Protego: Cloud-Scale Multitenant IPsec Gateway

Jeongseok Son, Yongqiang Xiong, Kun Tan, Paul Wang, Ze Gan, Sue Moon

Microsoft Research

KAIST

HUAWEI

Enterprises are Moving to the Cloud

$ 58B → $ 202B

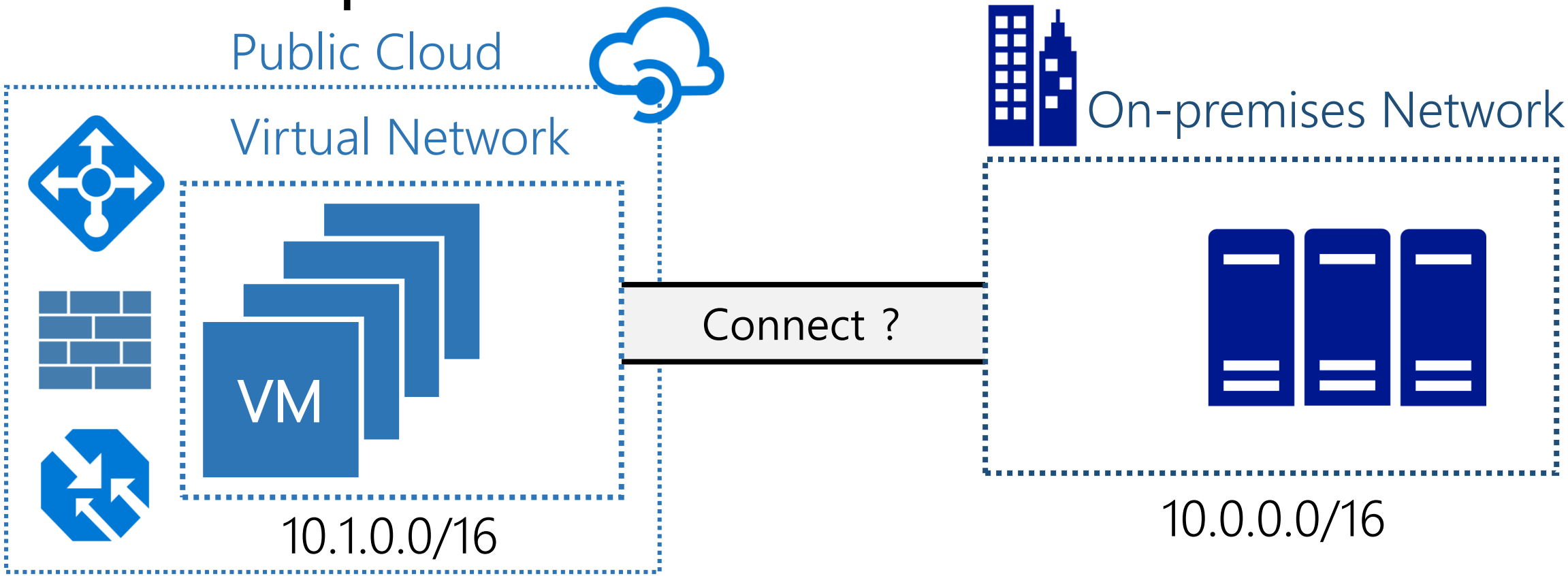
Public cloud services revenue
from 2009 to 2016

48 out of 50

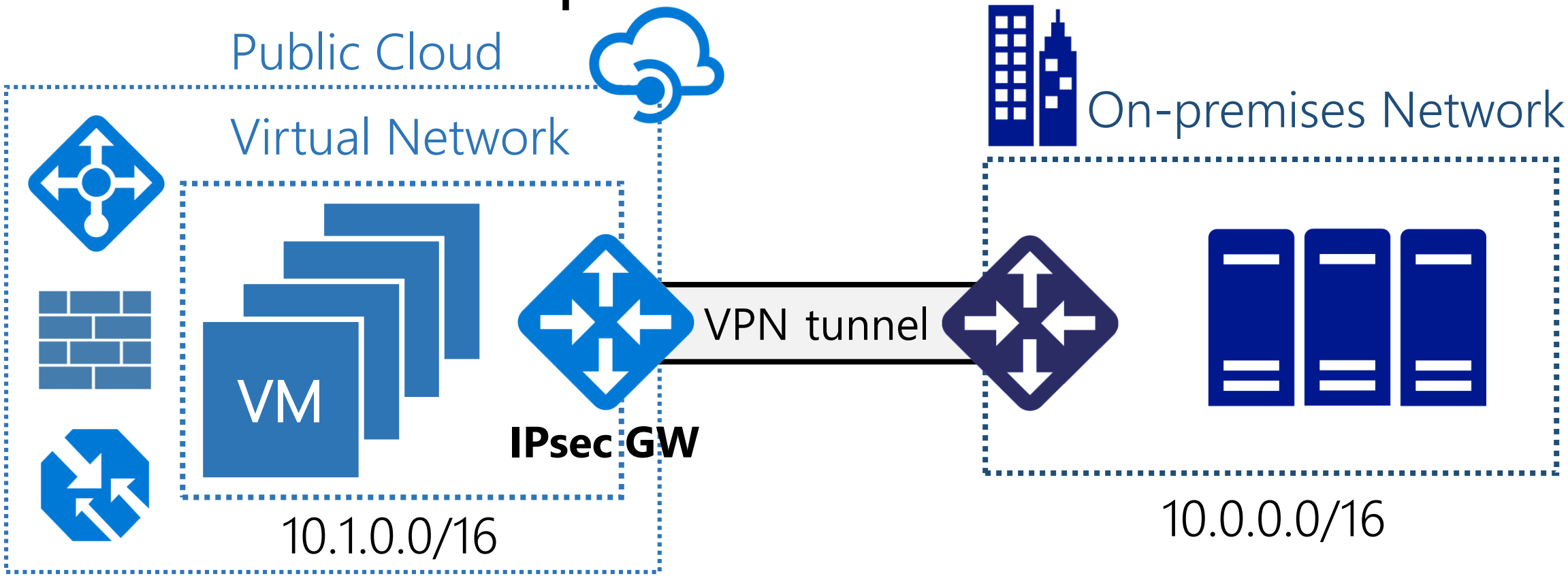
Fortune Global 50 companies
have announced cloud adoption

1. Forecast: Public Cloud Services, Worldwide and Regions, Industry Sectors, 2009-2014, Gartner
2. Global Cloud Spending Predicted To Reach $390B By 2020, Forbes

Cloud Services Provide Virtual Networks for Enterprises



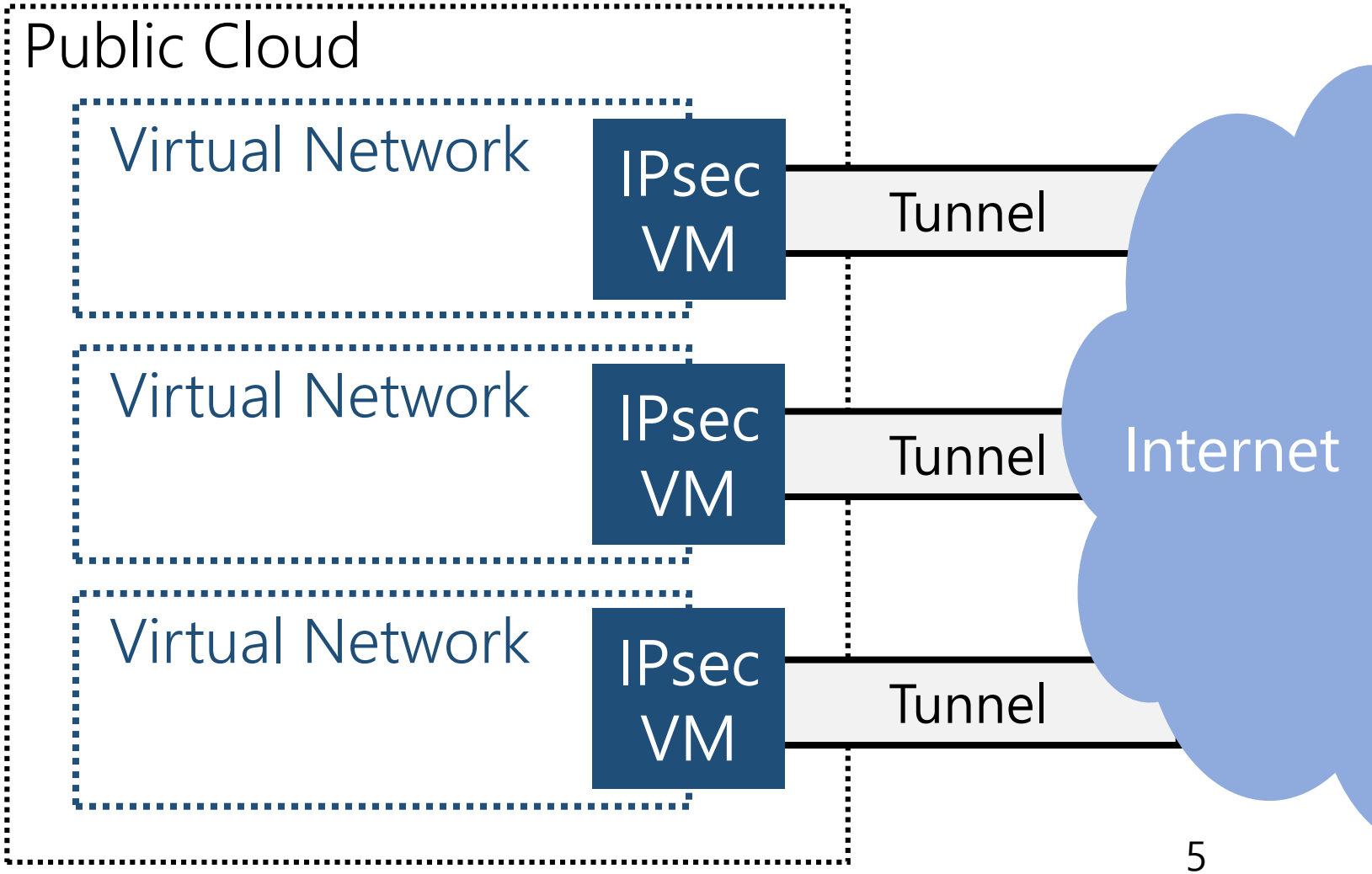
VPN Tunnels Connect Cloud and On-premises Data Center



Current IPsec GW Deployment: Assign VM to Each Virtual Network

Advantages:

- No additional HW installation
- Performance isolation
- Dynamic scaling



Problem:
IPsec GW VMs Are Under-utilized

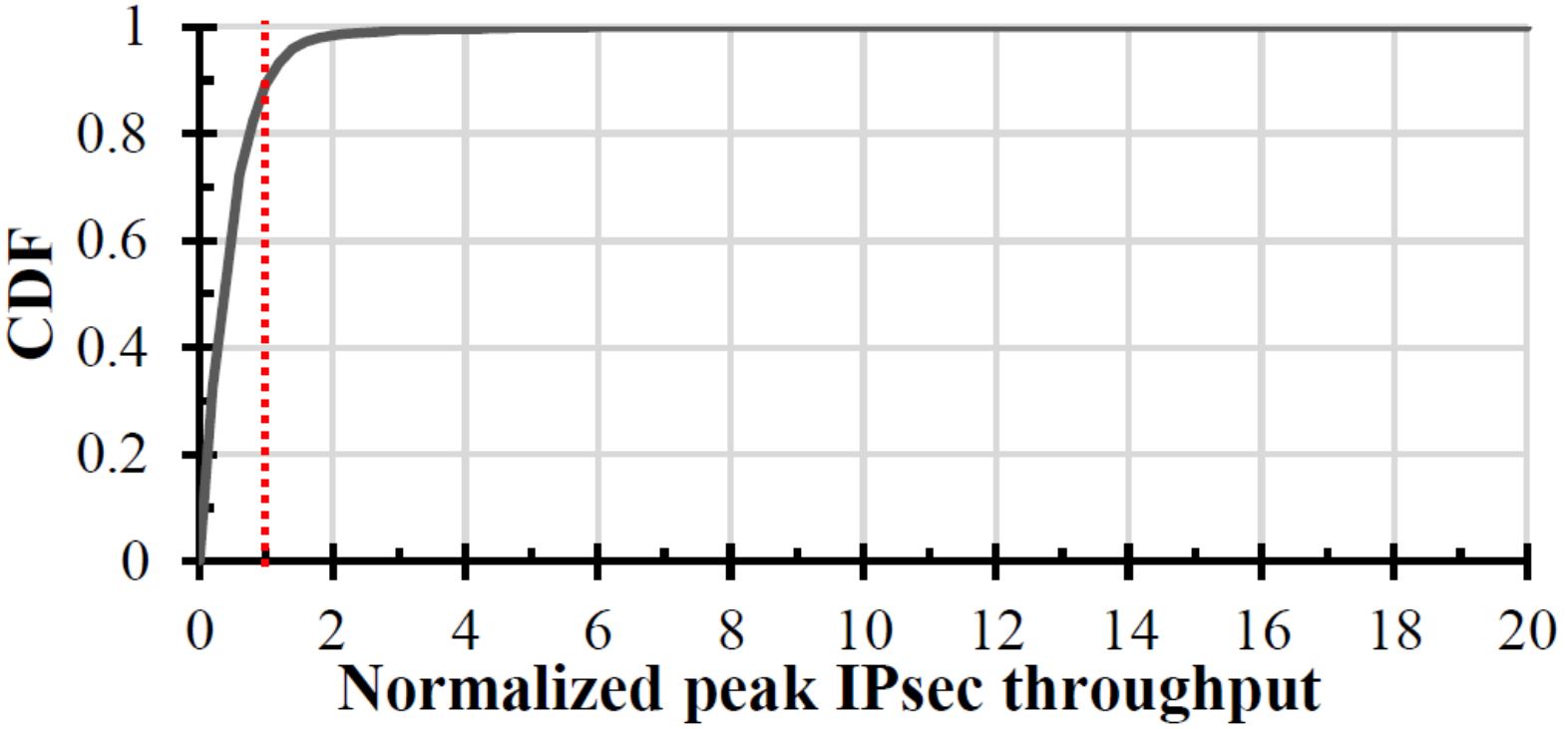


Figure: CDF of the peak IPsec traffic of data centers

In 90% of DCs,

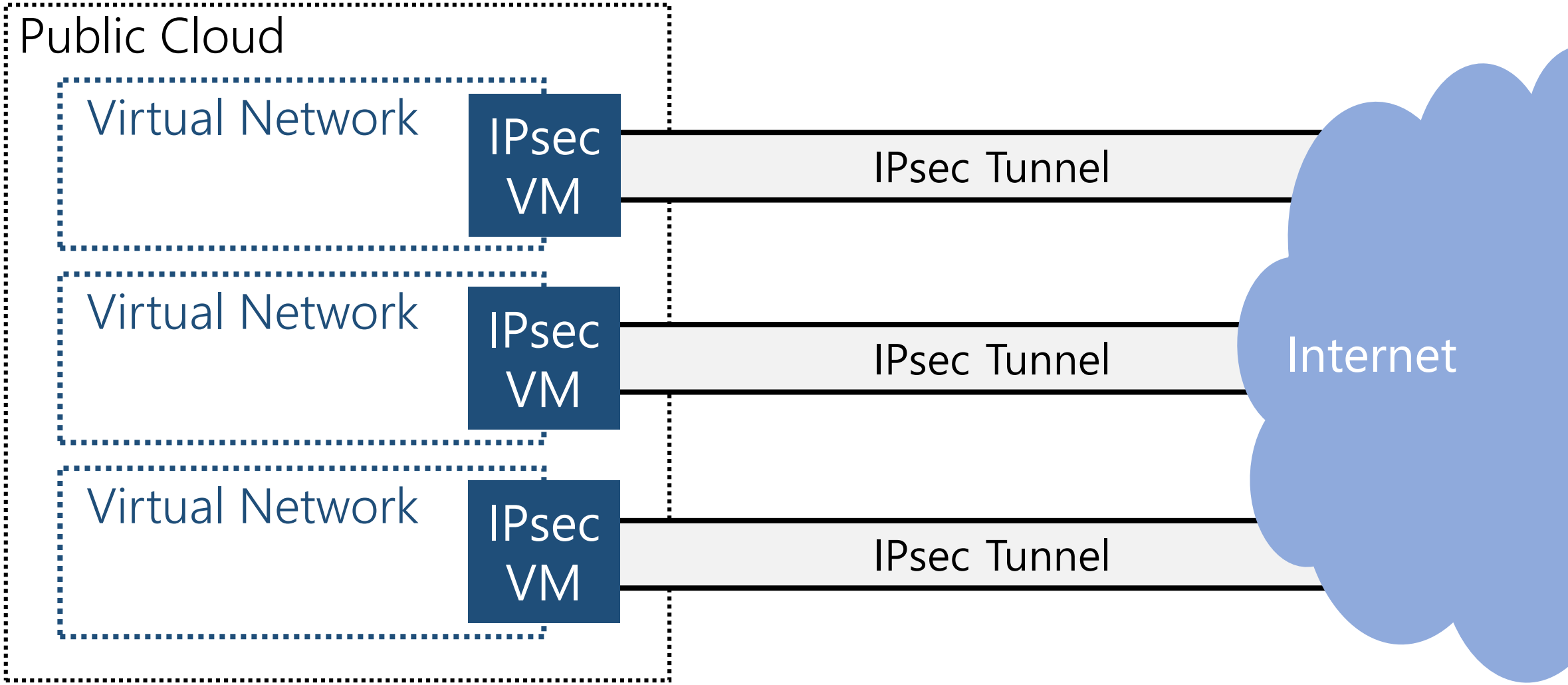
Daily *peak IPsec traffic* < *One GW capacity*

If GW is shared,

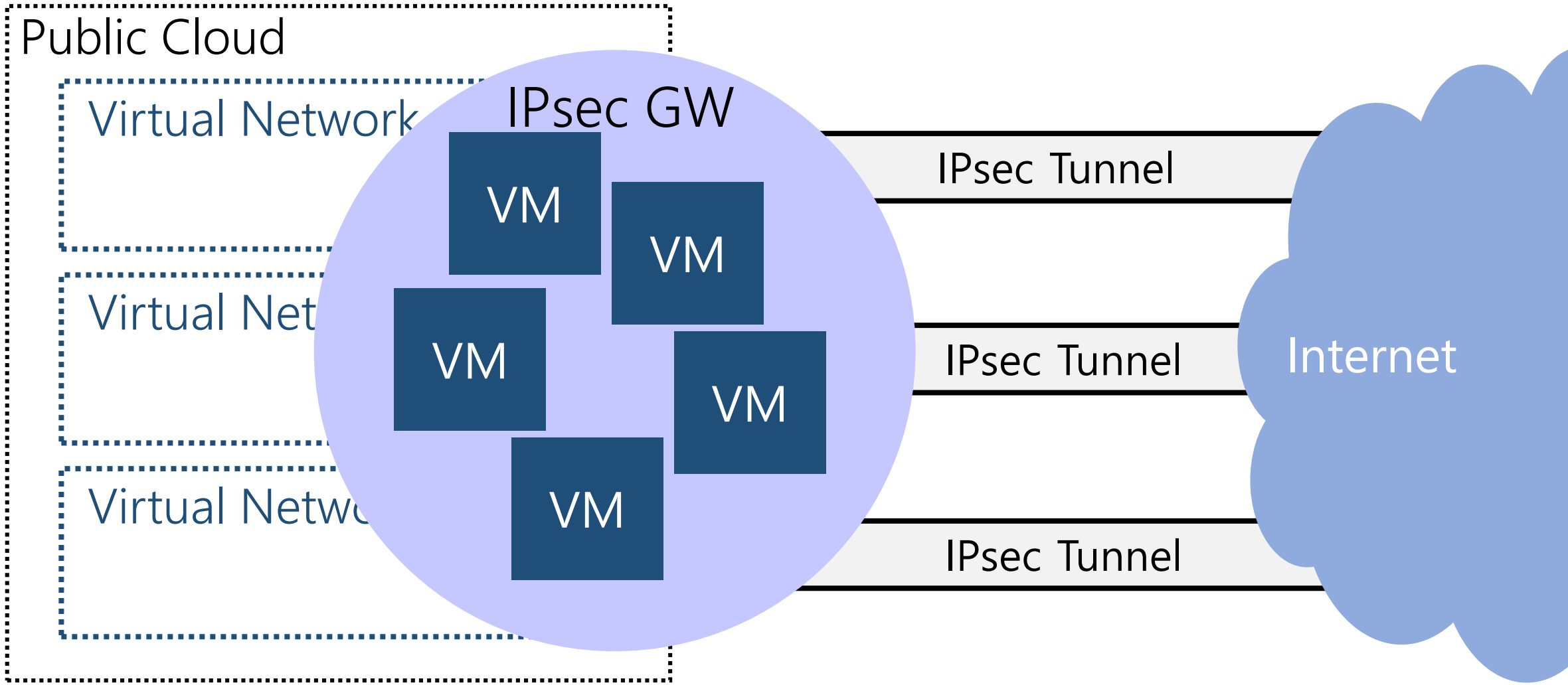
99+% of VMs can be saved.

How to serve multiple tunnels with shared resources for elasticity?

Current IPsec GW Deployment



Cloud-Scale Multitenant IPsec Gateway



Seamless Migration of IPsec Tunnel is the Key to Elasticity

To achieve elasticity

- When GWs are overloaded as the IPsec traffic increases

→ Add more VMs and migrate some tunnels to new VMs

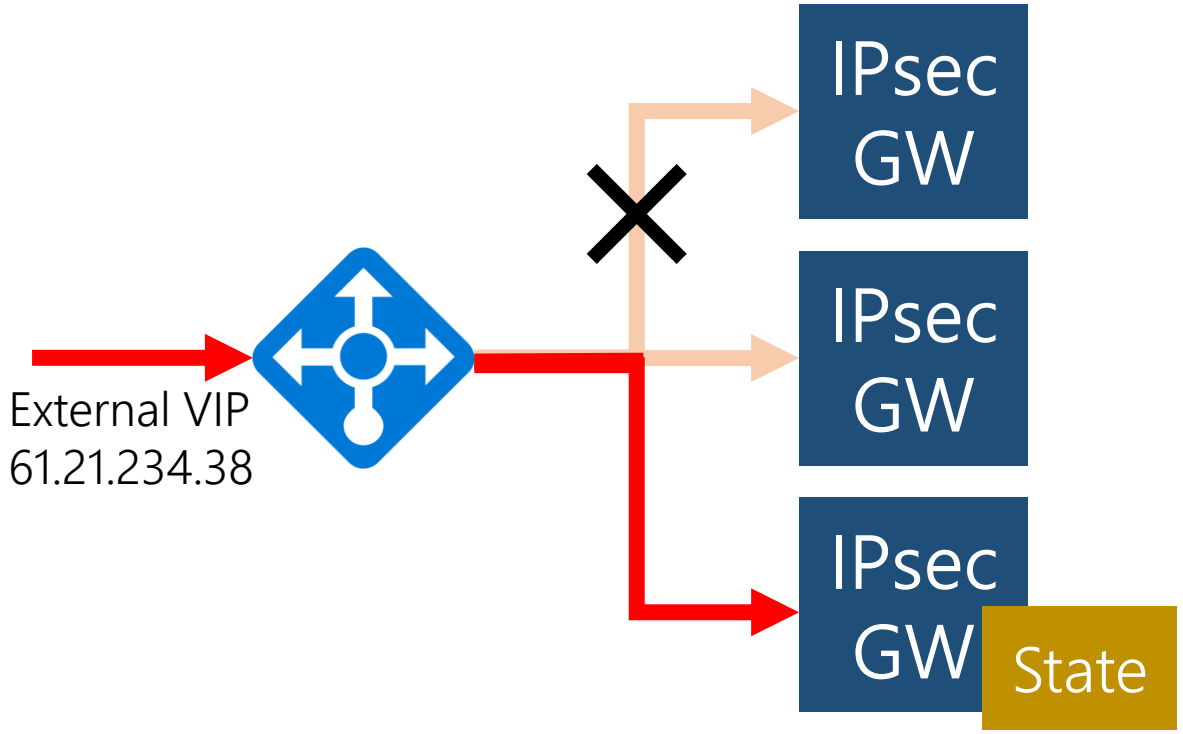
- When GWs are under-utilized as the IPsec traffic decreases

→ Migrate tunnels from some VMs and return the idle VMs

Quick migration scheme with minimal overhead is a key enabler of elasticity

Statefulness of IPsec Hinders Seamless Migration

- Strawman approach: Redirect IPsec packets to a different GW



Leads to tunnel destruction

How to move or share state between gateways?

Core Ideas of Protego

Separation of control and data planes

- Control Plane: Single control node
- Data Plane: Set of data nodes

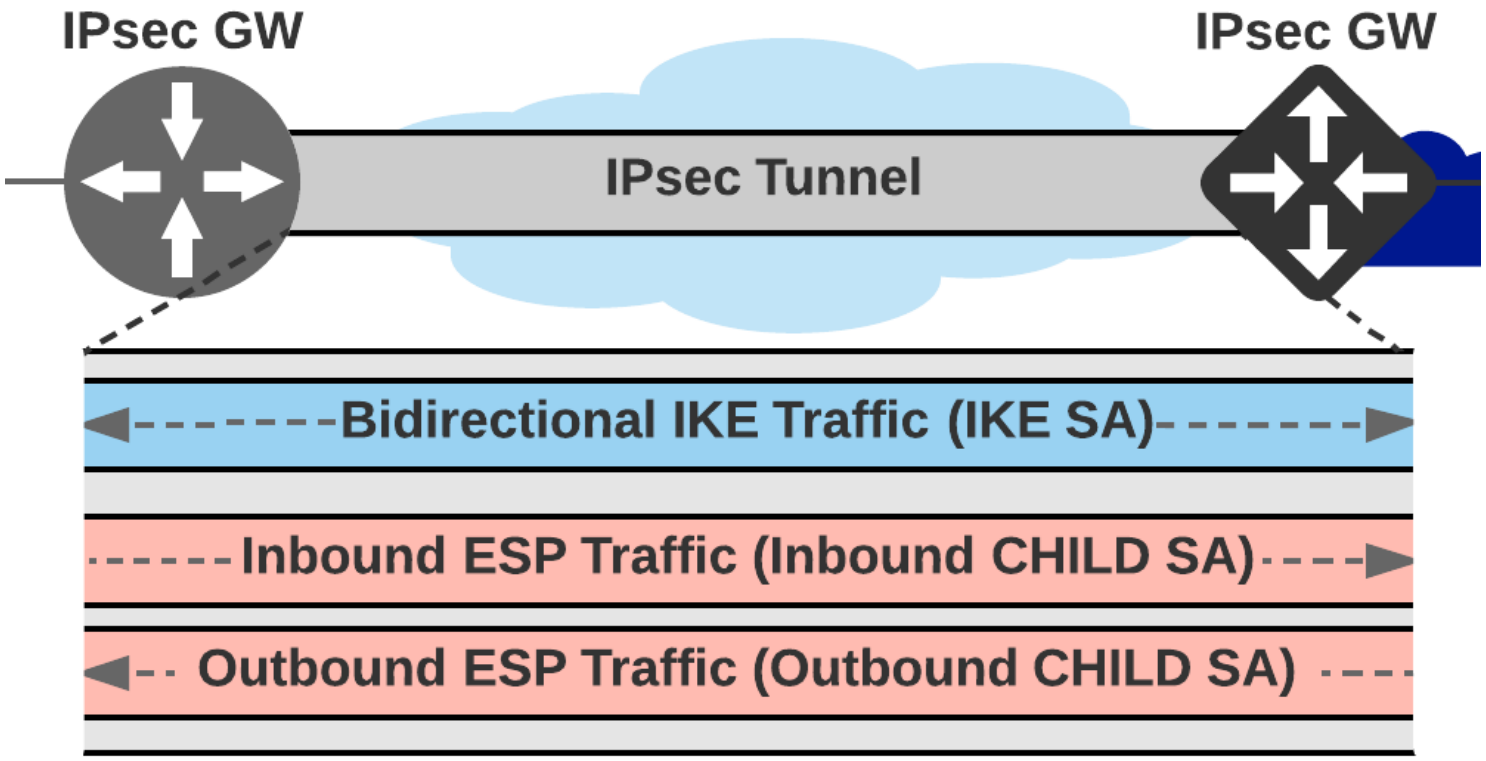
→ *Make IPsec (nearly) stateless in the data plane*

Tunnel migration by IPsec rekeying

→ *Migrate tunnels without packet loss and buffering*

Elastic provisioning algorithm

Breakdown of IPsec Protocol



Internet Key Exchange

- Setup Shared Attributes (Security Association)

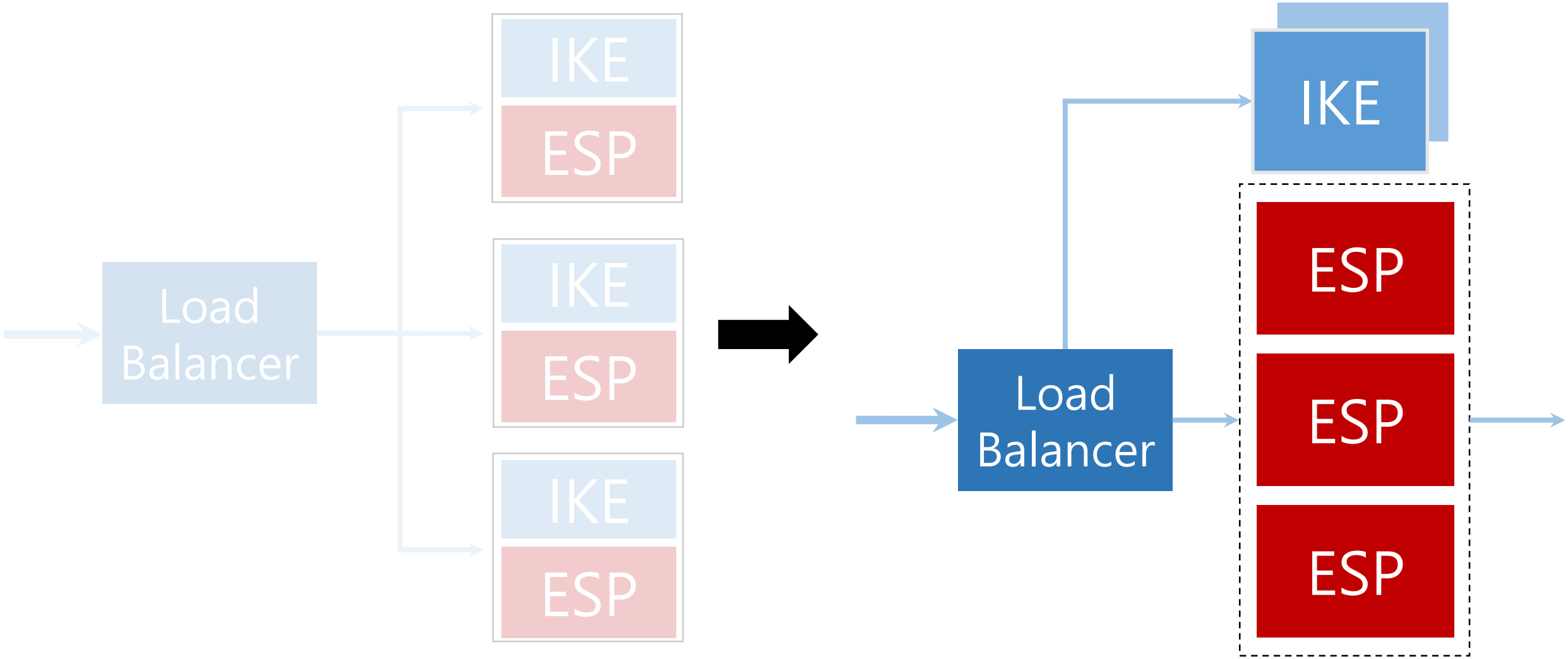
→ Carries control traffic

Encapsulating Security Payload

- Encryption/Decryption

→ Carries data traffic

Separation of Control and Data Plane



Rationale Behind the Separation

Infrequent IKE state update
and tiny IKE traffic compared to ESP traffic

- Stored in a central control node

  → *Data nodes do not maintain IKE state*

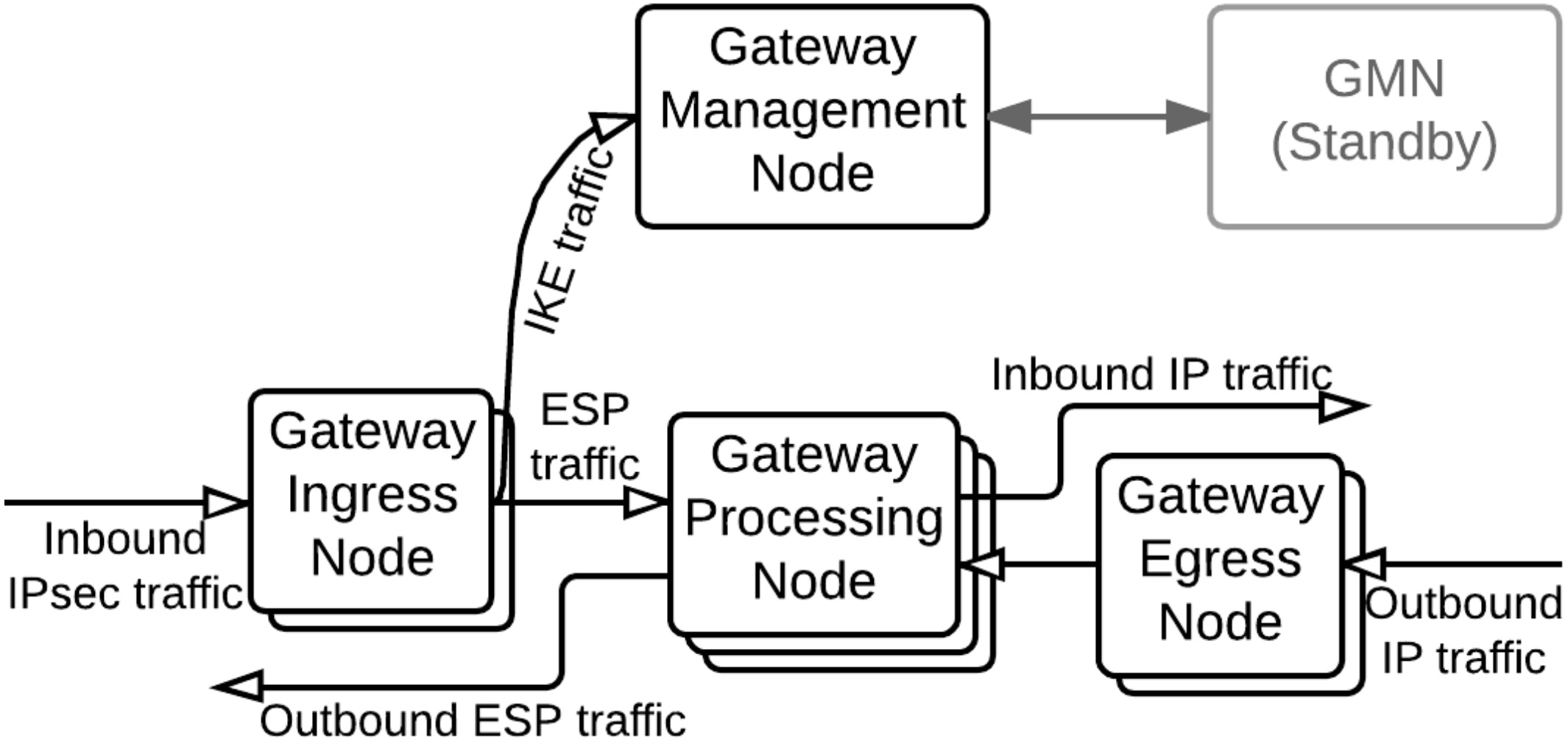
Frequent ESP state changes (every packet sent/received)
but quick re-initialization

- Reconstructed whenever necessary by rekeying

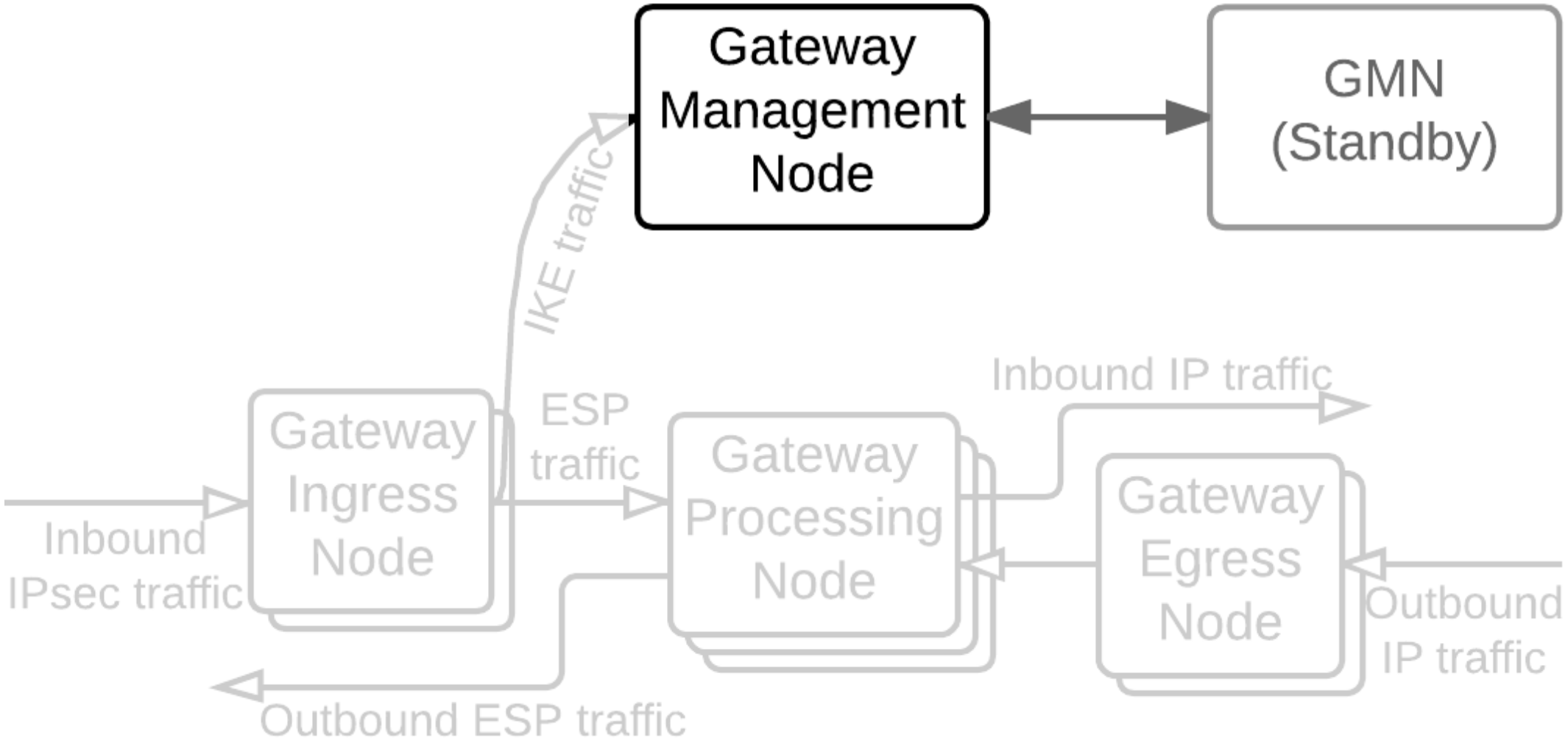
  → *Data nodes do not have to preserve ESP state*

Any data node can process any IPsec tunnel traffic

Protego Architecture Overview



Gateway Management Node



Gateway Management Node

IKE packet processing

- Negotiate a shared symmetric key for ESP
- Distribute the key to one of GPNs
- Save updated state to the standby GMN (High availability)

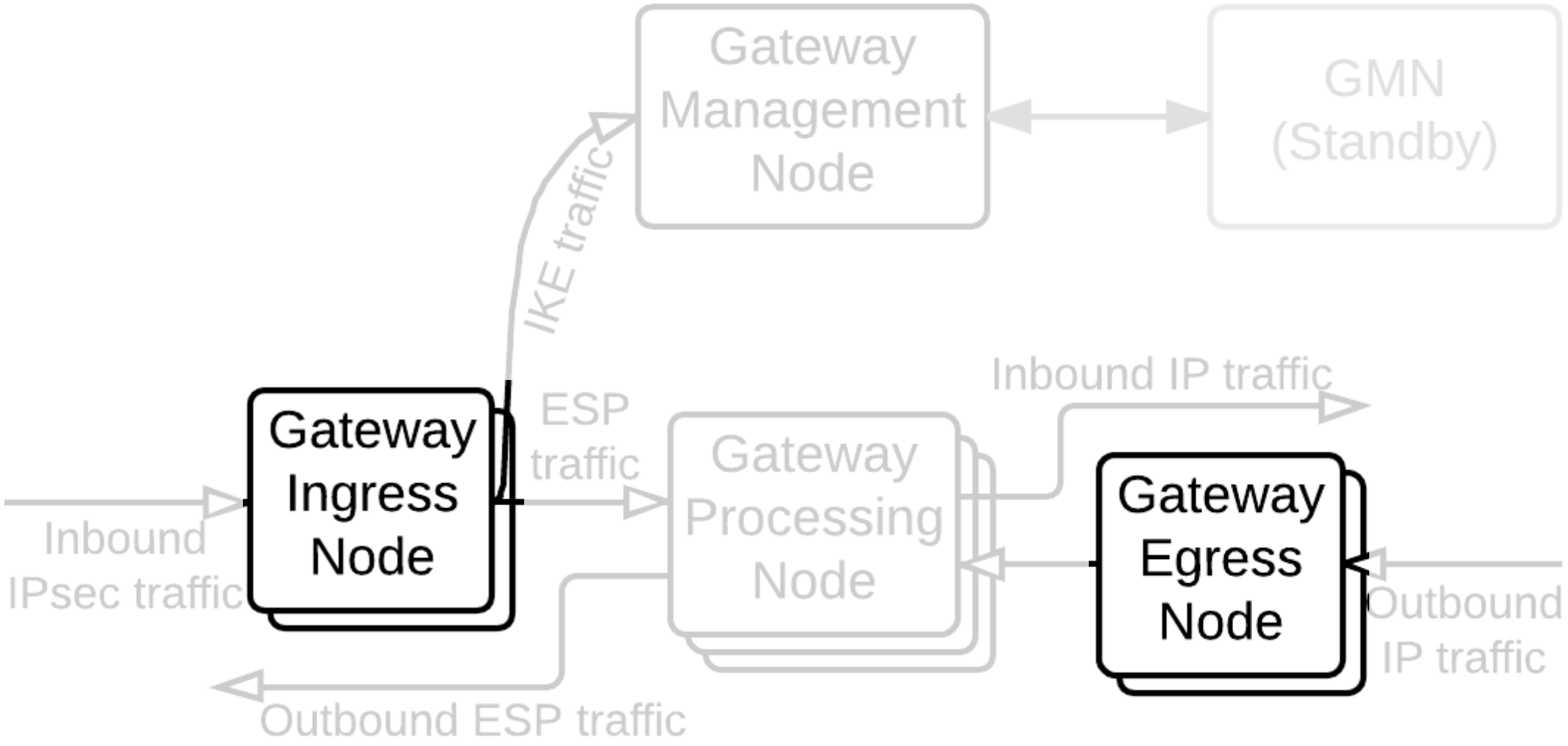
Resource management

- Adjust the number of GPNs by migrating tunnels

Traffic steering

- Insert forwarding rules to load balancers (GIN and GEN)

Gateway Ingress and Egress Node



Gateway Ingress and Egress Node

Traffic forwarding

- Rewrite the destination address to the address of a GPN

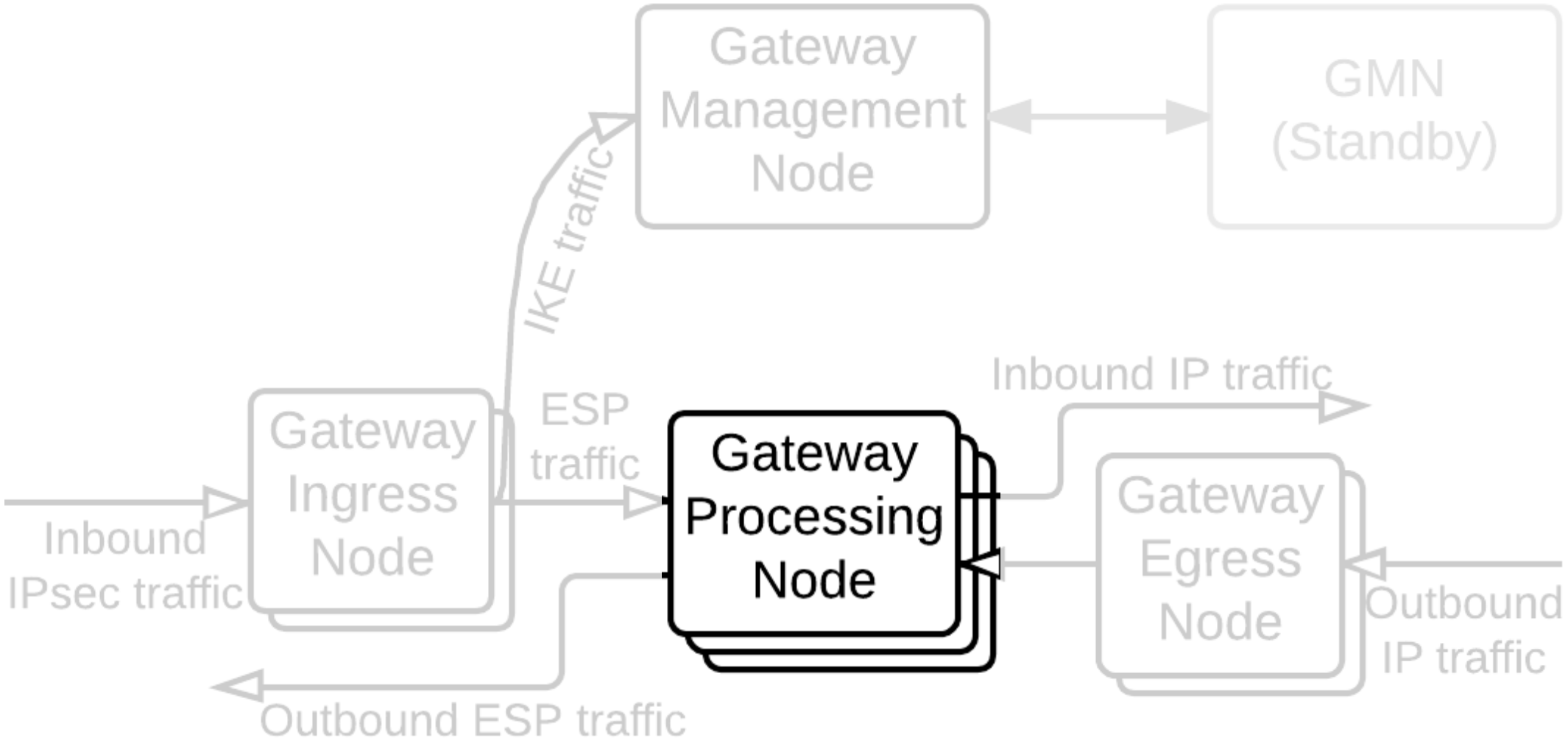
Rate limiting

- Enforce per-tunnel performance isolation

GPN failure detection

- Adaptive heartbeat by sampling and tagging

Gateway Processing Node



Gateway Processing Node

ESP packet processing

- Encryption and decryption of ESP packets

ESP processing is tricky to parallelize due to sequence number

→ Designed lock-free ESP processor (Check out the paper)

Leverage Rekeying Process to Migrate IPsec Tunnels Seamlessly

Original purpose of rekeying

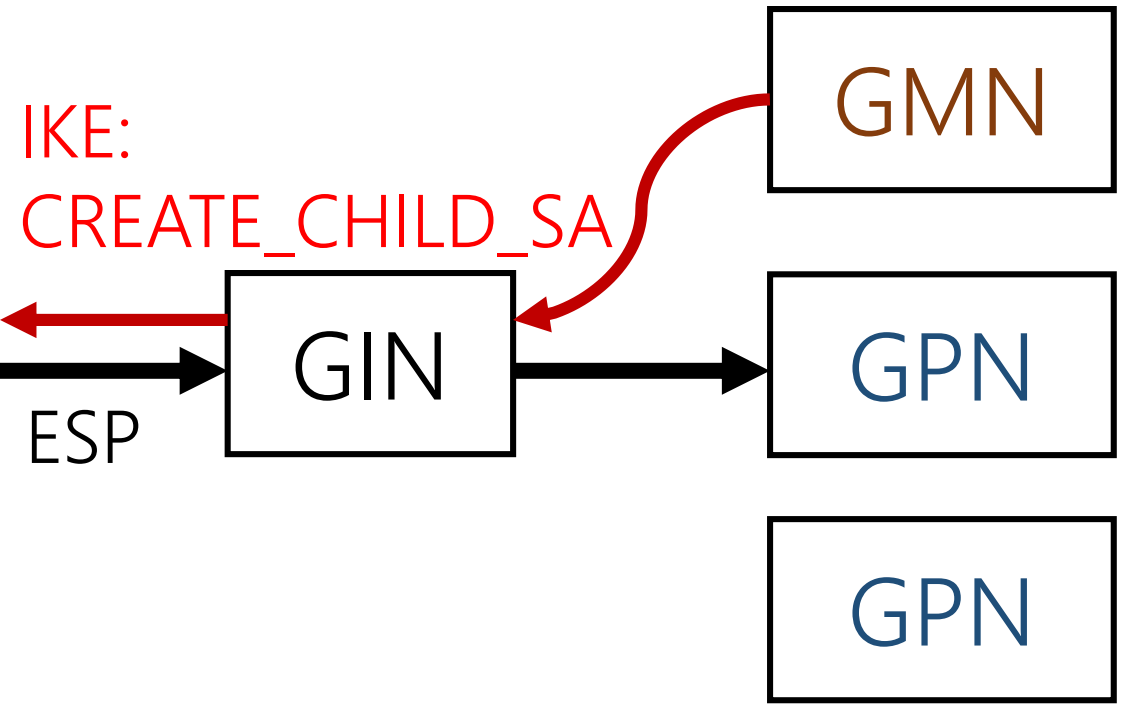
- IPsec gateways use keys for a limited amount of time/data
- Quickly re-negotiates ESP SA in single RTT

Leverage rekeying to quickly construct ESP state

- Create new ESP SA in the destination node
- Old ESP SA is alive until new ESP SA is used

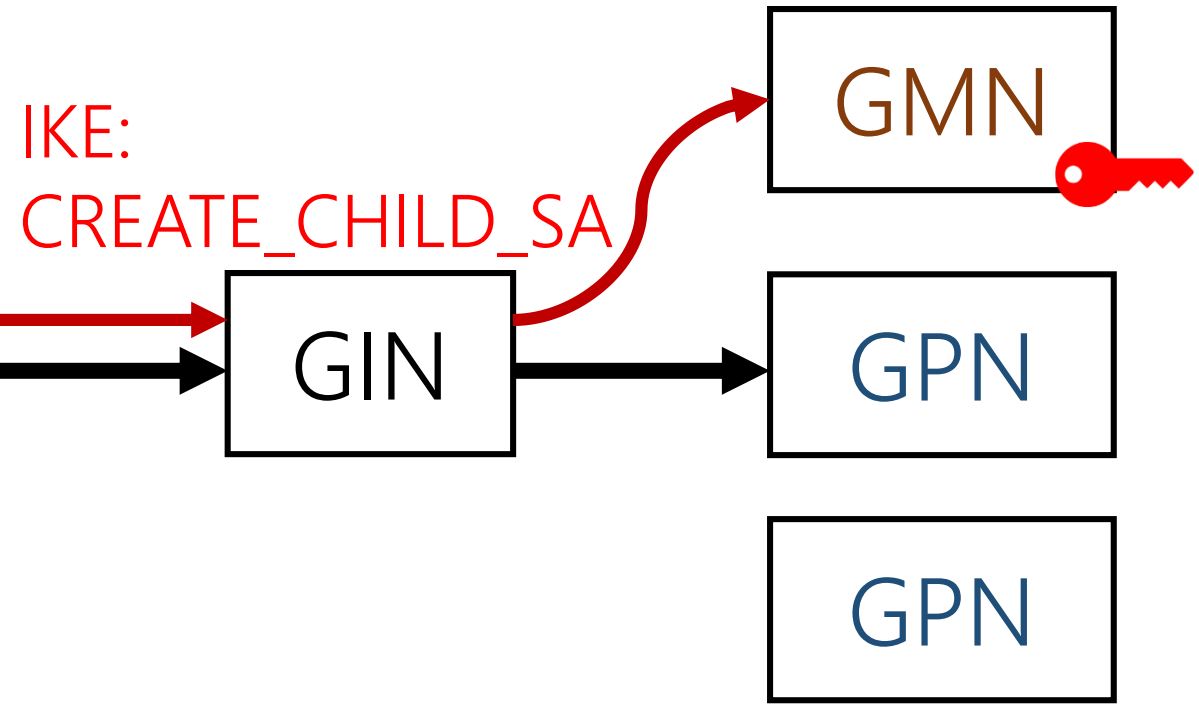
  → *No packet loss and buffering during migration process*

IPsec Tunnel Migration Process



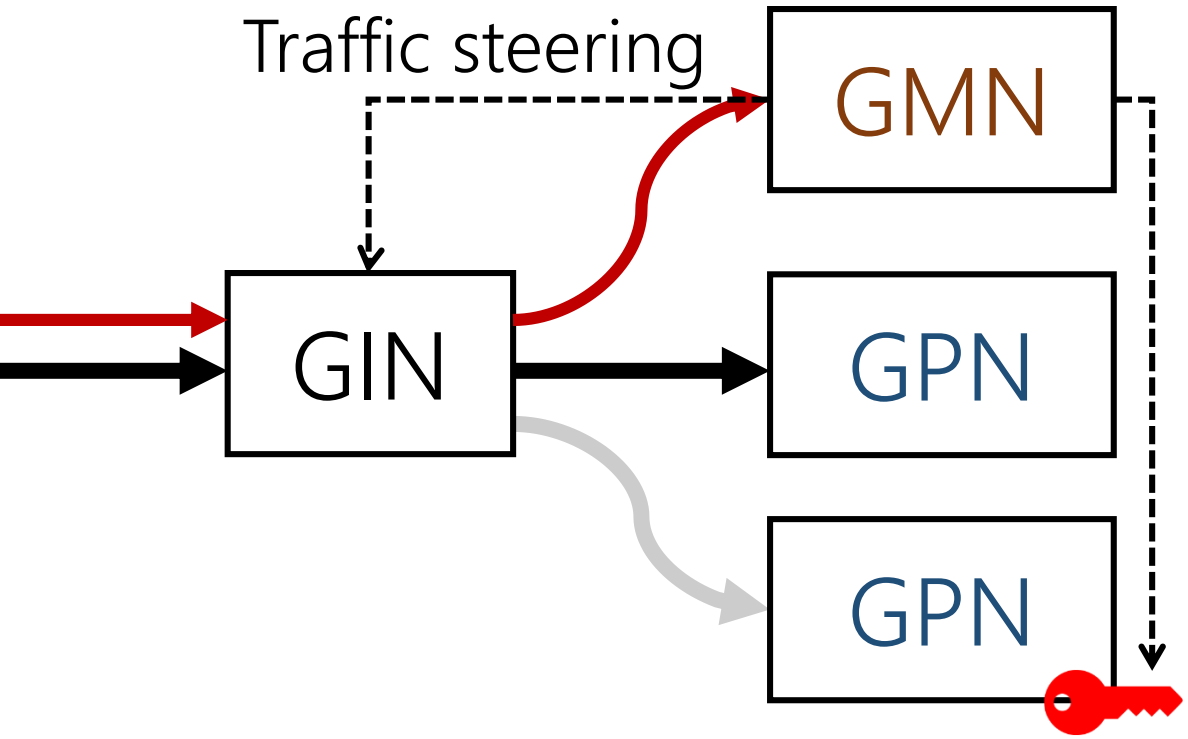
1. **GMN sends the CREATE_CHILD_SA request**

IPsec Tunnel Migration Process



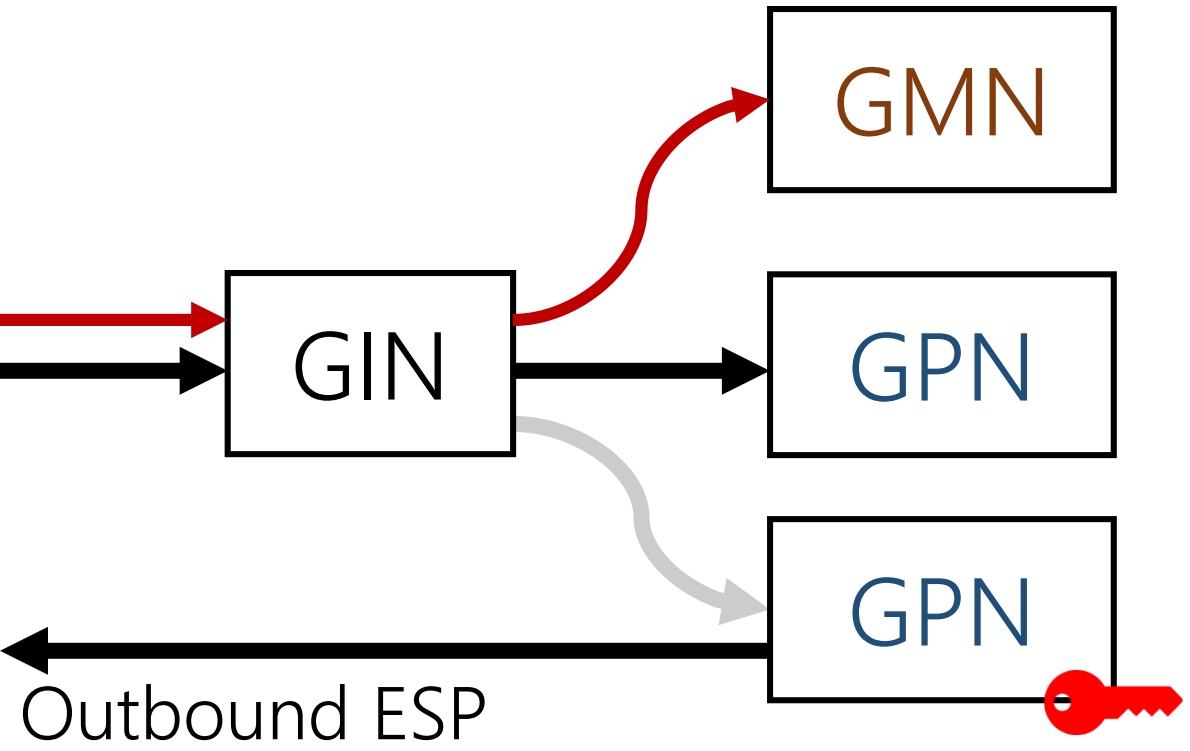
1. GMN sends the CREATE_CHILD_SA request
2. **GMN receives the CREATE_CHILD_SA response. (New CHILD_SAs are created)**

IPsec Tunnel Migration Process



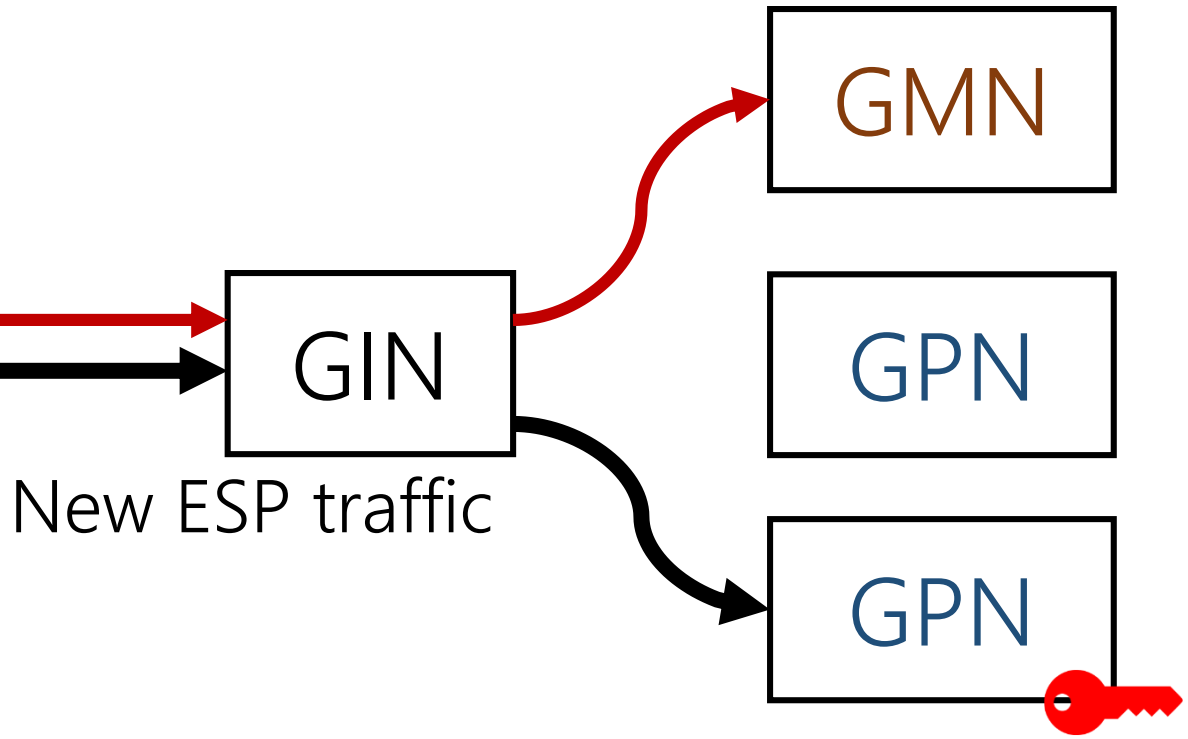
1. GMN sends the CREATE_CHILD_SA request
2. GMN receives the CREATE_CHILD_SA response. (New CHILD_SAs are created)
3. **GMN hands the new SAs over to a GPN, and inserts forwarding rules**

IPsec Tunnel Migration Process



1. GMN sends the CREATE_CHILD_SA request
2. GMN receives the CREATE_CHILD_SA response. (New CHILD_SAs are created)
3. GMN hands the new SAs over to a GPN, and inserts forwarding rules
4. **GPN starts to use the new outbound SA**

IPsec Tunnel Migration Process



1. GMN sends the CREATE_CHILD_SA request
2. GMN receives the CREATE_CHILD_SA response. (New CHILD_SAs are created)
3. GMN hands the new SAs over to a GPN, and inserts forwarding rules
4. GPN starts to use the new outbound SA
5. **GIN forwards the new ESP packets to the new GPN**

Elastic Resource Provisioning Algorithm

Objectives

- Minimize the resource usage
- Satisfying the throughput requirement of tenants

Model: 1-D bin packing (CPU usage)

- Item: IPsec tunnel
- Bin: GPN

Consolidation & Load Balancing

- Periodically consolidate GPNs (Consolidation interval)
- Instantly mitigate hotspots by migrating tunnels

Implementation

Gateway Ingress & Egress Node

- Extended Mux of Ananta load balancer (SIGCOMM ‘13)
- Packet filtering based on Windows NDIS Lightweight filter driver

Gateway Management Node

- Based on the IPsec service module of the Routing and Remote Access Service in Windows Server 2012 R2.

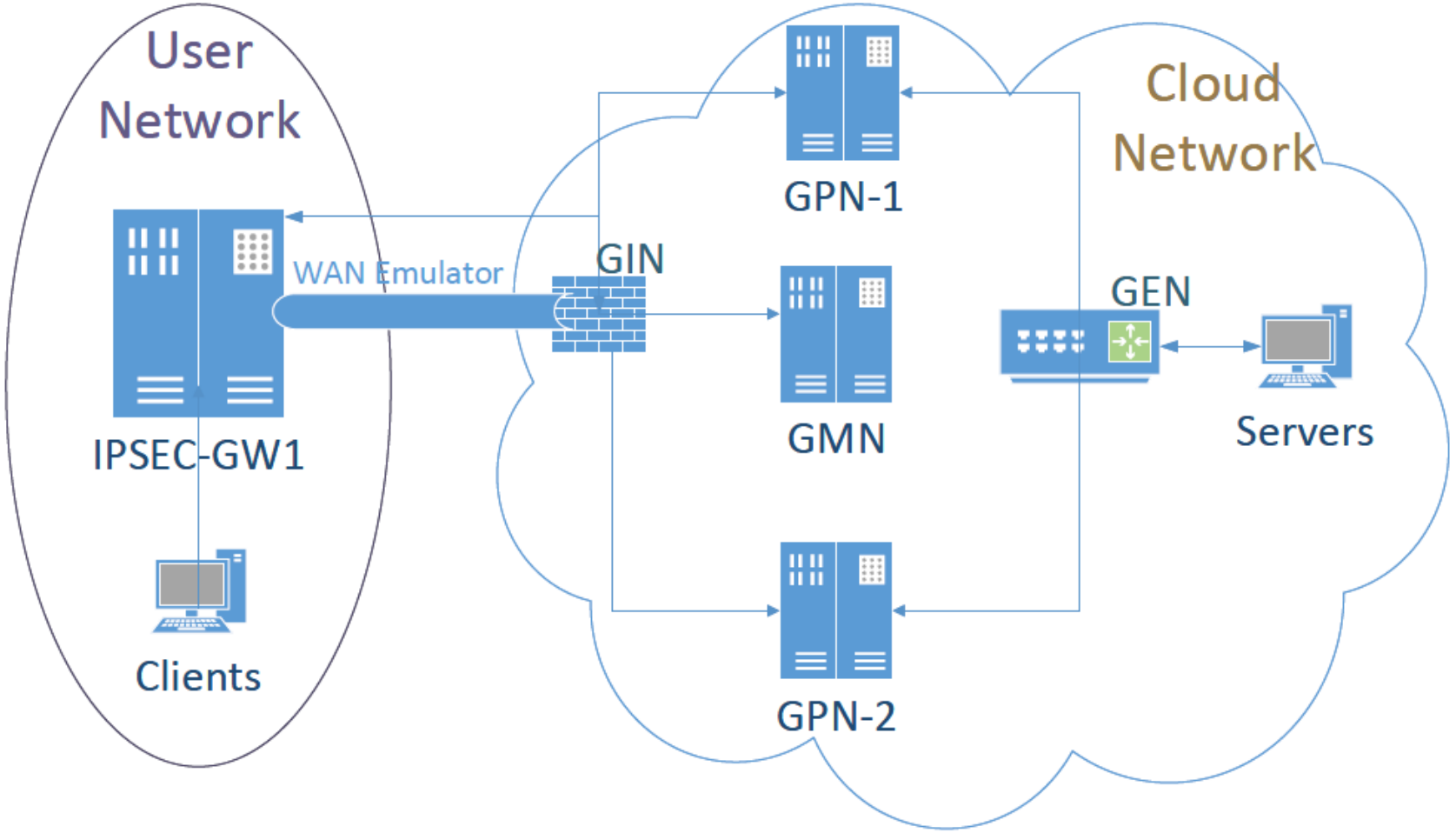
Gateway Processing Node

Refer to the paper for details

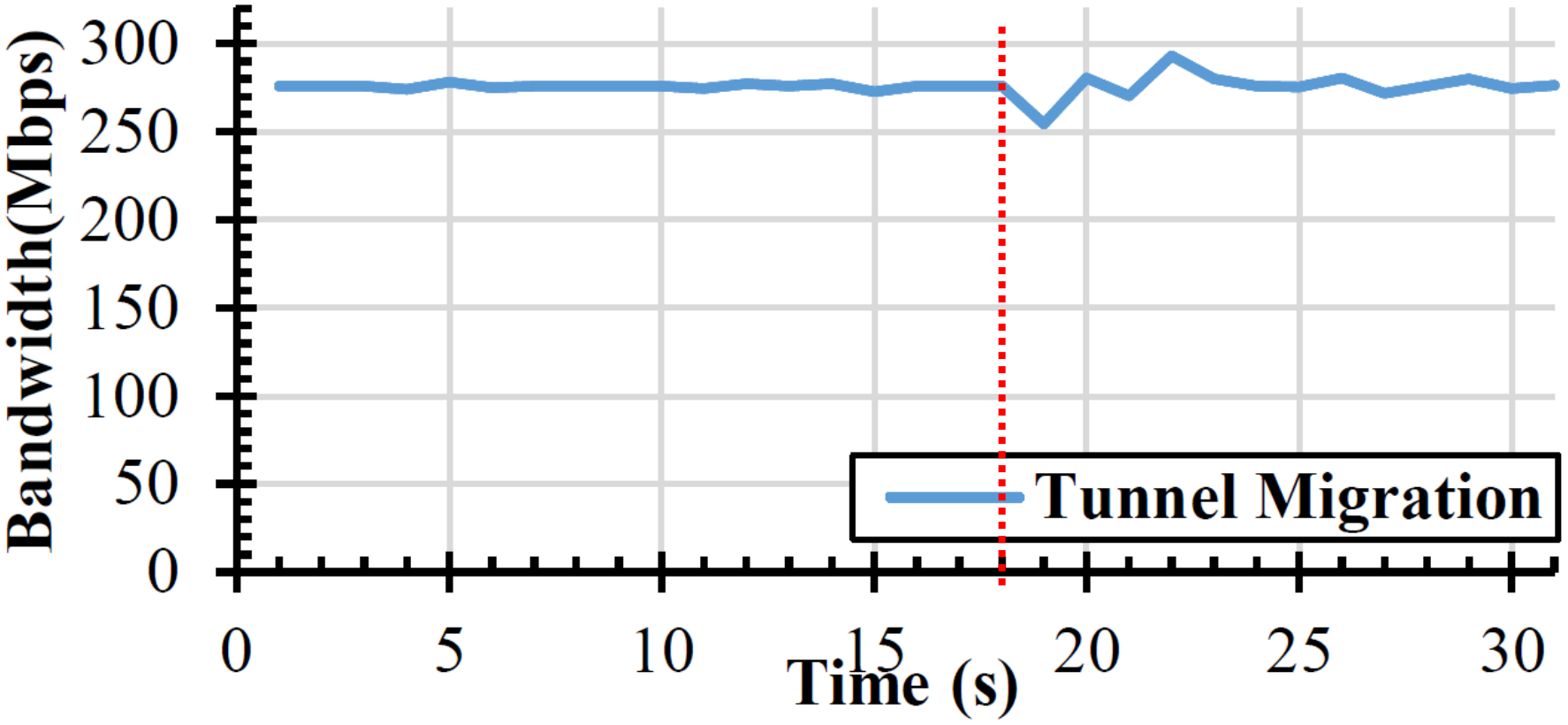
Evaluation

Server specification

- 16-core Intel Xeon E5-2650 v2 at 2.6Ghz
- Mellanox Connect-3 Pro 40Gbps
- Windows Server 2012 R2
- Hyper-V



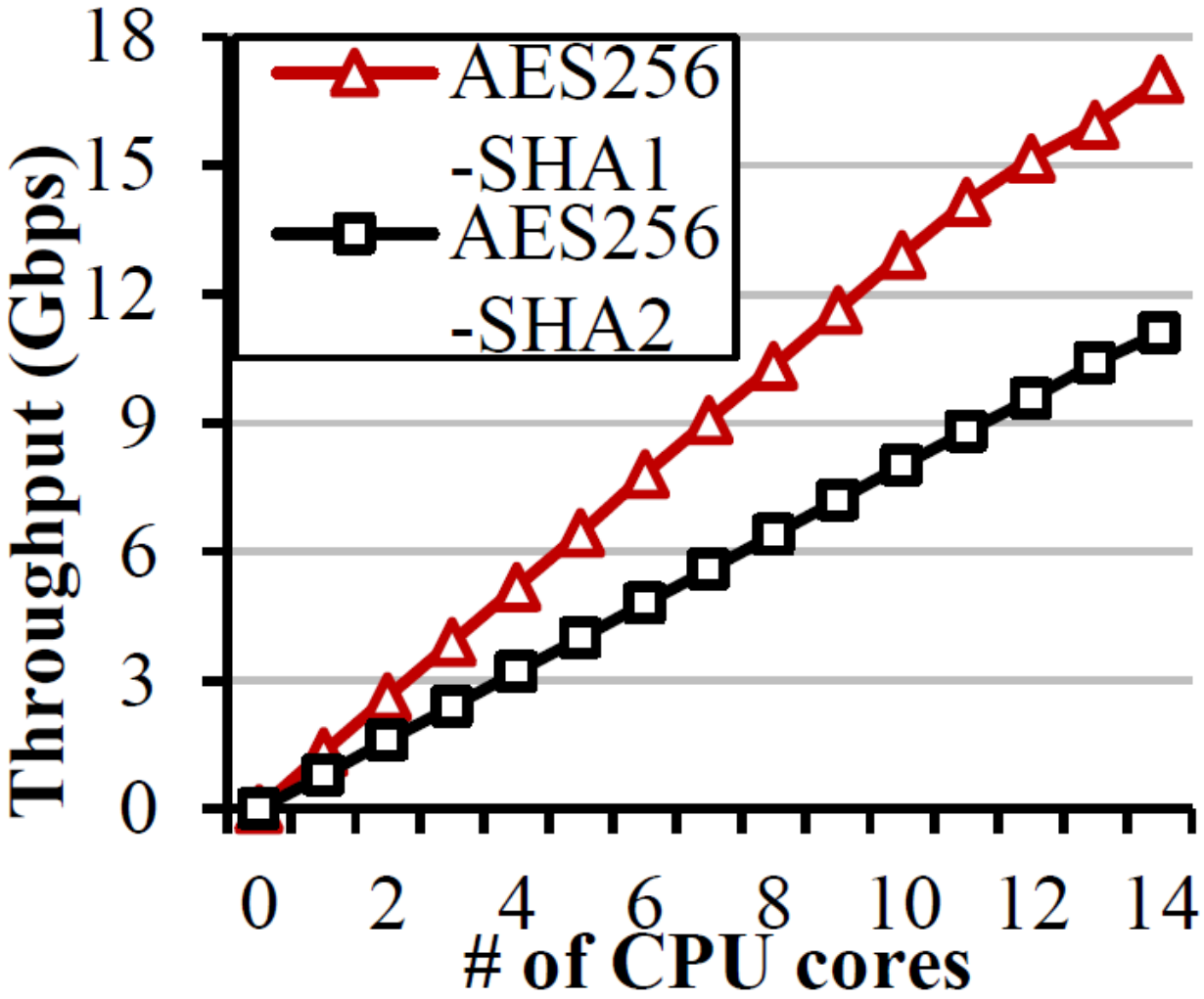
Migration Does Not Degrade Throughput



Rekeying:
192 ms

Key Distribution:
56 ms

GPN Throughput Scales Linearly



To achieve 10 Gbps,

AES256-CBC/SHA1:
8 cores

AES256-CBC/SHA2:
12 cores

Provisioning Simulation

- Measured the average throughput of IPsec gateways every minute in one data center for a day
- Collected the throughput trace of 170 tunnels and injected to our simulator, which replay the traffic trace
- Simulated our provisioning algorithm with different consolidation intervals

Protego Saves a Large Amount of Resources

Consolidation Interval: 3 min – 60 min

- Throughput Guarantee (99% of tunnels): 90.21 % – 98.63 %
- Resource Saving: **81.72 % – 88.00 %**

Summary

- IPsec gateway is an essential and common component for cloud providers to offer virtual network services
- Protego is a software IPsec Gateway that serves multiple IPsec tunnels using shared resources for better resource utilization
- Protego saves a significant amount of resources with the separation of control and data plane and seamless tunnel migration by rekeying

Microsoft® Research

KAIST

HUAWEI