

TRANSFORM
TOMORROW
TOGETHER

Managing Compute for Amazon EKS

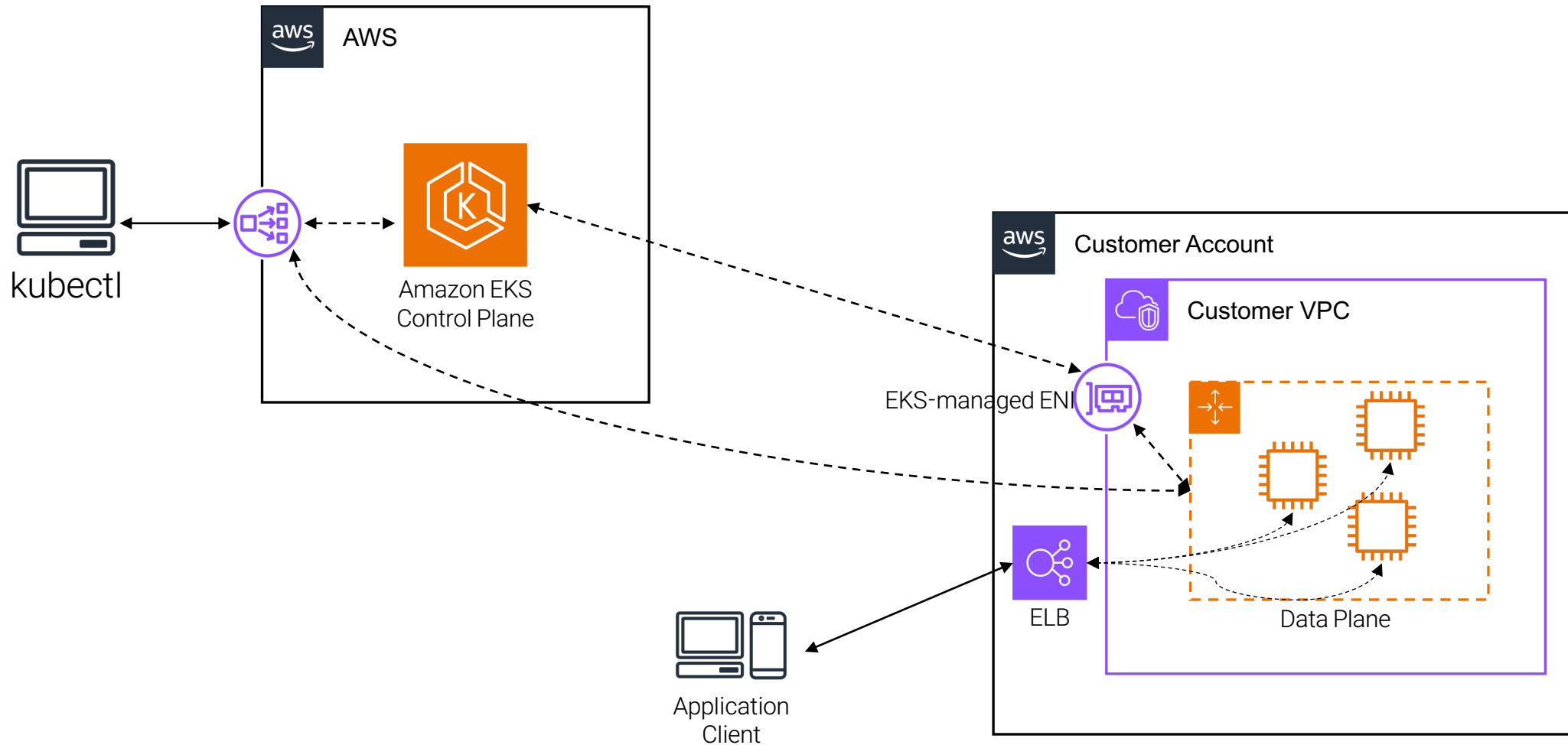
Specialty Service Unit

April. 2025

Contents

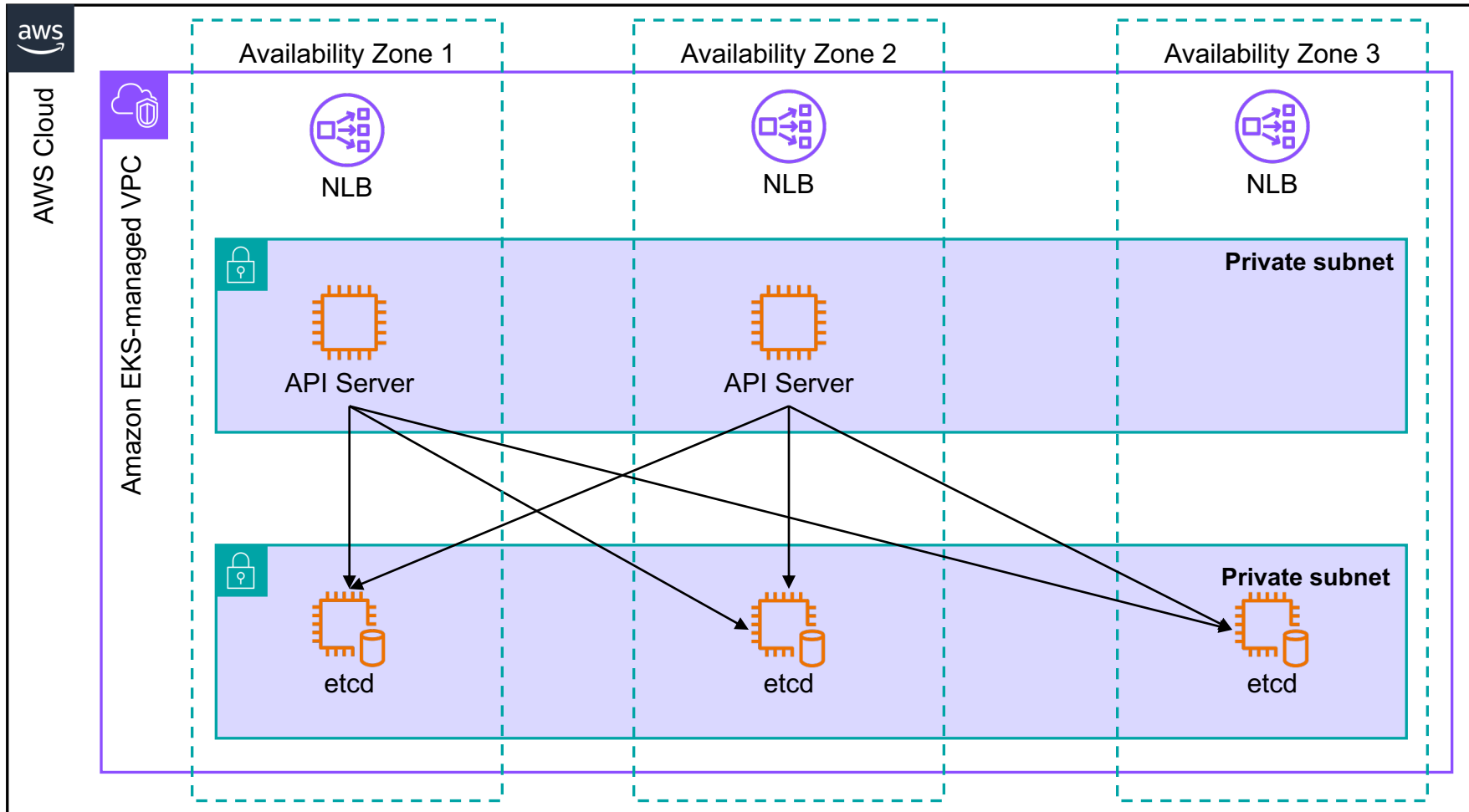
1. Amazon EKS: Cluster
2. Amazon EKS: Control Plane
3. Amazon EKS: Data Plane
4. Amazon EKS: Managed Node Group
5. Amazon EKS: Managed Node Group & Fargate
6. Amazon EKS: Fargate node
7. Amazon EKS: Fargate 고려사항
8. Amazon EKS: Fargate Lifecycle

1. Amazon EKS: Cluster



2. Amazon EKS: Control Plane

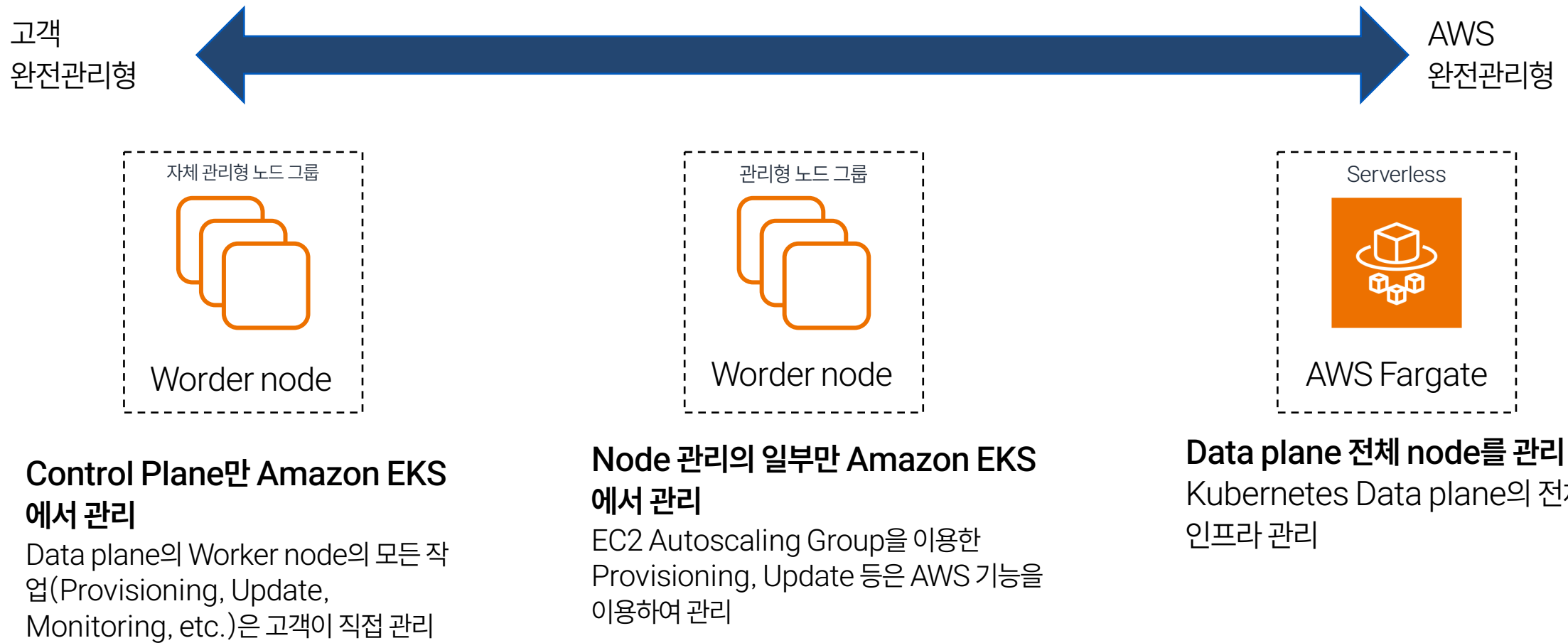
Amazon EKS Cluster 생성시 제공되는 Control Plane은 완전 관리형 기능으로 AWS 직접 운영을 책임지고 관리합니다.



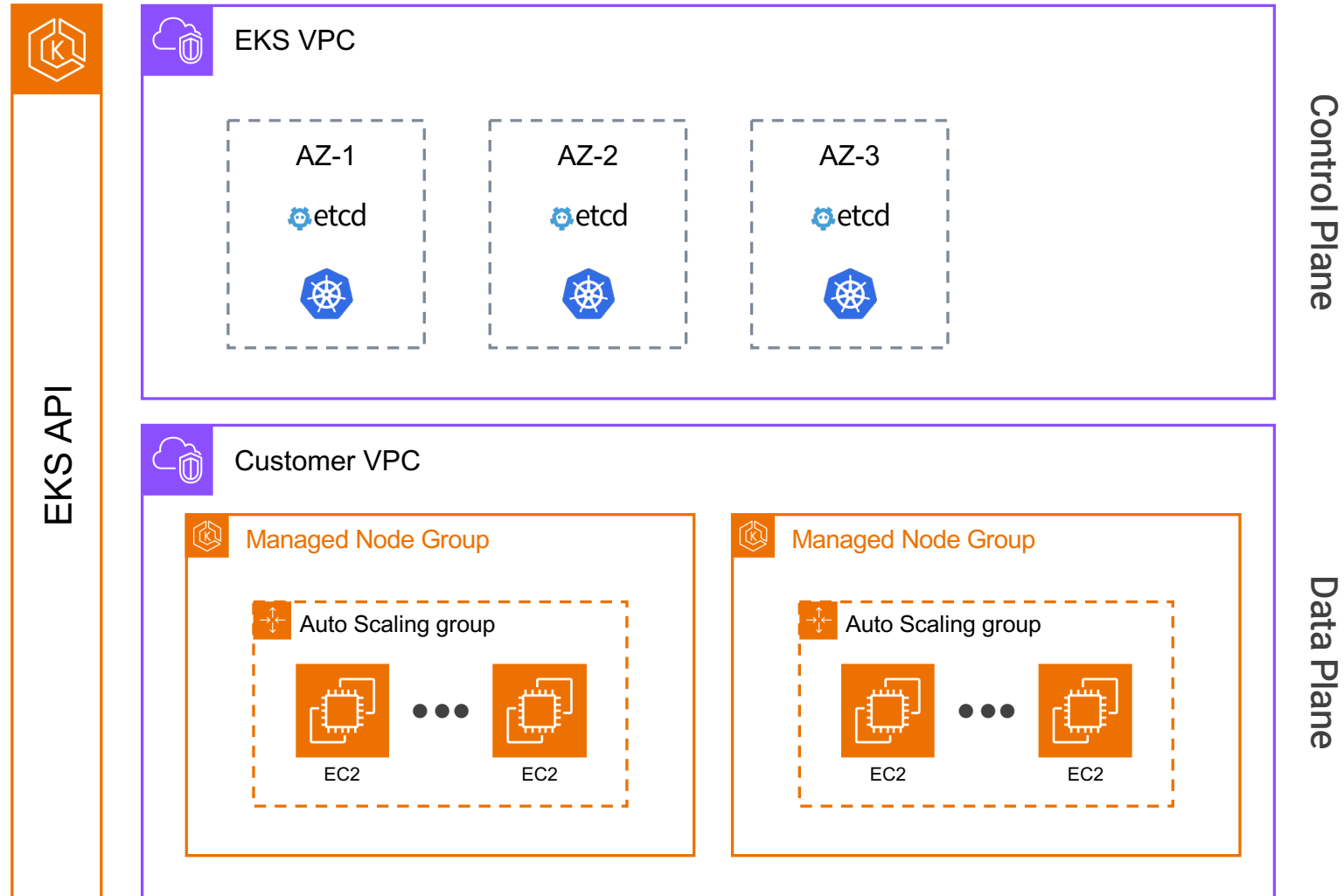
Fully managed

- API Server x 2
- etcd x 3
- Multi-AZ

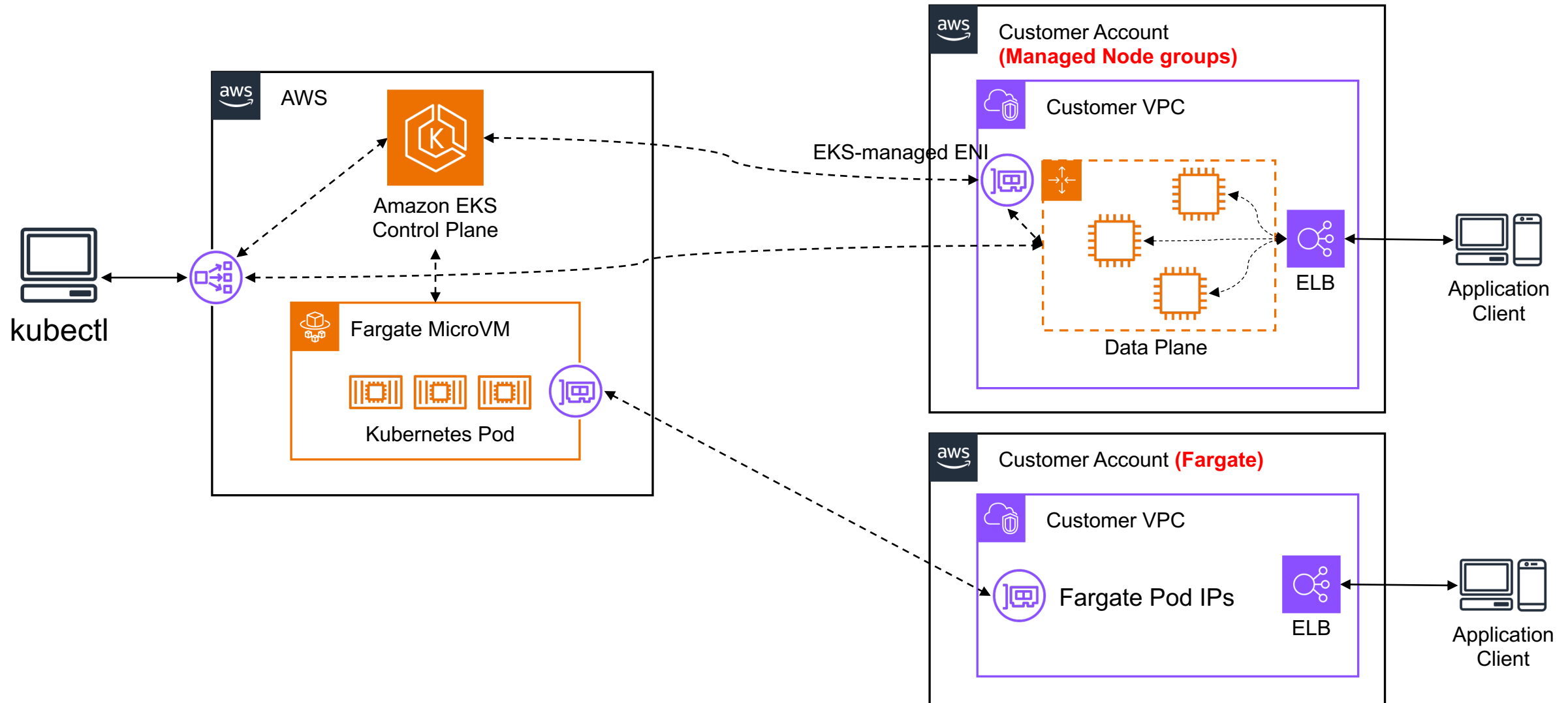
3. Amazon EKS: Data Plane



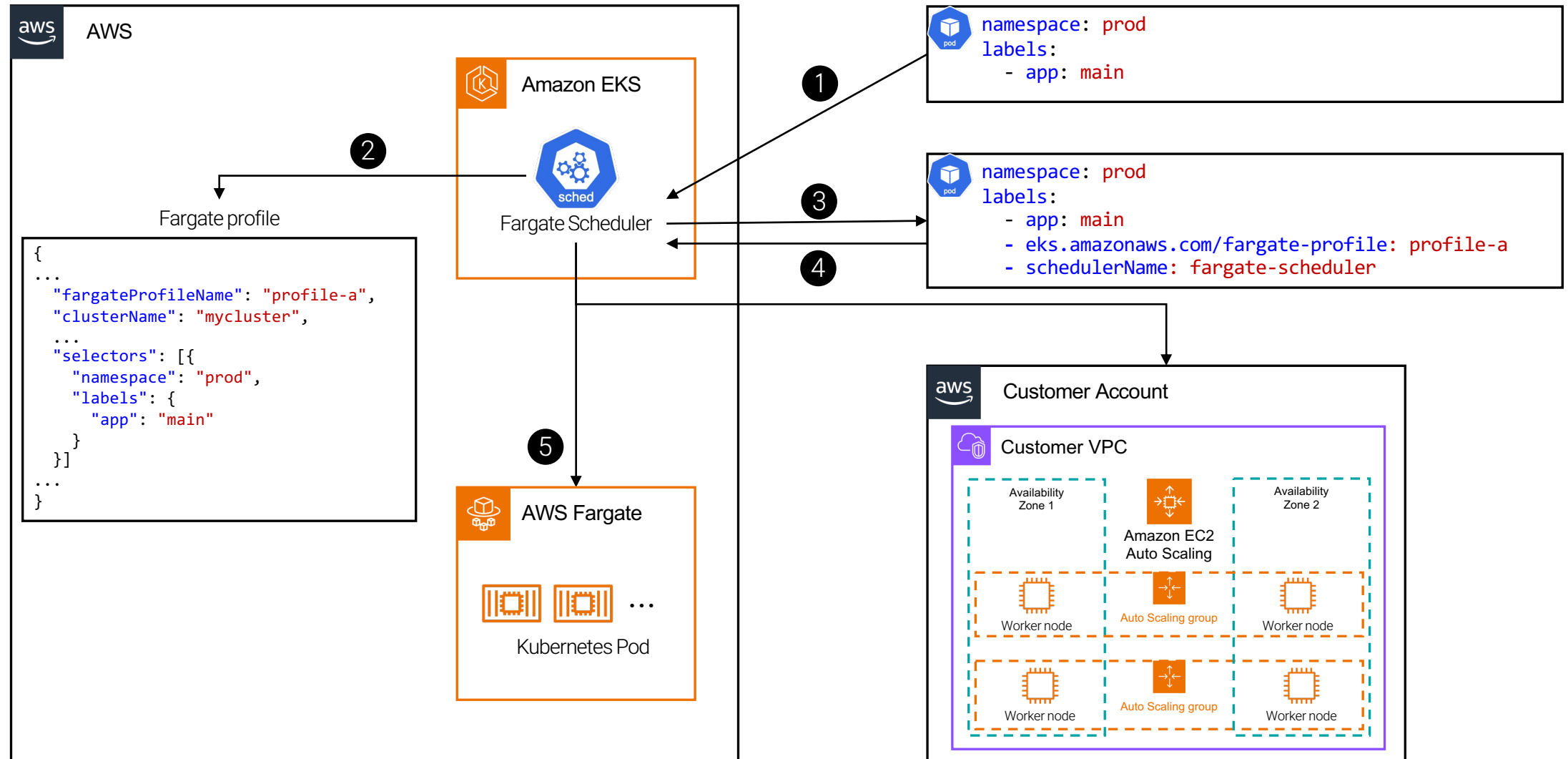
4. Amazon EKS: Managed Node Group



5. Amazon EKS: Managed Node Group & Fargate



6. Amazon EKS: Fargate node



7. Amazon EKS: Fargate 고려사항 (1)

Category	Description
Infra	Node에 직접 접속 불가 (ex: SSH)
	프라이빗 서브넷에만 Fargate 노드 실행 가능
	각 Pod들은 고유한 CPU/메모리 리소스, 기본 커널, Network Interface를 할당 받음
	EBS 사용 불가
	<u>Fargate VM의 OS 패치가 AWS가 자동으로 수행하기 때문에 OS 패치 도중 Pod 가 재시작 될 수 있음</u> <ul style="list-style-type: none">Kubernetes PDB(Pod Disruption Budget) 설정을 통해 반드시 실행되어야 하는 최소 Pod 수를 지정하여 해결해야 함^{1) 2)}
Fargate Profile	Fargate profile을 생성해야 Fargate Node에 Pod 배포 가능.
	생성가능한 최대 Fargate profile은 10개 (기본) → Service Quota ³⁾ 를 통해 증가 요청 가능
	Fargate profile에 지정된 Pod Selector(namespace, label)에 포함하지 않는 Pod를 배포할 경우, Pending 상태로 보류될 수 있음

1) Fargate OS 패치 - https://docs.aws.amazon.com/ko_kr/eks/latest/userguide/fargate-pod-patching.html

2) Kubernetes – PodDisruptionBudget 예시 - <https://kubernetes.io/ko/docs/concepts/workloads/pods/disruptions/#pdb-example>

3) AWS Service Quota - <https://docs.aws.amazon.com/eks/latest/userguide/service-quotas.html>

7. Amazon EKS: Fargate 고려사항 (2)

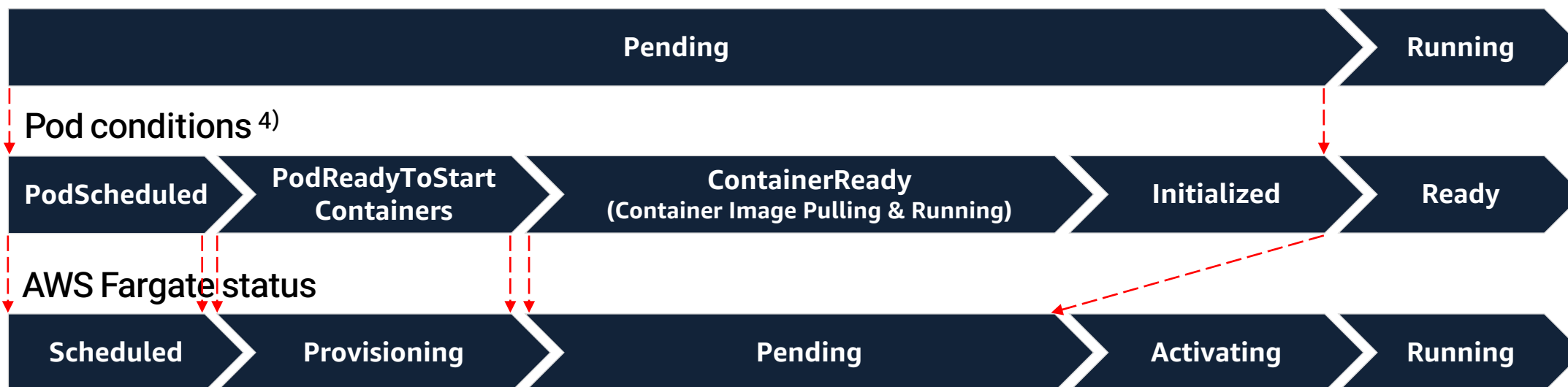
Category	Description																
Kubernetes	AWS VPC CNI 외 다른 CNI 사용 불가(ex: Calico, Flannel)																
	DaemonSet 지원 안함 → Daemon이 필요한 경우, Pod Sidecar 컨테이너 구성하여 사용																
	Privileged Container 지원 안함																
	Pod manifest에서 <u>HostPort</u> 또는 <u>HostNetwork</u> 지정불가																
	nofile, nproc의 soft limit 해제를 위해서는 Container image 빌드 시 직접 해제해야 함 (ulimit)																
	Autoscaling을 위해 VPA(Vertical Pod Autoscaler) 및 HPA(Horizontal Pod Autoscaler)를 사용하여 포드의 규모를 조정 가능 <ul style="list-style-type: none"> Pod Running 까지 약 3~5분 정도 소요되기 때문에, Autoscaling 기준을 Pod Running 소요 시간을 고려하여 설정 필요 																
과금 체계	<p>Pod의 CPU, Memory를 기준으로 부과되며, 사용 가능 조합은 다음과 같음 ¹⁾</p> <table> <tr> <th>vCPU</th><th>Memory</th></tr> <tr> <td>0.25</td><td>0.5GB, 1GB, 2GB</td></tr> <tr> <td>0.5</td><td>1GB, 2GB, 3GB, 4GB</td></tr> <tr> <td>1</td><td>2GB, 3GB, 4GB, 5GB, 6GB, 7GB, 8GB</td></tr> <tr> <td>2</td><td>4GB~16GB (1GB씩 증분)</td></tr> <tr> <td>4</td><td>8GB~30GB (1GB씩 증분)</td></tr> <tr> <td>8</td><td>16GB~60GB (4GB씩 증분)</td></tr> <tr> <td>16</td><td>32GB~120GB (8GB씩 증분)</td></tr> </table>	vCPU	Memory	0.25	0.5GB, 1GB, 2GB	0.5	1GB, 2GB, 3GB, 4GB	1	2GB, 3GB, 4GB, 5GB, 6GB, 7GB, 8GB	2	4GB~16GB (1GB씩 증분)	4	8GB~30GB (1GB씩 증분)	8	16GB~60GB (4GB씩 증분)	16	32GB~120GB (8GB씩 증분)
vCPU	Memory																
0.25	0.5GB, 1GB, 2GB																
0.5	1GB, 2GB, 3GB, 4GB																
1	2GB, 3GB, 4GB, 5GB, 6GB, 7GB, 8GB																
2	4GB~16GB (1GB씩 증분)																
4	8GB~30GB (1GB씩 증분)																
8	16GB~60GB (4GB씩 증분)																
16	32GB~120GB (8GB씩 증분)																

1) Fargate 포드 구성 세부 정보 - https://docs.aws.amazon.com/ko_kr/eks/latest/userguide/fargate-pod-configuration.html#fargate-cpu-and-memory

8. Amazon EKS: Fargate Lifecycle (Unofficial)

1. Fargate 기반 EKS에서는 새로운 pod를 생성하기 전에 Fargate node를 provisioning하기까지 시간이 소요됨.
2. AWS 에서 공식적으로 Fargate node의 provisioning 시간이 얼마나 소요되는지 공개한 적은 없지만, 사용자들이 추측하기로는 45초~60초 정도 소요된다고 공통적으로 이야기 하고 있으며, Provisioning 시간을 단축해달라는 요청을 지속적으로 하고 있지만, 수정되지는 않고 있음.
- 참고) “[EKS/Fargate] request: Improve Fargate Node Startup Time”
- <https://github.com/aws/containers-roadmap/issues/649>

Pod phase ³⁾



1) Kubernetes – Pod Lifecycle - <https://kubernetes.io/docs/concepts/workloads/pods/pod-lifecycle/>

2) AWS Fargate task startup time (p.10) - https://d1.awsstatic.com/events/Summits/reinvent2023/CON307_Reducing-AWS-Fargate-startup-times-by-lazy-loading-container-images.pdf

3) Pod phase - <https://kubernetes.io/docs/concepts/workloads/pods/pod-lifecycle/#pod-phase>

4) Pod conditions - <https://kubernetes.io/docs/concepts/workloads/pods/pod-lifecycle/#pod-conditions>

The background is a dark blue gradient. It features a complex pattern of glowing, translucent blue lines that resemble fiber optics or data streams, some of which are thicker and more prominent. These lines are set against a backdrop of faint, intricate circuit board patterns, including various traces, pads, and circular components. The overall effect is a high-tech, digital aesthetic.

Thank you