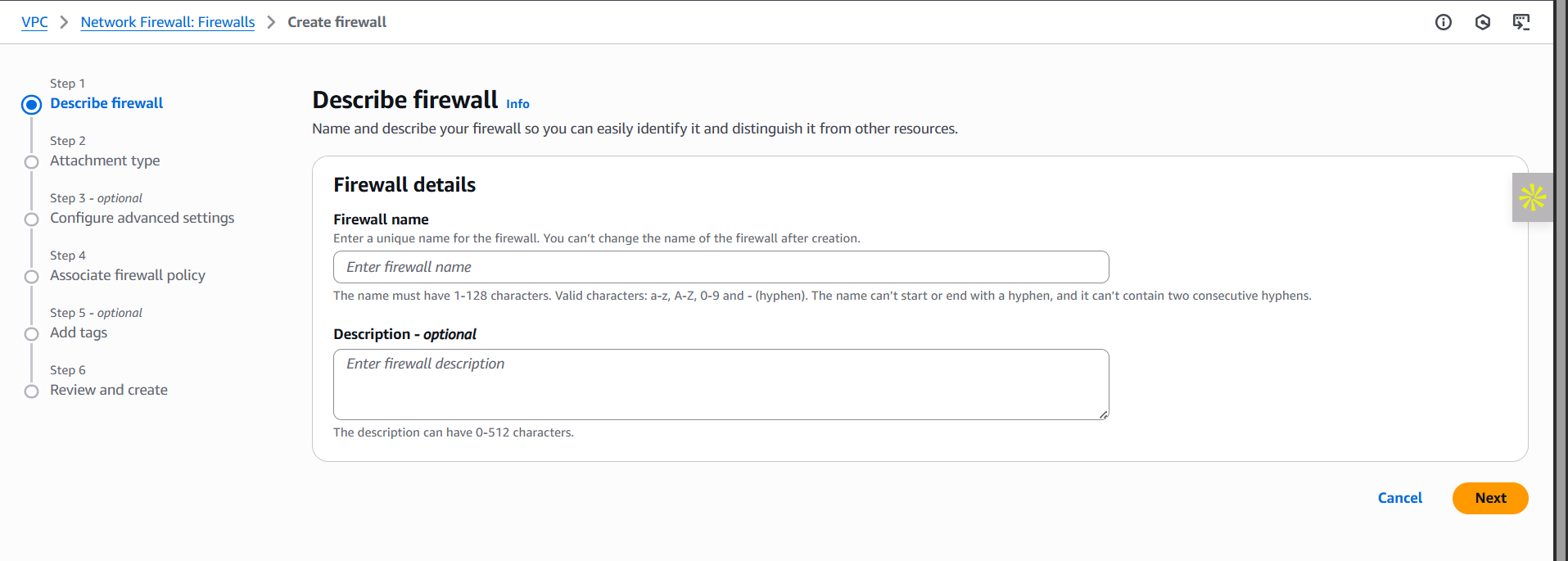
* Enterprises needing **intrusion detection**
* Organizations needing **centralized firewall policies**
* Workloads under **compliance** (e.g., PCI-DSS)
* Security-first deployments requiring **custom traffic rules**

**Creating the Network Firewall:**

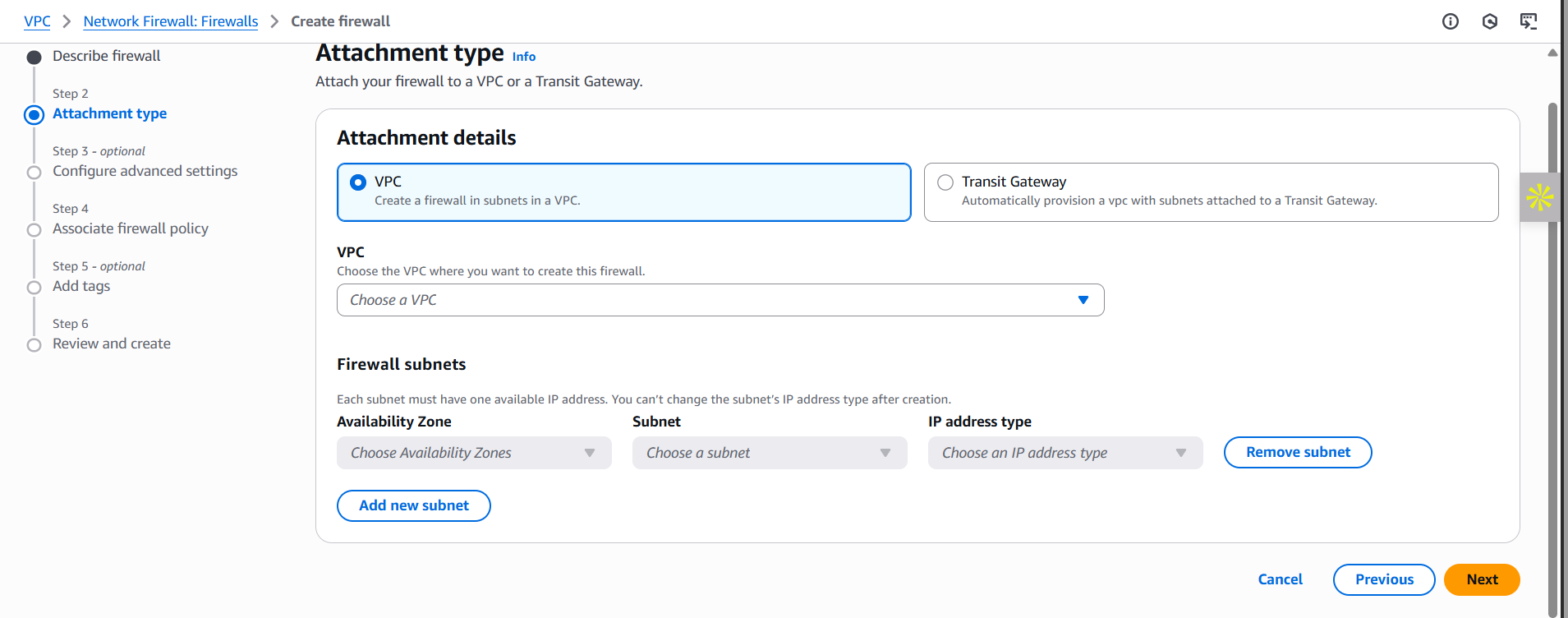
Steps:

Open the “Firewall” tab: following screen will appear.

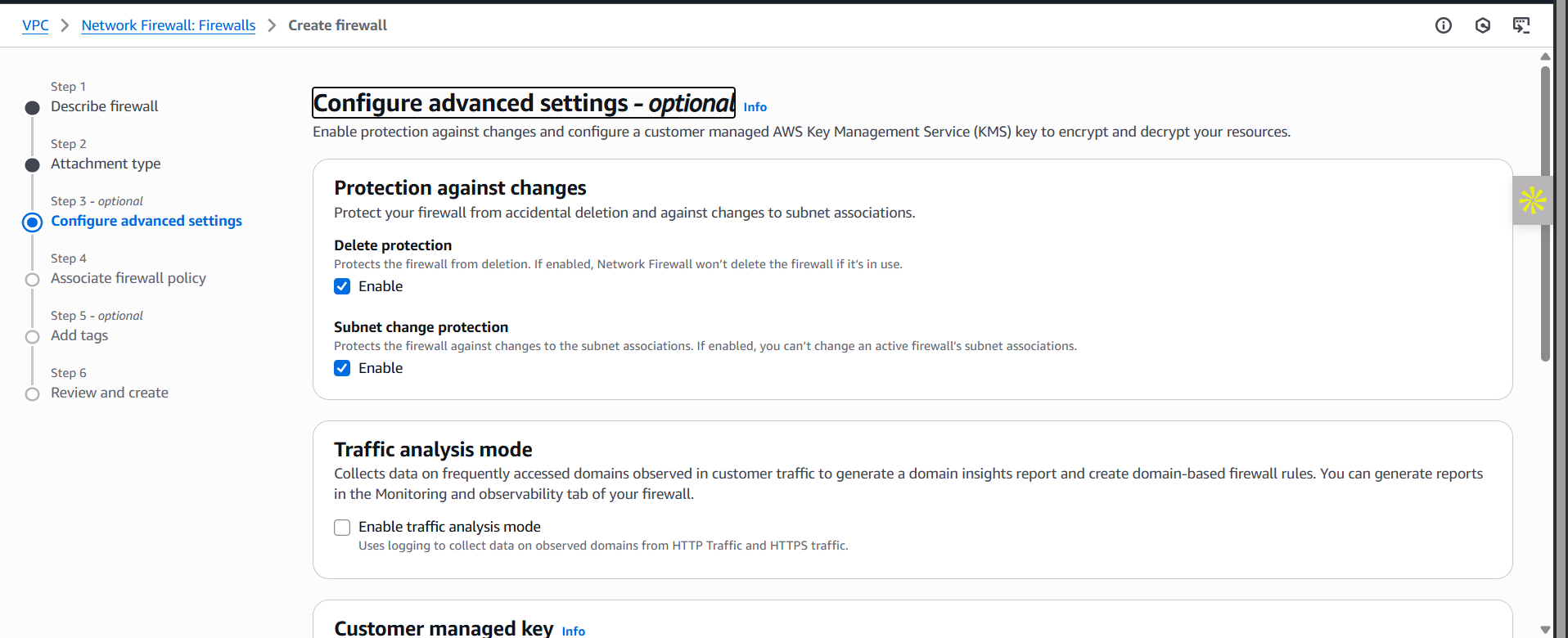


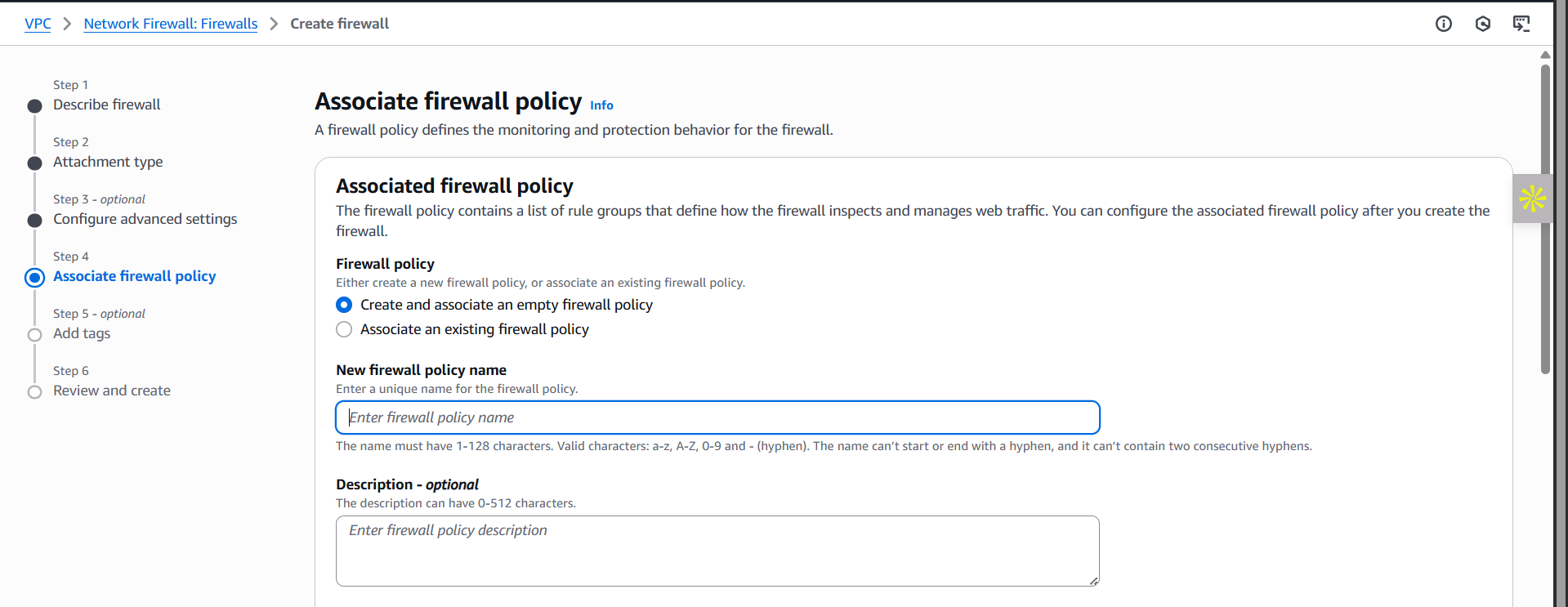
Click on the “create firewall” orange coloured button: following screen will appear. 

Fill the first appeared “Describe Firewall” section, and move to the “Next”: following screen with the following options will appear.

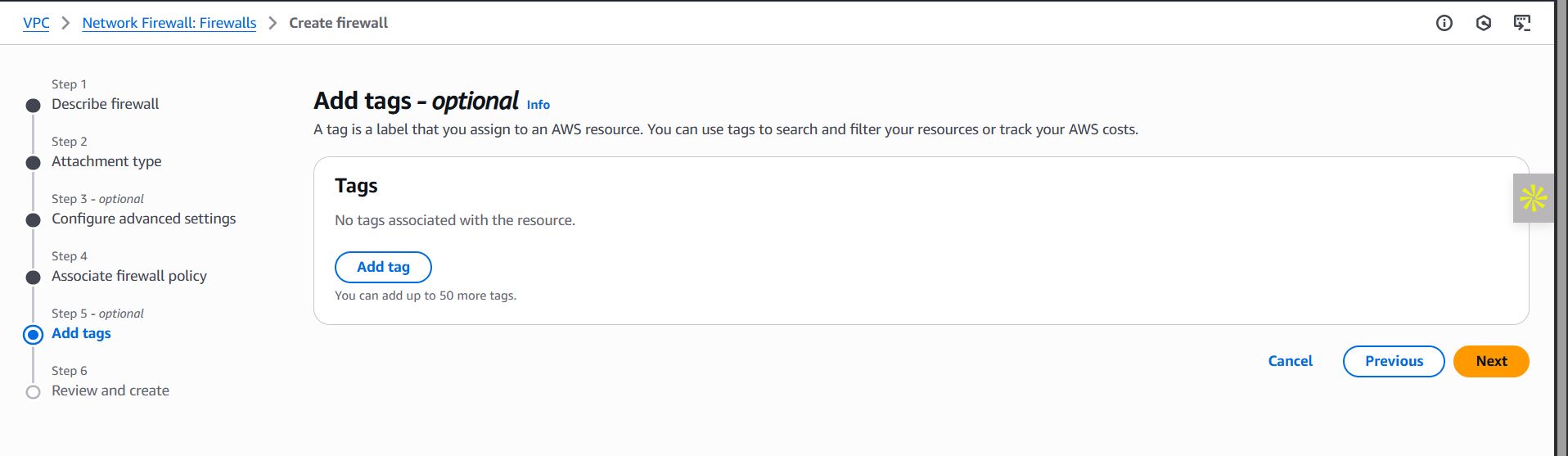


Fill it, and click on the “Next” button: following screen will appear. Fill and move to the next.

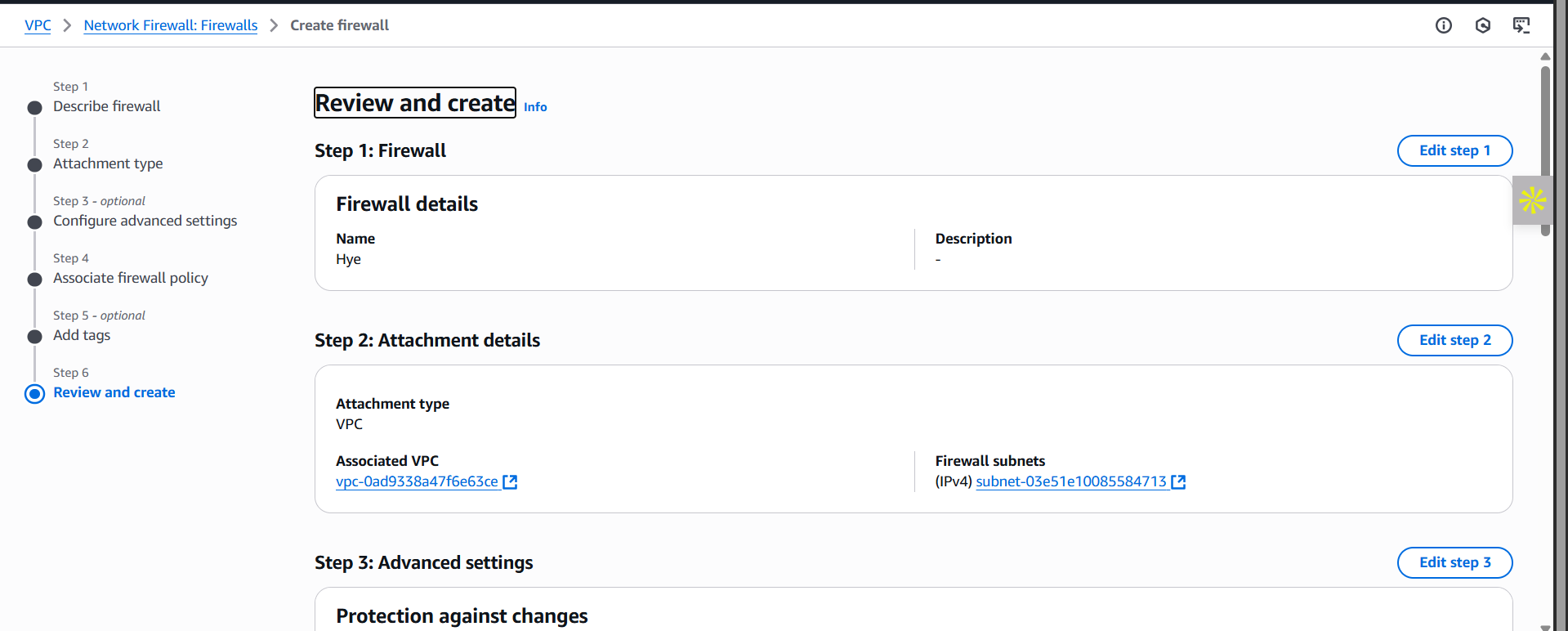


At next, we have to encounter the firewall policy:

Fill it, and move to the next: add tags if you need.



Review and create the firewall in the last step, ‘step 6’:



**Best Practices for a Secure Firewall in AWS:**

**1. Use a Least Privilege Rule Model**

* Only allow necessary traffic.
* Deny all other traffic by default.
* Follow the zero-trust principle: no traffic is trusted unless explicitly permitted.

**2. Separate Stateless and Stateful Rule Groups**

* Use stateless rules to handle high-volume, simple filters (e.g., IP blocks).
* Use stateful rules for deep inspection (e.g., domain blocking, protocol rules, DPI).

**3. Update Rule Groups Regularly**

* Regularly update signature/rule sets to reflect latest threat intelligence.
* Monitor AWS Managed Rule Groups if integrated with Firewall Manager.

**4. Use Domain Filtering**

* Block access to malicious domains using DNS domain filtering in stateful rules.
* Enforce allow-listed domains for outbound connections, especially for sensitive workloads.

**5. Enable Logging and Monitoring**

* Send logs to CloudWatch, S3, or Kinesis Firehose.
* Enable alerting for suspicious patterns like port scans or repeated denials.
* Use CloudTrail to track configuration changes.

**6. Test Rules Before Production Deployment**

* Use test environments to verify firewall rule behavior.
* Simulate traffic flows to confirm intended behavior (allowed/blocked).

**7. Align with Compliance Requirements**

* Define rules according to your industry’s security frameworks (e.g., PCI DSS, HIPAA).
* Document and audit firewall configurations periodically.

**8. Use Firewall Policies Efficiently**

* Create centralized policies and reuse them across multiple firewalls in different VPCs.
* Use tag-based access control to manage large deployments.

**9. Protect Critical Subnets First**

* Place the firewall in public-facing VPC subnets, especially where Internet or VPN/NAT Gateway is involved.
* Use dedicated subnets for firewall endpoints.

**10. Integrate with AWS Firewall Manager**

* If managing multiple accounts, use AWS Firewall Manager for centralized rule enforcement and compliance across all VPCs.

--The End--